

**The Geochemist's Workbench®**  
Release 17

**GWB**  
**Command**  
**Reference**



**The Geochemist's Workbench®**  
**Release 17**

---

---

**GWB**  
**Command**  
**Reference**

---

---

**Craig M. Bethke**  
**Brian Farrell**

**Aqueous Solutions, LLC**  
**Champaign, Illinois**

**Printed March 26, 2024**

This document © Copyright 2024 by Aqueous Solutions LLC. All rights reserved. Earlier editions copyright 2000–2021. This document may be reproduced freely to support any licensed use of the GWB software package.

Software copyright notice: Programs GSS, Rxn, Act2, Tact, SpecE8, Gtplot, TEdit, React, Phase2, P2plot, X1t, X2t, Xtplot, and ChemPlugin © Copyright 1983–2024 by Aqueous Solutions LLC. An unpublished work distributed via trade secrecy license. All rights reserved under the copyright laws.

The Geochemist's Workbench®, ChemPlugin™, We put bugs in our software™, and The Geochemist's Spreadsheet™ are a registered trademark and trademarks of Aqueous Solutions LLC; Microsoft®, MS®, Windows 11®, and Windows 10® are registered trademarks of Microsoft Corporation; PostScript® is a registered trademark of Adobe Systems, Inc. Other products mentioned in this document are identified by the trademarks of their respective companies; the authors disclaim responsibility for specifying which marks are owned by which companies. The software uses zlib © 1995-2005 Jean-Loup Gailly and Mark Adler, and Expat © 1998-2006 Thai Open Source Center Ltd. and Clark Cooper.

The GWB software was originally developed by the students, staff, and faculty of the Hydrogeology Program in the Department of Geology at the University of Illinois Urbana-Champaign. The package is currently developed and maintained by Aqueous Solutions LLC at the University of Illinois Research Park.

Address inquiries to

Aqueous Solutions LLC  
301 North Neil Street, Suite 400  
Champaign, IL 61820 USA

Warranty: The Aqueous Solutions LLC warrants only that it has the right to convey license to the GWB software. Aqueous Solutions makes no other warranties, express or implied, with respect to the licensed software and/or associated written documentation. Aqueous Solutions disclaims any express or implied warranties of merchantability, fitness for a particular purpose, and non-infringement. Aqueous Solutions does not warrant, guarantee, or make any representations regarding the use, or the results of the use, of the Licensed Software or documentation in terms of correctness, accuracy, reliability, currentness, or otherwise. Aqueous Solutions shall not be liable for any direct, indirect, consequential, or incidental damages (including damages for loss of profits, business interruption, loss of business information, and the like) arising out of any claim by Licensee or a third party regarding the use of or inability to use Licensed Software. The entire risk as to the results and performance of Licensed Software is assumed by the Licensee. See License Agreement for complete details.

License Agreement: Use of the GWB is subject to the terms of the accompanying License Agreement. Please refer to that Agreement for details.

Cover photo: Salinas de Janubio by Jorg Hackemann.

# Contents

---

## Chapter List

<b>1 Introduction</b> . . . . .	1
<b>2 Rxn Commands</b> . . . . .	3
<b>3 Act2 Commands</b> . . . . .	27
<b>4 Tact Commands</b> . . . . .	57
<b>5 SpecE8 Commands</b> . . . . .	85
<b>6 React Commands</b> . . . . .	127
<b>7 Phase2 Commands</b> . . . . .	189
<b>8 X1t Commands</b> . . . . .	249
<b>9 X2t Commands</b> . . . . .	319
<b>Index</b> . . . . .	397



# Contents

---

<b>1 Introduction</b>	<b>1</b>	2.30 react . . . . .	13
1.1 GWB commands . . . . .	1	2.31 read . . . . .	14
1.2 Field variables . . . . .	1	2.32 remove . . . . .	14
<b>2 Rxn Commands</b>	<b>3</b>	2.33 report . . . . .	15
2.1 activity . . . . .	3	2.34 reset . . . . .	15
2.2 add . . . . .	3	2.35 resume . . . . .	15
2.3 alter . . . . .	4	2.36 reverse . . . . .	16
2.4 block . . . . .	5	2.37 save . . . . .	16
2.5 chdir . . . . .	5	2.38 script . . . . .	16
2.6 clear . . . . .	6	2.39 short . . . . .	17
2.7 clipboard . . . . .	6	2.40 show . . . . .	17
2.8 couple . . . . .	6	2.41 solid_solution . . . . .	18
2.9 data . . . . .	7	2.42 sorbed_mass . . . . .	19
2.10 decouple . . . . .	7	2.43 stoich_IS . . . . .	19
2.11 Eh . . . . .	7	2.44 suffix . . . . .	20
2.12 explain . . . . .	8	2.45 surface_capacitance . . . . .	20
2.13 factor . . . . .	8	2.46 surface_charge . . . . .	21
2.14 fugacity . . . . .	8	2.47 surface_data . . . . .	21
2.15 go . . . . .	9	2.48 surface_potential . . . . .	22
2.16 help . . . . .	9	2.49 swap . . . . .	22
2.17 history . . . . .	9	2.50 system . . . . .	23
2.18 ionic_strength . . . . .	10	2.51 temperature . . . . .	24
2.19 log . . . . .	10	2.52 unalter . . . . .	24
2.20 long . . . . .	10	2.53 unswap . . . . .	25
2.21 molality . . . . .	11	2.54 usgovt . . . . .	25
2.22 output . . . . .	11	2.55 xss . . . . .	25
2.23 pause . . . . .	11	<b>3 Act2 Commands</b>	<b>27</b>
2.24 pe . . . . .	11	3.1 activity . . . . .	27
2.25 pH . . . . .	12	3.2 add . . . . .	27
2.26 polymorphs . . . . .	12	3.3 alter . . . . .	28
2.27 pwd . . . . .	13	3.4 background . . . . .	29
2.28 quit . . . . .	13	3.5 banner . . . . .	29
2.29 ratio . . . . .	13	3.6 chdir . . . . .	29

## Contents

---

3.7	clear	30	3.51	solid_solution	46
3.8	clipboard	30	3.52	span	47
3.9	color_by_sample	30	3.53	speciate	48
3.10	couple	30	3.54	suffix	48
3.11	custom	31	3.55	suppress	49
3.12	data	31	3.56	swap	49
3.13	decouple	31	3.57	system	50
3.14	diagram	32	3.58	temperature	50
3.15	Eh	32	3.59	title	51
3.16	explain	33	3.60	trace	51
3.17	extrapolate	33	3.61	truncate	52
3.18	font	33	3.62	unalter	52
3.19	fugacity	34	3.63	unspeciate	52
3.20	gases	34	3.64	unsuppress	53
3.21	go	35	3.65	unswap	53
3.22	grep	35	3.66	usgovt	54
3.23	gridlines	35	3.67	water_limits	54
3.24	help	35	3.68	x-axis	54
3.25	history	36	3.69	y-axis	55
3.26	labels	36			
3.27	legend	36	<b>4 Tact Commands</b>	<b>57</b>	
3.28	line	37	4.1	activity	57
3.29	log	38	4.2	add	57
3.30	minerals	38	4.3	alter	58
3.31	mosaic_bounds	38	4.4	background	59
3.32	mosaic_labels	39	4.5	banner	59
3.33	paper	39	4.6	chdir	59
3.34	pause	39	4.7	clear	60
3.35	pe	40	4.8	clipboard	60
3.36	pH	40	4.9	color_by_sample	60
3.37	polymorphs	40	4.10	couple	60
3.38	pressure	41	4.11	custom	61
3.39	print	41	4.12	data	61
3.40	pwd	41	4.13	decouple	61
3.41	quit	42	4.14	diagram	62
3.42	ratio	42	4.15	Eh	62
3.43	read	42	4.16	explain	63
3.44	remove	43	4.17	extrapolate	63
3.45	reset	43	4.18	font	63
3.46	resume	43	4.19	fugacity	64
3.47	save	44	4.20	gases	64
3.48	scatter	44	4.21	go	65
3.49	script	44	4.22	grep	65
3.50	show	45	4.23	gridlines	65
			4.24	help	65



4.25	history	66	<b>5 SpecE8 Commands</b>	<b>85</b>
4.26	labels	66	5.1 <unit>	85
4.27	legend	66	5.2 <isotope>	86
4.28	line	67	5.3 activity	87
4.29	log	68	5.4 add	87
4.30	minerals	68	5.5 alkalinity	88
4.31	mosaic_bounds	68	5.6 alter	88
4.32	mosaic_labels	69	5.7 b-dot	89
4.33	paper	69	5.8 balance	90
4.34	pause	69	5.9 carbon-13	90
4.35	pe	70	5.10 chdir	90
4.36	pH	70	5.11 clear	91
4.37	polymorphs	70	5.12 clipboard	91
4.38	pressure	71	5.13 conductivity	91
4.39	print	71	5.14 couple	92
4.40	pwd	71	5.15 data	92
4.41	quit	72	5.16 decouple	92
4.42	ratio	72	5.17 density	93
4.43	read	72	5.18 dual_porosity	94
4.44	remove	73	5.19 Eh	94
4.45	reset	73	5.20 epsilon	95
4.46	resume	73	5.21 exchange_capacity	95
4.47	save	74	5.22 explain	96
4.48	scatter	74	5.23 extrapolate	96
4.49	script	74	5.24 fugacity	96
4.50	show	75	5.25 go	97
4.51	solid_solution	76	5.26 grep	97
4.52	span	77	5.27 gtplot	97
4.53	speciate	78	5.28 h-m-w	98
4.54	suffix	78	5.29 help	98
4.55	suppress	78	5.30 history	98
4.56	swap	79	5.31 hydrogen-2	98
4.57	system	79	5.32 inert	99
4.58	title	80	5.33 isotope_data	99
4.59	trace	80	5.34 itmax	100
4.60	truncate	81	5.35 Kd	100
4.61	unalter	81	5.36 log	100
4.62	unspeciate	82	5.37 mobility	101
4.63	unsuppress	82	5.38 oxygen-18	102
4.64	unswap	83	5.39 pause	102
4.65	usgovt	83	5.40 pe	102
4.66	water_limits	83	5.41 permeability	103
4.67	x-axis	84	5.42 pH	104
4.68	y-axis	84	5.43 phrqpitz	104

## Contents

---

5.44	pitz_dgamma	104	6.2	<isotope>	129
5.45	pitz_precon	105	6.3	activity	129
5.46	pitz_relax	105	6.4	add	130
5.47	plot	105	6.5	alkalinity	130
5.48	polymorphs	106	6.6	alter	130
5.49	porosity	106	6.7	audit	132
5.50	press_model	106	6.8	b-dot	132
5.51	print	107	6.9	balance	132
5.52	pwd	108	6.10	carbon-13	133
5.53	quit	108	6.11	chdir	133
5.54	ratio	108	6.12	clear	134
5.55	read	109	6.13	clipboard	134
5.56	remove	109	6.14	conductivity	134
5.57	report	110	6.15	couple	134
5.58	reset	110	6.16	cpr	135
5.59	resume	110	6.17	cpu_max	135
5.60	save	111	6.18	cpw	135
5.61	script	111	6.19	data	136
5.62	segregate	112	6.20	decouple	136
5.63	show	112	6.21	delQ	137
5.64	simax	113	6.22	delxi	137
5.65	solid_solution	114	6.23	density	138
5.66	sorbate	115	6.24	dual_porosity	139
5.67	span	115	6.25	dump	140
5.68	suffix	116	6.26	dx_init	140
5.69	sulfur-34	117	6.27	dxplot	141
5.70	suppress	117	6.28	dxprint	141
5.71	surface_capacitance	118	6.29	Eh	141
5.72	surface_data	118	6.30	end-dump	142
5.73	surface_potential	119	6.31	epsilon	142
5.74	swap	119	6.32	exchange_capacity	142
5.75	system	120	6.33	explain	143
5.76	TDS	121	6.34	explain_step	143
5.77	temperature	121	6.35	extrapolate	143
5.78	timax	121	6.36	fix	144
5.79	title	122	6.37	flash	144
5.80	unalter	122	6.38	flow-through	145
5.81	unsegregate	122	6.39	flush	145
5.82	unsuppress	123	6.40	fugacity	145
5.83	unswap	123	6.41	go	146
5.84	usgovt	123	6.42	grep	146
5.85	volume	124	6.43	gtplot	146
			6.44	h-m-w	147
			6.45	heat_source	147
<b>6</b>	<b>React Commands</b>	<b>127</b>			
6.1	<unit>	127			

6.46	help	148	6.90	slide	173
6.47	history	148	6.91	solid_solution	173
6.48	hydrogen-2	149	6.92	sorbate	175
6.49	inert	149	6.93	span	175
6.50	isotope_data	150	6.94	start_date	176
6.51	itmax	150	6.95	start_time	177
6.52	Kd	150	6.96	step_increase	177
6.53	kinetic	151	6.97	step_max	177
6.54	log	155	6.98	suffix	177
6.55	mobility	156	6.99	sulfur-34	178
6.56	no-precip	157	6.100	suppress	178
6.57	nswap	157	6.101	surface_capacitance	179
6.58	oxygen-18	157	6.102	surface_data	180
6.59	pause	158	6.103	surface_potential	180
6.60	pe	158	6.104	swap	181
6.61	permeability	159	6.105	system	182
6.62	pH	159	6.106	TDS	182
6.63	phrqpitz	160	6.107	temperature	183
6.64	pickup	160	6.108	theta	183
6.65	pitz_dgamma	161	6.109	timax	184
6.66	pitz_precon	161	6.110	time	184
6.67	pitz_relax	161	6.111	title	184
6.68	plot	162	6.112	unalter	185
6.69	pluses	162	6.113	unsegregate	185
6.70	polymorphs	162	6.114	unsuppress	185
6.71	porosity	163	6.115	unswap	186
6.72	precip	163	6.116	usgovt	186
6.73	press_model	164	6.117	volume	186
6.74	print	164			
6.75	pwd	165	<b>7</b>	<b>Phase2 Commands</b>	<b>189</b>
6.76	quit	165	7.1	<unit>	189
6.77	ratio	166	7.2	<isotope>	191
6.78	react	166	7.3	activity	191
6.79	reactants	167	7.4	add	192
6.80	read	168	7.5	alkalinity	192
6.81	remove	168	7.6	alter	192
6.82	report	169	7.7	b-dot	194
6.83	reset	169	7.8	balance	194
6.84	resume	169	7.9	carbon-13	194
6.85	save	170	7.10	chdir	195
6.86	script	170	7.11	clear	195
6.87	segregate	171	7.12	clipboard	195
6.88	show	171	7.13	conductivity	196
6.89	simax	172	7.14	couple	196
			7.15	cpr	196

## Contents

---

7.16	cpu_max	197	7.60	pH	220
7.17	cpw	197	7.61	phrqpitz	220
7.18	data	197	7.62	pitz_dgamma	221
7.19	decouple	198	7.63	pitz_precon	221
7.20	delQ	198	7.64	pitz_relax	221
7.21	delxi	198	7.65	plot	222
7.22	density	199	7.66	pluses	222
7.23	dump	200	7.67	polymorphs	222
7.24	dx_init	200	7.68	porosity	223
7.25	dxprint	201	7.69	precip	223
7.26	Eh	201	7.70	press_model	224
7.27	epsilon	201	7.71	print	224
7.28	exchange_capacity	202	7.72	pwd	225
7.29	explain	202	7.73	quit	225
7.30	explain_step	203	7.74	ratio	225
7.31	extrapolate	203	7.75	react	226
7.32	fix	204	7.76	reactants	227
7.33	flash	204	7.77	read	228
7.34	flow-through	205	7.78	remove	228
7.35	flush	205	7.79	reset	229
7.36	fugacity	205	7.80	resume	229
7.37	go	206	7.81	save	229
7.38	grep	206	7.82	scope	230
7.39	h-m-w	206	7.83	script	230
7.40	heat_source	207	7.84	segregate	231
7.41	help	208	7.85	show	231
7.42	history	208	7.86	simax	232
7.43	hydrogen-2	208	7.87	slide	233
7.44	inert	209	7.88	solid_solution	233
7.45	isotope_data	209	7.89	sorbate	235
7.46	itmax	210	7.90	span	235
7.47	Kd	210	7.91	start_date	236
7.48	kinetic	211	7.92	start_time	237
7.49	log	215	7.93	step_increase	237
7.50	mobility	216	7.94	step_max	237
7.51	no-precip	217	7.95	suffix	238
7.52	nswap	217	7.96	sulfur-34	238
7.53	Nx	217	7.97	suppress	238
7.54	Ny	217	7.98	surface_capacitance	239
7.55	oxygen-18	218	7.99	surface_data	240
7.56	p2plot	218	7.100	surface_potential	240
7.57	pause	219	7.101	swap	241
7.58	pe	219	7.102	system	242
7.59	permeability	219	7.103	TDS	242

7.104	temperature	243	8.32	dxplot	265
7.105	theta	243	8.33	dxprint	266
7.106	threads	244	8.34	Eh	267
7.107	timax	244	8.35	epsilon	267
7.108	time	244	8.36	exchange_capacity	267
7.109	title	245	8.37	explain	268
7.110	unalter	245	8.38	explain_step	268
7.111	unsegregate	245	8.39	extrapolate	269
7.112	unsuppress	246	8.40	fix	269
7.113	unswap	246	8.41	fugacity	270
7.114	usgovt	246	8.42	go	270
7.115	volume	247	8.43	grep	271
<b>8</b>	<b>X1t Commands</b>	<b>249</b>	8.44	h-m-w	271
8.1	<unit>	249	8.45	head_drop	271
8.2	<isotope>	251	8.46	heat_source	272
8.3	activity	252	8.47	height	273
8.4	add	252	8.48	help	274
8.5	alkalinity	252	8.49	history	274
8.6	alter	253	8.50	hydrogen-2	274
8.7	audit	254	8.51	inert	275
8.8	b-dot	255	8.52	interval	275
8.9	balance	255	8.53	isotope_data	276
8.10	carbon-13	255	8.54	itmax	276
8.11	chdir	256	8.55	Kd	277
8.12	clear	256	8.56	kinetic	277
8.13	clipboard	256	8.57	left	282
8.14	conductivity	257	8.58	length	283
8.15	couple	257	8.59	log	283
8.16	Courant	257	8.60	mobility	283
8.17	cpr	258	8.61	no-precip	285
8.18	cpu_max	258	8.62	nswap	285
8.19	cpw	258	8.63	Nx	285
8.20	data	259	8.64	oxygen-18	285
8.21	decouple	259	8.65	pause	286
8.22	delQ	260	8.66	pe	286
8.23	deltar	260	8.67	permeability	287
8.24	deltax	260	8.68	pH	288
8.25	delxi	261	8.69	phrqpitz	288
8.26	density	261	8.70	pitz_dgamma	288
8.27	diffusion_coef	262	8.71	pitz_precon	289
8.28	discharge	262	8.72	pitz_relax	289
8.29	dispersivity	263	8.73	plot	289
8.30	dual_porosity	264	8.74	pluses	290
8.31	dx_init	265	8.75	polymorphs	290

## Contents

---

8.76	porosity	290	8.120	time	315
8.77	precip	291	8.121	title	315
8.78	press_model	291	8.122	unalter	316
8.79	print	292	8.123	unsegregate	316
8.80	pwd	293	8.124	unsuppress	316
8.81	quit	293	8.125	unswap	317
8.82	radial	293	8.126	usgovt	317
8.83	ratio	294	8.127	width	318
8.84	react	294	8.128	xtplot	318
8.85	read	295	8.129	Xstable	318
8.86	remove	296			
8.87	report	297	<b>9</b>	<b>X2t Commands</b>	<b>319</b>
8.88	reset	297	9.1	<unit>	319
8.89	resume	297	9.2	<isotope>	321
8.90	right	298	9.3	activity	322
8.91	save	299	9.4	add	322
8.92	scope	299	9.5	alkalinity	322
8.93	script	300	9.6	alter	323
8.94	segregate	301	9.7	audit	324
8.95	show	301	9.8	b-dot	325
8.96	simax	302	9.9	balance	325
8.97	slide	303	9.10	bottom	325
8.98	solid_solution	303	9.11	carbon-13	326
8.99	sorbate	305	9.12	chdir	327
8.100	span	305	9.13	clear	327
8.101	spherical	306	9.14	clipboard	327
8.102	start_date	307	9.15	conductivity	328
8.103	start_time	307	9.16	couple	328
8.104	step_increase	307	9.17	Courant	328
8.105	step_max	308	9.18	cpr	329
8.106	suffix	308	9.19	cpu_max	329
8.107	sulfur-34	308	9.20	cpw	329
8.108	suppress	309	9.21	data	330
8.109	surface_capacitance	309	9.22	decouple	330
8.110	surface_data	310	9.23	delQ	331
8.111	surface_potential	311	9.24	deltar	331
8.112	swap	311	9.25	deltax	331
8.113	system	312	9.26	deltay	332
8.114	TDS	312	9.27	delxi	333
8.115	temperature	313	9.28	density	333
8.116	thermal_cond	314	9.29	diffusion_coef	334
8.117	theta	314	9.30	discharge	334
8.118	threads	314	9.31	dispersivity	336
8.119	timax	315	9.32	dual_porosity	337
			9.33	dx_init	338

9.34	dxplot	338	9.78	plot	364
9.35	dxprint	339	9.79	pluses	365
9.36	Eh	340	9.80	polymorphs	365
9.37	epsilon	340	9.81	porosity	365
9.38	exchange_capacity	340	9.82	precip	366
9.39	explain	341	9.83	press_model	366
9.40	explain_step	341	9.84	print	367
9.41	extrapolate	342	9.85	pwd	368
9.42	fix	342	9.86	quit	368
9.43	fugacity	343	9.87	radial	368
9.44	go	343	9.88	ratio	368
9.45	grep	344	9.89	react	369
9.46	h-m-w	344	9.90	read	370
9.47	head_drop	344	9.91	remove	371
9.48	heat_source	345	9.92	report	372
9.49	height	346	9.93	reset	372
9.50	help	347	9.94	resume	372
9.51	history	347	9.95	right	373
9.52	hydrogen-2	347	9.96	save	374
9.53	inert	348	9.97	scope	375
9.54	interval	348	9.98	script	376
9.55	isotope_data	349	9.99	segregate	376
9.56	itmax	349	9.100	show	377
9.57	Kd	350	9.101	simax	378
9.58	kinetic	350	9.102	slide	378
9.59	left	355	9.103	solid_solution	379
9.60	length	356	9.104	sorbate	380
9.61	log	356	9.105	span	380
9.62	mobility	357	9.106	start_date	382
9.63	modflow	358	9.107	start_time	382
9.64	no-precip	359	9.108	step_increase	382
9.65	nswap	359	9.109	step_max	383
9.66	Nx	359	9.110	suffix	383
9.67	Ny	360	9.111	sulfur-34	383
9.68	off-diag	360	9.112	suppress	384
9.69	oxygen-18	360	9.113	surface_capacitance	384
9.70	pause	361	9.114	surface_data	385
9.71	pe	361	9.115	surface_potential	386
9.72	permeability	362	9.116	swap	386
9.73	pH	363	9.117	system	387
9.74	phrqpitz	363	9.118	TDS	387
9.75	pitz_dgamma	363	9.119	temperature	388
9.76	pitz_precon	364	9.120	thermal_cond	389
9.77	pitz_relax	364	9.121	theta	389

## Contents

---

9.122 threads . . . . .	389
9.123 timax . . . . .	390
9.124 time . . . . .	390
9.125 title . . . . .	390
9.126 top . . . . .	391
9.127 unalter . . . . .	392
9.128 unsegregate . . . . .	392
9.129 unsuppress . . . . .	392
9.130 unswap . . . . .	393
9.131 usgovt . . . . .	393
9.132 well . . . . .	394
9.133 width . . . . .	396
9.134 xplot . . . . .	396
9.135 Xstable . . . . .	396
<b>Index</b>	<b>397</b>



# Introduction

---

This **GWB Command Reference** contains specifics of the formats of the various GWB commands. The manual is intended as a supplement to the GWB manuals: **GWB Essentials Guide**, **GWB Reaction Modeling Guide**, **GWB Reactive Transport Modeling Guide**, and the **GWB Reference Manual**.

## 1.1 GWB commands

Chapters of the manual contain details of the formats of the various commands available in **Rxn**, **Act2**, **Tact**, **SpecE8**, **React**, **Phase2**, **X1t**, and **X2t**. Each command listed may be typed directly into the **Command** pane, using the syntax shown. Alternatively, almost all of the commands have GUI equivalents, and these are also shown.

Two or more commands may appear separated by semicolons on the same line of input, and you may continue one line onto another with a backslash (\). Comments may appear anywhere in the input; they begin with a pound sign (#) followed by a blank and extend to the end of the line. Equal signs appear in the commands for clarity and may be included or omitted as desired.

You need enter only enough characters to uniquely identify a command. You can, furthermore, use the spelling completion feature: touch “[tab]” or “[esc]” to have the program cycle through the possible completions, or Ctrl+D to list possible completions.

Within the command reference sections, the entry <value> signifies that you should enter a numerical value. You can enter values as numbers (e.g., 4.15, 3.8e-6), exponentials ( $10^{-6}$ ), fractions (1/2), percentages (25%), or hexadecimal values (0x3f847ae147ae147b).

## 1.2 Field variables

For programs **X1t** and **X2t**, some keywords may be set to an entry identified as <field variable>. These are keywords representing properties of the medium or the simulation initial conditions that can be set to vary over the domain. In other words, these are values that can be heterogeneous.

You can enter several things in a command in place of <field variable>:

- A value, in any of the forms described above.

## GWB Command Reference

---

- A question mark (?), to unset the variable.
- A mean  $\pm$  a standard deviation, in which case values are assigned to nodal blocks randomly, within a Gaussian distribution.
- The name of a dataset containing a table of values, one for each nodal block in the simulation.
- An expression used to calculate values at individual nodal blocks.
- A basic script used to calculate values.
- A compiled function in a dll, or dynamic load library.

A subset of field variables can be evaluated not only initially, but upon undertaking each time step over the course of a run, when set with an equation, script, or function. These transient field variables, indicated with <steady | transient>, are available to **X1t** and **X2t**, as well as **React** and **Phase2**. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation. In contrast, in the default steady behavior, the program evaluates the variable just once, at the start of the run.

For more information about setting field variables, see the **Heterogeneity** appendix in the **GWB Reactive Transport Modeling Guide**.

# Rxn Commands

---

## 2.1 activity

**activity** <species> = <value | ?>

**Basis** →  ▼ activity ▼

Use the “activity” command (abbrev.: “a”) to constrain the activity of an aqueous species, the water activity, the activity of an end member in a continuous solid solution, or the activity of a surface complex from an ion exchange model. Examples:

```
activity Na+ = 0.3  
a H2O = 0.9  
log a H+ = -5
```

Use “?” to unset an activity value. See also the “pH”, “Eh”, “pe”, “molality”, “fugacity”, “sorbed\_mass”, “xss”, and “ratio” commands.

## 2.2 add

**add** <basis species>

**Basis** →

Use the “add” command to include a basis species in the calculation. Example:

```
add HCO3-
```

See also the “swap”, “activity”, “fugacity”, “ratio”, “pH”, “pe”, and “Eh” commands.

## 2.3 alter

```

alter <species | mineral | gas> <log Ks>
alter <species | mineral | gas> <poly coefs> TminK = <value> TmaxK = <value>
alter <surface species> <poly coefs> TminK = <value> TmaxK = <value>
alter <surface species> logK = <value> dlogK/dT = <value>
alter <exchange species> beta = <value>
alter <sorbed species> Kd = <value>
alter <sorbed species> Kf = <value> nf = <value>
alter <solid solution> <type> <discrete | continuous> \
  <from <value> to <value> step <value>> <<variable> = <value>>

Config → Alter Log Ks...
Config → Solid Solutions...

```

Use the “alter” command to change the temperature expansion for a species’, mineral’s, or gas’ log  $K$ , to adjust the stability of a surface species, or to change the properties of a solid solution.

Temperature expansions for log  $K$ s are given in the thermo dataset either as  $T$ -tables, or by up to six coefficients of a polynomial. When the current thermo dataset uses  $T$ -tables, you list replacement values at each of the principal temperatures specified in the dataset, most commonly 0°C, 25°C, 60°C, 100°C, 150°C, 200°C, 250°C, and 300°C. Example:

```
alter Anhydrite -4.3009 -4.4199 -4.7126 -5.1758 -6.2299 500 500 500
```

Values of “500” represent a lack of data at the corresponding temperature.

For a thermo dataset constructed from polynomial expansions, on the other hand, list instead up to six polynomial coefficients

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4
```

You can optionally append a temperature range of validity, in Kelvins, for the polynomial:

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4 TminK= 293 TmaxK= 383
```

Absent a range, the polynomial is taken to span the principal temperatures. You may adjust a reaction’s temperature range without specifying polynomial coefficients, but setting only coefficients resets the temperature range.

For a surface complex, use the “alter” command to set log  $K$ , its temperature derivative, or both

```
alter >(w)FeOCa+ logK = 6.0 dlogK/dT = 0.02
```

whereas for an ion exchange reaction, you set the selectivity coefficient  $\beta$

```
alter >X2:Ca beta = .033
```

directly, rather than as a logarithm. For a sorbing species, set  $K_d$  or, for a Freundlich species,  $K_f$ ,  $n_f$ , or both, as follows

```
alter >Pb++ Kd = .03
alter >Sr++ Kf = .015 nf = .8
```

For a solid solution, use the “alter” command to set the activity coefficient model, discrete or continuous behavior, the composition range in terms of the mole fraction of the most recently specified end member, and parameters for the chosen activity model, as described for the “solid\_solution” command. A solid solution “my\_ss” might be set as a subregular Guggenheim solution with the command

```
alter my_ss guggenheim a0 = 1 a1 = 2
```

You may prefer to use the “alter” command to adjust an entry from the thermo database, and the “solid\_solution” command to reconfigure a user-defined solution, but the two commands work interchangeably for this purpose.

Type “show alter” to list altered species or solutions and their settings; the “unalter” command reverses the process.

## 2.4 block

```
block <append>
```

**Run** → **Block to clipboard** → **Overwrite** or **Append**

Use the “block” command to create blocks of text for species, minerals, and gases, in the form of entries in the thermo dataset. When you issue the command, **Rxn** copies the current reaction and associated log  $K$ 's to the clipboard, so you can paste the block directly into a thermo dataset open in a text editor or **TEdit**. With the “append” argument, the block is written at the end of any previous blocks on the clipboard, from previous invocations of the “block” command. The command is convenient for updating and manipulating the contents of a thermo dataset, such as if you were to add redox coupling reactions.

## 2.5 chdir

```
chdir <directory> <ALL>
```

**File** → **Working Directory...**

## GWB Command Reference

---

Use the “chdir” command (abbrev.: “work\_dir”, “cd”) to change the working directory. The program reads input scripts relative to the current working directory and writes output into it. Typing the command “chdir” without an argument causes the program to display the name of the working directory. The “ALL” argument changes the working directories of all of the GWB apps. The command

```
chdir ~
```

changes to the user’s home directory, if one is defined by the operating system.

### 2.6 clear

```
clear
```

```
View → Clear Screen
```

Use the “clear” command (abbrev.: “c”) or right-click in the text window and choose **Clear Screen** to clear it.

### 2.7 clipboard

```
clipboard
```

```
Edit → Copy
```

The “clipboard” command causes the contents of the **Results** pane to be copied to the clipboard, from where it can be pasted into various types of documents.

### 2.8 couple

```
couple <redox species | element(s) | ALL>
```

```
Config → Redox Couples... → select couple(s) → select with...
```

Use the “couple” command to enable any redox coupling reactions that have been disabled with the “decouple” command. You specify one or more redox species or elements. For example, the command

```
couple Carbon
```

couple all redox reactions involving the element carbon. Argument “ALL” enables all of the coupling reactions in the thermo dataset.

## 2.9 data

**data** <thermo dataset> <verify>

**File** → **Open** → **Thermo Data...**

Use the “data” command to change the input file of thermodynamic data. Example:

```
data "..\my_thermo.tdat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. The “verify” option causes the program to read the named dataset only if it has not already been read.

## 2.10 decouple

**decouple** <redox species | element(s) | **ALL**>

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “decouple” command to disable the coupling reactions for one or more redox species, in order to calculate a model assuming redox disequilibrium. The redox species then become available for use as basis species and may be constrained independently of the original basis entries. You can disable as many coupling reactions as you want.

You specify either one or more redox species or elements. For example, the command

```
decouple Carbon
```

decouples all redox reactions involving the element carbon. Argument “ALL” disables all of the coupling reactions in the thermo dataset. Use the “couple” command to enable coupling reactions, once they have been disabled.

## 2.11 Eh

**Eh =** <value in volts | ?>

**Basis** →  ▼ Eh ▼

Use the “Eh” command to set Eh for the calculation. Example:

```
Eh = 1/2
```

sets the system's oxidation state to correspond to an Eh of 0.5 volts. Use "?" to unset an Eh value. See also the "activity", "pH", "pe", "fugacity", and "ratio" commands.

## 2.12 explain

**explain** <species | mineral(s) | solid solution(s) | gas(es) | surface species>

**Config** → **Show...** → *select species*

Use the "explain" command to get more information (such as the mole weight of a species, a mineral's formula, mole volume, and density, or a solid solution's end-member minerals) about species, minerals, solid solutions, and gases in the dataset. Example:

```
explain Analcime
```

## 2.13 factor

**factor** = <reaction multiplier | ? | ->

**factor** <explicit> <reaction multiplier | ? | ->

**Basis** → factor reaction by  ▼ **reverse**

Use the "factor" command to multiply reaction coefficients by a numeric value. Without the "explicit" qualifier, the current factor is multiplied by the value. Negative values reverse the reaction. Examples:

```
factor 5/2
factor -1.5
factor -
factor ?
factor explicit 1.7
```

The third example reverses the current reaction, and the fourth resets the multiplier to 1. The last example explicitly sets the factor to the value, rather than multiplying the current factor by the value. See also the "reverse" command.

## 2.14 fugacity

**fugacity** <gas> = <value | ?>

**Basis** →  ▼ fugacity ▼



Use the “fugacity” command (abbrev.: “f”) to set gas fugacities (on an atm scale) in the calculation. Examples:

```
fugacity O2(g) = .2
f CO2(g) = 0.0003
log f S2(g) = -30
```

Use “?” to unset a fugacity value: See also the “activity”, “pH”, “Eh”, “pe”, and “ratio” commands.

## 2.15 go

go

Run → Go

Results → Run

Use the “go” command to initiate an **Rxn** calculation based on the current chemical system.

## 2.16 help

help <help | subject>

Help

Use the “help” command (also: “man”) to display online documentation for the **GWB** package. The command “help help” shows a list of the types of documentation available.

## 2.17 history

history

Run → History...

Use the “history” command to view a list of previously executed commands. You can re-execute commands in the **History...** dialog box by double-clicking on a command, selecting one or more commands, and then choosing **Execute**, or with a history substitution command, as explained in the **Command Line Interface** chapter in the **GWB Reference Manual**.

## 2.18 ionic\_strength

**ionic\_strength** = <value | ?>

**Basis** → ionic strength true  ▼ molal

Use the “ionic\_strength” command (also: “I”, “IS”) to set, in molal units, the solution ionic strength. Use the “?” argument to unset this value. Setting the ionic strength when **Rxn** is invoked in “long” mode has two effects. First, if a temperature has been set, it causes the program to use a version of Debye-Hückel theory to calculate the activity coefficients of any aqueous species in the equilibrium equation. As well, the setting partially constrains the water activity calculation of Wolery (1992b), when in use. The program uses the calculated values to simplify the equilibrium equation, if possible. Second, if you have set temperature to 25°C and set a value for surface charge or surface potential, **Rxn** will evaluate the Boltzman factor for two-layer surface complexation reactions and use this factor to simplify the equilibrium equation. See also the “stoich\_IS”, “temperature”, “surface\_data”, “surface\_charge”, and “surface\_potential” commands.

## 2.19 log

**log** <variable> = <value>

Use the “log” command to set variables on a logarithmic scale. Examples:

```
log fugacity O2(g) = -65
log activity U++++ = -10
react Cu+ log activity = -8
```

## 2.20 long

**long**

**Results** →

Use the “long” command to get a more detailed output than that produced by default (see the “short” command).

## 2.21 molality

**molality** <surface complex> = <value | ?>

**Basis** →  ▼ molality ▼

Use the “molality” command (abbrev.: “m”) to set for a surface complex from a two-layer or Langmuir model the mole number of the complex present per kilogram of solvent water. Example:

```
molality >(w)FeOH = 10^-2
```

An argument of “?” unsets the value. Use the “activity” command to constrain aqueous species, as well as surface complexes from an ion exchange model, and the “sorbed\_mass” command for surface complexes from a  $K_d$  or Freundlich sorption model. See also the “pH”, “Eh”, “pe”, “activity”, “fugacity”, “xss”, “ratio”, “sorbed\_mass”, and “surface\_data” commands.

## 2.22 output

**output** <append>

**Run** → **Output to file** → **Overwrite or Append**

Use the “output” command to make a run and write the results to a dataset in the working directory, instead of to the **Results** pane (as would be the case for the “go” command). The output appears in file “Rxn\_output.txt”. The “append” option causes the results to be written at the end of the dataset.

## 2.23 pause

**pause**

Use the “pause” command to cause **Rxn** to pause temporarily during input. This command is useful when you are debugging scripts.

## 2.24 pe

**pe** = <value | ?>

**Basis** →  ▼ pe ▼

## GWB Command Reference

---

Use the “pe” command to set oxidation state in the initial system in terms of pe.  
Example:

```
pe = 10
```

is equivalent to

```
log activity e- = -10
```

where “e-” is the electron. Use “?” to unset a pe value: See also the “activity”, “Eh”, “pH”, “fugacity”, and “ratio” commands.

### 2.25 pH

```
pH = <value | ?>
```

```
Basis →  ▼ pH ▼
```

Use the “pH” command to set pH in the initial system. Example:

```
pH = 5
```

is equivalent to

```
log activity H+ = -5
```

Use “?” to unset a pH value: See also the “activity”, “Eh”, “pe”, “fugacity”, and “ratio” commands.

### 2.26 polymorphs

```
polymorphs <mineral>
```

```
Config → Show... → Minerals → select mineral
```

Use the “polymorphs” command to list each polymorph (mineral identical in composition to the specified mineral) in the current database. Example:

```
polymorphs Albite
```

## 2.27 pwd

**pwd**

**File** → Working Directory...

The “pwd” command returns the name of the current working directory. The command has the same effect as typing “show directory”. See the “chdir” command.

## 2.28 quit

**quit**

**File** → Quit

Use the “quit” command (also: “end”, “q”) to exit the program. Touching **Ctrl+Q** serves the same purpose as the “quit” command.

## 2.29 ratio

**ratio** <act./fug. ratio> = <value | ?>

**Basis** →  ▼ ratio ▼

Use the “ratio” command to constrain an activity or fugacity ratios in the calculation. Example:

```
swap Ca++/Na+2 for Ca++
ratio Ca++/Na+2 = 0.2
```

Use “?” to unset a ratio value: See also the “activity”, “pH”, “Eh”, “pe”, and “fugacity” commands.

## 2.30 react

**react** <main species> **activity** = <value | ?>

**Basis** →  balance reaction for

???

in terms of

Aqueous...  
Mineral...  
Gas...

## GWB Command Reference

---

Use the “react” command (also: “reaction for”) to set the species to appear on the left side of the reaction, and optionally the species’ activity, fugacity, or mole fraction. Examples:

```
react Pyrite
react CH4(g) log fugacity = -1
reaction for H3PO4 activity = 1e-4
react Plag::Ab xss = .9
```

### 2.31 read

```
read <dataset>
```

**File** → **Open** → **Read Script...**

Use the “read” command to begin reading commands from a script stored in a dataset. Example:

```
read Cu_solubility
```

Control returns to the user after the script has been read, unless the script contains a “quit” command. You can also use the “read” command in place of the “data” or “surface\_data” command to read a thermo or surface reaction dataset.

When typing a “read” command, you can use the spelling completion feature to complete dataset names: touch “[tab]” or “[esc]” to cycle through the possible completions, or **Ctrl+D** to list possible completions.

### 2.32 remove

```
remove <basis specie(s)> <solid solution(s)>
remove basis <basis specie(s)>
remove solid_solution <solid solution(s)>
```

**Basis** or

**Config** → **Solid Solutions...** →

*select entry(ies)* →

Use the “remove” command (also: “rm”) to eliminate one or more basis entries from consideration in the calculation. Example:

```
remove Na+
remove Quartz Calcite
```

Components can be reentered into the basis using the “swap”, “add”, “activity”, and “fugacity” commands. You can also use the “remove” command to remove solid solutions configured in a run (but not those set in the thermo dataset):

```
remove solid_solution mySS
```

## 2.33 report

```
report <option>  
report set_digits <value>
```

Once the program has completed a calculation, you can use the “report” command to return aspects of the calculation results. You can use this command from the **Command** pane to query the program interactively. More significantly, the “report” command provides a means of communicating calculation results to control scripts and external programs running GWB programs by remote control.

For arguments available, see the **Report Command** chapter in the **GWB Reference Manual**. For details about control scripts and the plug-in feature, see the **Control Scripts** and **Plug-in Feature** chapters of the **GWB Reference Manual**. Use the “set\_digits” argument to set the number of significant digits in the values reported.

## 2.34 reset

```
reset  
  
File → Reset Configuration
```

Use the “reset” command to begin defining the chemical system again with a clean slate. Your current settings will be lost, and all options will be returned to their default states. The command, however, does not alter the setting for the thermo dataset.

## 2.35 resume

```
resume  
  
File → Resume
```

Use the “resume” command to restore the configuration from when you last exited the program. The program configuration is stored in a file “rxn\_resume.rxn” in your profile directory (e.g., “c:\Documents and Settings\jones\Application Data\GWB”). Your current configuration is discarded.

## 2.36 reverse

reverse

**Basis** → reverse

Use the “reverse” command to reverse the sense of the reaction, so that species on the left appear on the right and vice versa. The command is the same as “factor -1” or “factor -”.

## 2.37 save

save <dataset> <hex>

**File** → **Save As...**

Use the “save” command to write the current chemical system into a dataset in **Rxn** format commands. The dataset can be used as an **Rxn** input script. Examples:

```
save
save kspar.rxn
```

If you don't specify a dataset name, **Rxn** will ask for one. The optional keyword “hex” causes the program to output numbers as hexadecimal values.

## 2.38 script

script  
script end

Use the “script” command to mark the beginning, and optionally the end, of a control script. Control scripts differ from standard input files in that they can contain not only **Rxn** commands, but control structures such as loops and if-else branches. Control scripts follow the Tcl syntax, described in [www.tcl.tk](http://www.tcl.tk) and [mini.net/tcl](http://mini.net/tcl), as well as several widely available textbooks.

Within a control script, filenames are written with double rather than single backslashes. For example, a “read” command might appear as

```
read GWB_files\\My_file.rxn
```

within a control script.



## 2.39 short

short

**Results** → 

Use the “short” command to produce output in a concise format that consists of the balanced reaction and, if the temperature is set, the log  $K$ .

## 2.40 show

**show** <option>  
**show** <aqueous | minerals | solid\_solutions | gases | surfaces> \  
 <with | w/> <basis entry | string>

**Config** → **Show...**

Use the “show” command to display specific information about the current system or database. The options are:

show	show altered	show all
show basis	show commands	show couples
show directory	show elements	show gases
show minerals	show oxides	show show
show solid_solutions	show species	show surfaces
show system		

The command “show show” gives a list of show command options. When you type “show aqueous” (or “minerals”, “solid\_solutions”, “gases”, “oxides”, or “surfaces”), the program lists all entries of that type in the thermo database. The “solid\_solutions” option additionally includes solutions defined locally. A long form lets you limit the query to entries composed of a particular basis species or containing a text string in the name:

```
show aqueous with Al+++
show minerals w/ chal
```

There is also a compound form of the “show couples” command:

```
show coupling reactions
```

This command produces a complete list of the redox couples, in reaction form.

## 2.41 solid\_solution

```

solid_solution <name = <label>> <type> <discrete | continuous> \
  <<mineral> tag = <label>> <contains> <<mineral> tag = <label>> \
  <from <value>> <to <value>> <step = <value>> <<variable> = <value>>
solid_solution <solid solution> <options>
solid_solution remove <solid solution>

```

**Config** → **Solid Solutions...** →

**Config** → **Solid Solutions...** → *select entry(ies)* →

Use the “solid\_solution” command (abbrev.: “solid\_soln”) to define binary solid solutions in addition to any provided in the current thermo database. Supply a name with the “name” keyword, or simply set one as the first argument.

You can select the type of solid solution, either “ideal” (the default), “site\_mixing”, “guggenheim”, “regular”, “cubic”, or “third\_order”. For a site-mixing solution, set the site number “bsite”, which defaults to 1. For a Guggenheim solution, set dimensionless parameters “a0”, “a1”, and “a2” or a temperature expansion for their J/mol counterparts using coefficients “p1”-“p9”, each of which defaults to zero. For the other nonideal models, similarly, set coefficients “p1”, “p2”, etc., as appropriate. For more information, see **Solid solutions** under **Configuring the programs** in the **GWB Essentials Guide** and the **Thermo Datasets** chapter in the **GWB Reference Manual**. You can as well choose whether the solid solution is to be treated as continuous, the default option, or as a sequence of “discrete” minerals.

You need to specify two end members for a solid solution, each associated with an abbreviated “tag”. The tag cannot be the name of another entry in the database. You can set a composition range in terms of the mole fraction of the most recently referenced end member, using keywords “from” and “to”, as well as a compositional “step” separating tranches of a discrete solution; by default, composition ranges from zero to one and the step is one-twentieth of the range. Keyword “contains” may be inserted for clarity. Examples:

```

solid_solution name = my_ss Calcite contains Strontianite from 0% to 10% a0= 0.05
solid_solution my_ss Calcite Strontianite from 0 to .1 a0= 0.05
solid_solution Biotite site_mixing Annite Phlogopite bsite= 3

```

If you don’t specify a solid solution’s name, the app will create one automatically using the tags for the two end members, separated by a hyphen. If you don’t set a tag for an end member, the app will pick up the first six characters of the corresponding mineral’s name. For example, the command

```

solid_solution Calcite contains Strontianite

```

produces a solid solution named “Calcit-Stront”, whereas typing

```
solid_solution Calcite tag Cc contains Strontianite tag Str
```

yields the name “Cc-Str”. To redefine the properties of a solid solution, enter the name followed by the properties to be changed:

```
solid_solution my_ss a0 = 1 a1 = 2
```

(You may prefer to use the “alter” command to change solutions defined within the thermo database; the two commands are functionally equivalent.) The command

```
solid_solution remove my_ss
```

discards a previously defined solution.

## 2.42 sorbed\_mass

```
sorbed_mass <surface complex> = <value | ?>
```

**Basis** →  ▼ mol/dry g ▼

Use the “sorbed\_mass” command (abbrev.: “SM”) to set the mass, expressed in moles per gram of solid phase, of a surface complex from a  $K_d$  or Freundlich sorption model. Example:

```
sorbed_mass >Pb++ = 0.002
```

An argument of “?” unsets the value. Use the “activity” command to constrain aqueous species and surface complexes from an ion exchange model, and the “molality” command to set concentrations of surface complexes from a two-layer or Langmuir model. See also the “pH”, “Eh”, “pe”, “activity”, “molality”, “fugacity”, “xss”, “ratio”, and “surface\_data” commands.

## 2.43 stoich\_IS

```
stoich_IS = <value | ?>
```

**Basis** → stoichiometric  ▼ molal

Use the “stoich\_IS” command (also: “SI”, “SIS”) to set, in molal units, the stoichiometric ionic strength. Use the “?” argument to unset this value. Setting the ionic strength when a temperature has been specified and **Rxn** is invoked in “long” mode causes the

program to calculate the activity of water, if water appears in the reaction considered, and to use this value to simplify the equation. **Rxn** uses the setting directly when Helgeson's method is invoked in the dataset, and as a proxy for the sum of species concentrations in the Garrels and Christ (1965) and Wolery (1992b) methods. See also the "ionic\_strength" command.

### 2.44 suffix

```
suffix <string>
```

**Config** → **Output...**

Use the "suffix" command to alter the names of the output datasets ("Rxn\_output.txt") by adding a trailing string. Example:

```
suffix _run2
```

produces output datasets with names such as "Rxn\_output\_run2.txt".

### 2.45 surface\_capacitance

```
surface_capacitance = <value | ?>  
surface_capacitance on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant capacitance  ▼ F/m<sup>2</sup>  
or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: "surf\_capacitance") to set, in units of F/m<sup>2</sup>, the capacitance of a two-layer sorbing surface. When you set this value (or if a capacitance has been set in the surface reaction dataset), **Rxn** in long mode will calculate the relationship between surface charge and surface potential for the surface in question using the constant capacitance method, rather than the full two-layer model.

If you have set more than one sorbing surface (using the "surface\_data" command), you identify the surface in question by its "type". For example,

```
surface_capacitance on HFO = 2
```

The "type" associated with each surface is listed at the top of each dataset of surface reactions. The "type" of the hydrous ferric oxide surface represented by the dataset "FeOH.sdat", for example, is "HFO". You can use the "show" command to display the "type" of each active surface. See also the "ionic\_strength", "temperature", "surface\_charge", and "surface\_potential" commands.

## 2.46 surface\_charge

```
surface_charge = <value | ?>
surface_charge on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant charge  ▼ uC/cm2  
 or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_charge”) to set, in units of  $\mu\text{C}/\text{cm}^2$ , the charge density of a two-layer sorbing surface. When you set this value (or a value for surface potential) as well as a value for ionic strength, **Rxn** in long mode will calculate a value for the Boltzman factor and use it to simplify the equilibrium equation.

If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_charge on HFO = 0
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface. See also the “ionic\_strength”, “temperature”, “surface\_capacitance”, “surface\_data”, and “surface\_potential” commands.

## 2.47 surface\_data

```
surface_data <sorption dataset>
surface_data remove <sorption dataset | surface type>
surface_data OFF
```

**File** → **Open** → **Sorbing Surfaces...** →

**File** → **Open** → **Sorbing Surfaces...** → *select surface(s)* →

or **Config** → **Sorbing Surfaces...**

Use the “surface\_data” command (abbrev.: “surf\_data”) to specify an input dataset of surface sorption reactions to be considered in the calculation. The dataset name should be enclosed in quotes if it contains any unusual characters. Use the “remove” argument to eliminate a surface dataset, specified by name or surface type (e.g., “HFO”), from consideration. The argument “OFF” disables consideration of all surface complexes.

You can specify more than one sorbing surface in a model by repeating the “surface\_data” command for different datasets (a two-layer dataset of surface reactions for sorption onto hydrous ferric oxide, several triple-layer and CD-MUSIC datasets,

## GWB Command Reference

---

as well as example datasets for the ion exchange,  $K_d$ , Freundlich, and Langmuir models are distributed with the software). To remove a dataset of surface reactions from consideration, you use commands such as

```
surface_data remove FeOH.sdat
surface_data remove HFO
surface_data OFF
```

The latter command removes all of the surface datasets that have been loaded.

### 2.48 surface\_potential

```
surface_potential = <value | ?>
surface_potential on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant potential  ▼ mV  
**or File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_potential”) to set, in units of mV, the electrical potential for a two-layer sorbing surface. When you set this value and a temperature, **Rxn** in long mode will calculate a value for the Boltzman factor and use it to simplify the equilibrium equation. Alternatively, when you provide values for surface charge and ionic strength, the program in long mode will figure surface potential and use it to compute the Boltzman factor.


If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_potential on HFO = -10
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface. See also the “ionic\_strength”, “temperature”, “surface\_charge”, and “surface\_data” commands.

### 2.49 swap

```
swap <new basis> <for> <basis species>
```

**Basis** → 

Use the “swap” command to change the set of basis entries. All reactions are written internally in terms of an initial list of basis species. The basis can be altered over the

course of the calculation to reflect the species, minerals, and gases to be included in the reaction. To include an aqueous species, mineral, gas, or activity ratio in the calculation, swap it into the basis in place of a basis species from the original list, as carried in the database. Examples:

```
swap CO3--      for HCO3-
swap Quartz     for SiO2(aq)
swap CO2(g)     for H+
swap O2(g)      for O2(aq)
swap Ca++/Na+^2 for Ca++
```

An end member of a continuous solid solution can similarly be swapped into the basis. Use the syntax <solid solution>:<end member tag>. For example, the sodium end member Albite (tag "Ab") of the plagioclase feldspar solid solution (name "Plag") can be swapped into the basis simultaneously with the calcium end member Anorthite (tag "An") as follows:

```
swap Plag::Ab for Na+
swap Plag::An for Al+++
```

The new species must contain in its composition the original basis species being swapped out (you can't swap lead for gold). For example, CO<sub>2</sub>(g) is composed of HCO<sub>3</sub><sup>-</sup>, H<sup>+</sup>, and water in "thermo.tdat". The reactions in the thermo dataset (once reduced to the set of basis and redox species and modified to reflect enabled redox couples) show the basis entries for which a species may be swapped. For a list of original basis species, type "show basis". To reverse a swap, type "unswap <species>".

## 2.50 system

```
system <system command>
```

Use the "system" command to execute a command on the computer's operating system (a "DOS command") without leaving **Rxn**. Example:

```
system "copy Rxn_output.txt Rxn_output2.txt"
```

Enclose multiword system commands in quotes. An alternative way to fork a system command is to begin the **Rxn** command with a "\$":

```
$copy Rxn_output.txt Rxn_output2.txt
```

In this case, no quotation marks are needed.

When a "system" command is executed, a "Command Prompt" window will appear briefly on your screen. Due to limitations of the Windows operating system, you cannot

fork a command that requires user input, and you will not be able to see any output (including error messages) that might be generated by the command.

## 2.51 temperature

**temperature** = <value | ?> <unit>

**Basis** → temperature  ▼ C ▼

Use the “temperature” command (also: “T”) to set temperature in °C, °F, or K. Examples:

```
temperature 25 C
temperature 300 F
T = 298 K
```

Temperature values can range over the span of the thermo dataset, from 0°C to 300°C for “thermo.tdat”; centigrade is the default scale. Use “?” to unset the temperature:

```
T = ?
```

## 2.52 unalter

**unalter** <species | mineral | solid solution | gas | surface species | **ALL**>

**Config** → **Alter Log Ks...** → *select entry(ies)* →

Use the “unalter” command to reverse the effect of having changed the log  $K$ ’s for a species, mineral, gas, or surface reaction, the selectivity coefficient for an exchange species, the  $K_d$  for a sorbed species, the  $K_f$  and  $n_f$  for a Freundlich species, or the properties of a solid solution in the current dataset. Example:


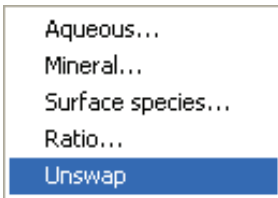
```
unalter Quartz
```

In this case, the log  $K$  values for quartz revert to those in the current thermodynamic dataset. The argument “ALL” resets the log  $K$ ’s, selectivity coefficients,  $K_d$ ’s, or  $K_f$ ’s and  $n_f$ ’s for all species, minerals, gases, and surface species, as well as the properties of all solid solutions in the dataset.



## 2.53 unswap

**unswap** <species | ALL>

**Basis** →  → 

- Aqueous...
- Mineral...
- Surface species...
- Ratio...
- Unswap**

Use the “unswap” command to reverse a basis swap. Example:

```
unswap Quartz (or unswap SiO2(aq))
```

to reverse the effect of the previous command

```
swap Quartz for SiO2(aq)
```

At this point, SiO<sub>2</sub>(aq) is back in the basis. The “ALL” argument reverses all basis swaps.

## 2.54 usgovt

**usgovt**

**Help** → **About Rxn...**

Display the restricted rights legend that governs use, duplication, or disclosure by the U.S. Government.

## 2.55 xss

**xss** <end member> = <value | ?>

**Basis** →  ▼ mole fraction ▼

Use the “xss” command (abbrev.: “X”) to constrain the mole fraction of an end member in a continuous solid solution. Example:

```
xss Plag::Ab = 0.3
```

## GWB Command Reference

---

If the paired end member is in the basis, its mole fraction will be set such that the total sums to one. Setting an end member's mole fraction will clear out any activity setting for end members. Use "?" to unset a mole fraction value:

```
xss Plag::Ab = ?
```

See also the "activity", "pH", "Eh", "pe", "molality", "fugacity", "sorbed\_mass", and "ratio" commands.

# Act2 Commands

---

## 3.1 activity

**activity** <species> = <value | ?>

**Basis** →  ▼ activity ▼

Use the “activity” command (abbrev.: “a”) to constrain the activity of an aqueous species or water over the diagram. Examples:

```
activity Na+ = 0.3
activity main = 10^-5
a H2O = 0.9
log a H+ = -5
```

The label “main” refers to the species being diagrammed. Use “?” to unset an activity value. See also the “pH”, “Eh”, “pe”, “fugacity”, and “ratio” commands.

## 3.2 add

**add** <basis species>

**Basis** →

Use the “add” command to include a basis species in the calculation. Example:

```
add HCO3-
```

See also the “swap”, “activity”, “fugacity”, “ratio”, “pH”, “pe”, and “Eh” commands.

### 3.3 alter

```

alter <species | mineral | gas> <log Ks>
alter <species | mineral | gas> <poly coefs> TminK = <value> TmaxK = <value>
alter <solid solution> <type> \
  <from <value> to <value> step <value>> <<variable> = <value>>

Config → Alter Log Ks...
Config → Solid Solutions...

```

Use the “alter” command to change the temperature expansion for a species’, mineral’s, or gas’ log  $K$  or to change the properties of a solid solution.

Temperature expansions for log  $K$ s are given in the thermo dataset either as  $T$ -tables, or by up to six coefficients of a polynomial. When the current thermo dataset uses  $T$ -tables, you list replacement values at each of the principal temperatures specified in the dataset, most commonly 0°C, 25°C, 60°C, 100°C, 150°C, 200°C, 250°C, and 300°C. Example:

```
alter Anhydrite -4.3009 -4.4199 -4.7126 -5.1758 -6.2299 500 500 500
```

Values of “500” represent a lack of data at the corresponding temperature.

For a thermo dataset constructed from polynomial expansions, on the other hand, list instead up to six polynomial coefficients

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4
```

You can optionally append a temperature range of validity, in Kelvins, for the polynomial:

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4 TminK= 293 TmaxK= 383
```

Absent a range, the polynomial is taken to span the principal temperatures. You may adjust a reaction’s temperature range without specifying polynomial coefficients, but setting only coefficients resets the temperature range.

For a solid solution, use the “alter” command to set the activity coefficient model, the composition range in terms of the mole fraction of the most recently specified end member, and parameters for the chosen activity model, as described for the “solid\_solution” command. A solid solution “my\_ss” might be set as a subregular Guggenheim solution with the command

```
alter my_ss guggenheim a0 = 1 a1 = 2
```

You may prefer to use the “alter” command to adjust an entry from the thermo database, and the “solid\_solution” command to reconfigure a user-defined solution, but the two commands work interchangeably for this purpose.

Type “show alter” to list altered species or solutions and their settings; the “unalter” command reverses the process.

### 3.4 background

**background** <color name>

**Format** → **Appearance...** → **Plot Area**

Use the “background” command to set the background color of the plot. Example:

```
background "sky blue"
```

Type “show lines” for current color settings; “show colors” for color choices. Enclose multiword colors in quotes.

### 3.5 banner

**banner** <off>

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Title**

The “banner” command controls whether **Act2** will plot a banner along the side of the diagram summarizing the calculation constraints. The optional argument “off” de-selects the option. Type “show options” to display the current option setting.

### 3.6 chdir

**chdir** <directory> <ALL>

**File** → **Working Directory...**

Use the “chdir” command (abbrev.: “work\_dir”, “cd”) to change the working directory. The program reads input scripts relative to the current working directory and writes output into it. Typing the command “chdir” without an argument causes the program to display the name of the working directory. The “ALL” argument changes the working directories of all of the GWB apps. The command

```
chdir ~
```

changes to the user’s home directory, if one is defined by the operating system.

### 3.7 clear

**clear**

**View** → **Clear Screen**

Use the “clear” command (abbrev.: “c”) or right-click in the text window and choose **Clear Screen** to clear it.

### 3.8 clipboard

**clipboard**  
**clipboard bitmap**

**Edit** → **Copy**

**Edit** → **Copy As** → **Bitmap**

The “clipboard” command causes the contents of the **Plot** pane to be copied to the clipboard, from where it can be pasted into various types of documents. The “bitmap” keyword causes the plot to be copied as a bitmap, rather than a vector graphic.

### 3.9 color\_by\_sample

**color\_by\_sample** <off>

**Format** → **Scatter Data...**

The “color\_by\_sample” command causes **Act2** to show each scatter data sample label in the color of its related marker. The optional argument “off” causes all of the sample labels to be shown in the color set in **Format** → **Scatter Data...**

### 3.10 couple

**couple** <redox species | element(s) | **ALL**>

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “couple” command to enable any redox coupling reactions that have been disabled with the “decouple” command. You specify one or more redox species or elements. For example, the command

`couple Carbon`

couples all redox reactions involving the element carbon. Argument “ALL” enables all of the coupling reactions in the thermo dataset.

### 3.11 custom

```
custom <color number> <name> <red green blue>
```

**Color** → **Define Custom Colors**

The “custom” command allows you to change the name and/or RGB values of the eleven custom colors. Specify the custom color number (1 - 11), the name in quotes, and the RGB values (0 - 255). Examples:

```
custom 2 "Med Blue" 128 128 255
custom 2 "Med Blue"
custom 2 128 128 255
```

### 3.12 data

```
data <thermo dataset> <verify>
```

**File** → **Open** → **Thermo Data...**

Use the “data” command to change the input file of thermodynamic data. Example:

```
data "..\my_thermo.tdat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. The “verify” option causes the program to read the named dataset only if it has not already been read.

### 3.13 decouple

```
decouple <redox species | element(s) | ALL>
```

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “decouple” command to disable the coupling reactions for one or more redox species, in order to calculate a model assuming redox disequilibrium. The redox species then become available for use as basis species and may be constrained independently of the original basis entries. You can disable as many coupling reactions as you want.

You specify either one or more redox species or elements. For example, the command

decouple Carbon

decouples all redox reactions involving the element carbon. Argument “ALL” disables all of the coupling reactions in the thermo dataset. Use the “couple” command to enable coupling reactions, once they have been disabled.

### 3.14 diagram

**diagram** <main species> <on axis> <vs axis> <activity = value>

Use the “diagram” command (also: “plot”) to set the species to be diagrammed, its activity, and the diagram axes. Examples:

diagram Fe++ on Eh vs pH activity = 1e-4  
 diagram U++++ log activity = -10  
 diagram on pe vs H+^-1  
 diagram x= Cl- y= H+^2/Ca++

The keywords “x” and “y” may be substituted for “on” and “vs”. The species to be diagrammed, as well as the species appearing on the axes, must have been swapped into the basis if they do not appear in the original basis.

### 3.15 Eh

**Eh** = <value in volts | ?>

**Basis** →  ▼ Eh ▼



Use the “Eh” command to set Eh over the diagram. Example:

```
Eh = 1/2
```

sets the system’s oxidation state to correspond to an Eh of 0.5 volts. Use “?” to unset an Eh value. See also the “activity”, “pH”, “pe”, “fugacity”, and “ratio” commands.

### 3.16 explain

```
explain <species | mineral(s) | solid solution(s) | gas(es)>
```

**Config** → **Show...** → *select species*

Use the “explain” command to get more information (such as the mole weight of a species, a mineral’s formula, mole volume, and density, or a solid solution’s end-member minerals) about species, minerals, solid solutions, and gases in the dataset. Example:

```
explain Analcime
```

### 3.17 extrapolate

```
extrapolate <on | off>
```

**Config** → **Options...**

Use the “extrapolate” option to cause the program to extrapolate log  $K$ s for reactions forming aqueous species, minerals, and gases to temperatures beyond the data’s known range of validity. The option should be used with considerable care.

The temperature range of validity for a species’ log  $K$  is defined by its temperature expansion, taken from the thermo dataset. For  $T$ -table expansions, it is the span of principal temperatures at which log  $K$ s are not 500, whereas for polynomial expansions, the range is listed explicitly.

In normal operation, the program loads only species with ranges of validity encompassing the calculation’s temperature range. When the “extrapolate” option is “on”, in contrast, the program projects to the temperature of interest the log  $K$ s of species that would not otherwise be loaded.

### 3.18 font

```
font <font name>
```

Use the “font” command to select the font to be used to label the diagram. Example:

```
font "courier new"
```

You may select any font name available in the **Choose Font...** dialog. Enclose multiple word names in quotes. To change the font for individual aspects of the diagram, right-click on the aspect and choose **Font** →.

### 3.19 fugacity

```
fugacity <gas> = <value | ?>
```

```
Basis →  ▼ fugacity ▼
```

Use the “fugacity” command (abbrev.: “f”) to set gas fugacities (on an atm scale) over the diagram. Examples:

```
fugacity O2(g) = .2  
f CO2(g) = 0.0003  
log f S2(g) = -30
```

Use “?” to unset a fugacity value: See also the “activity”, “pH”, “Eh”, “pe”, and “ratio” commands.

### 3.20 gases

```
gases <off>
```

```
Format → Quick Toggle...
```

```
Format → Appearance... → Diagram
```

The “gases” command controls whether **Act2** will consider gaseous species when it calculates the full diagram. By default, the program includes any gases that contain the species being diagrammed and that can be made up from the basis entries; it assigns each a fugacity equal to the total pressure. Use this command with the “off” argument to cause gases to be excluded from the calculation. See also the “pressure” command.

### 3.21 go

**go**

**Run → Go**

**Plot**

Use the “go” command to initiate calculation of a diagram based on the current chemical system.

### 3.22 grep

**grep <string>**

**File → View → .\Act2\_output.txt → Edit → Find**

Use the “grep” command to search for each occurrence of the given character string in the current output dataset. For example, use

grep Albite

to obtain a list of each line in “Act2\_output.txt” containing the string “Albite”.

### 3.23 gridlines

**gridlines <off>**

**Format → Quick Toggle...**

**Format → Appearance... → Axes and Ticks**

The “gridlines” command controls whether **Act2** will draw gridlines on the diagram. The optional argument “off” de-selects the option.

### 3.24 help

**help <help | subject>**

**Help**

Use the “help” command (also: “man”) to display online documentation for the GWB package. The command “help help” shows a list of the types of documentation available.

### 3.25 history

history

Run → History...

Use the “history” command to view a list of previously executed commands. You can re-execute commands in the **History...** dialog box by double-clicking on a command, selecting one or more commands, and then choosing **Execute**, or with a history substitution command, as explained in the **Command Line Interface** chapter in the **GWB Reference Manual**.

### 3.26 labels

labels <off>

Format → Quick Toggle...

Format → Appearance... → Diagram

The “labels” command controls whether **Act2** will label the diagram with the names of predominant aqueous species and stable minerals and gases. Other types of lettering on the plot, such as axis labels, may be eliminated by using the “line” command to omit the aspect. **Act2**, by default, includes species labels on the diagram. The optional argument “off” de-selects the option. Type “show options” to display the current option setting.

### 3.27 legend

legend <off>

legend x <value | ?> y <value | ?>

Format → Quick Toggle...

Format → Scatter Data...

Use the “legend” command to control whether a legend related to scatter data is shown. The legend is only available when scatter data is currently loaded. Keyword “off” hides the legend. Keywords “x” and “y” set the position of the legend. Examples:

```
legend
legend off
legend x 2 y 7
```

### 3.28 line

```

line <aspect> <pen thickness> <color> <size = character size>
      <font = <noitalic | italic>> <font = <nobold | bold>>
      <noomit | include | omit>
line <aspect> <index> <pen thickness> <color> <size = character size>
      <symbol = symbol> <noomit | include | omit>

```

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Diagram**

**Format** → **Appearance...** → **Axes and Ticks**

**Format** → **Appearance...** → **Plot Area**

**Format** → **Appearance...** → **Title**

**Format** → **Scatter Data...**

**Format** → **Trace...**

Use the “line” command to change the boldness and color of lines used to construct the diagram, set the point size for lettering, set bold and/or italic lettering, or omit an aspect of the lettering. Examples:

```

line axes fine cyan
line axis_labels size = 18
line axis_labels font = bold
line axis_labels omit

```

The command also sets the boldness and color of reaction traces and the symbol, color, and point size of trace markers. If multiple reaction traces are loaded and the “trace appearance individual” option has been enabled (see the “trace” command), set the appearance of the trace, markers, marker\_start, or marker\_end aspects for a specific dataset by specifying its index immediately after the aspect argument. Of the currently loaded datasets, that loaded first has an index of 1, the second an index of 2, and so on. Examples:

```

line trace 1 "Grey (90%)" xx-bold
line trace 2 "Grey (60%)" med-bold
line marker_start 1 noomit symbol blot

```

Set the lettering and marker size in points (values  $\geq 1$ ) or inches (values  $< 1$ ). Type “show lines” for aspects that can be changed and their current settings. Type “show pens”, “show colors”, and “show symbols” for available pens, colors, and symbols.

### 3.29 log

```
log <variable> = <value>
```

Use the “log” command to set variables on a logarithmic scale. Examples:

```
log fugacity O2(g) = -65  
log activity U++++ = -10  
diagram Cu+ log activity = -8
```

### 3.30 minerals

```
minerals <off>
```

```
Format → Quick Toggle...
```

```
Format → Appearance... → Diagram
```

The “minerals” command controls whether **Act2** will consider mineral species when it calculates the full diagram. By default, the program includes each mineral that contains the species being diagrammed and that can be made up from the basis entries. Use this command with the “off” argument to cause minerals to be excluded from the calculation.

### 3.31 mosaic\_bounds

```
mosaic_bounds <off>
```

```
Format → Quick Toggle...
```

```
Format → Appearance... → Diagram
```

When calculating a mosaic diagram (see the “speciate” command), **Act2** normally draws the boundaries among sub-diagrams using a finely dashed line. The “mosaic\_bounds” command controls whether these lines are drawn. **Act2**, by default, draws the mosaic boundaries. The optional argument “off” de-selects the option. Type “show options” to display the current option setting.

### 3.32 mosaic\_labels

**mosaic\_labels** <off>

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Diagram**

The “mosaic\_labels” command controls whether **Act2** labels the subdiagrams in a mosaic diagram (see the “speciate” command). The program, by default, does not label these fields. The command “mosaic\_labels” causes the labels to be drawn; the optional argument “off” de-selects the option. Type “show options” to display the current option setting.

### 3.33 paper

**paper height** = <value | ?> **width** = <value | ?> \  
**factor** = <value | ?> \  
**orientation** = <portrait | landscape | ?> \  
**units** = <in | cm | ?> **scale** = <on | off | ?>  
**paper reset**

**File** → **Page Setup...**

The “paper” command controls printing from **Act2**, including the paper size and diagram orientation. Keywords “height” and “width” specify paper dimensions, “factor” is an optional scaling factor applied to the plot upon printing, “orientation” controls whether the plot prints as portrait or landscape, “units” tells whether the page dimensions are in inches or centimeters, and “scale” controls whether the plot is scaled automatically to the size of the printed page. Keywords “selection” and “paper” are for internal use by the program.

Type

show paper

to see the current settings, and

paper reset

to restore settings to their default values.

### 3.34 pause

pause

Use the “pause” command to cause **Act2** to pause temporarily during input. This command is useful when you are debugging scripts.

### 3.35 pe

**pe** = <value | ?>

**Basis** →  ▼ pe ▼

Use the “pe” command to set oxidation state over the diagram in terms of pe. Example:

pe = 10

is equivalent to

log activity e- = -10

where “e-” is the electron. Use “?” to unset a pe value: See also the “activity”, “Eh”, “pH”, “fugacity”, and “ratio” commands.

### 3.36 pH

**pH** = <value | ?>

**Basis** →  ▼ pH ▼

Use the “pH” command to set pH over the diagram. Example:

pH = 5

is equivalent to

log activity H+ = -5

Use “?” to unset a pH value: See also the “activity”, “Eh”, “pe”, “fugacity”, and “ratio” commands.

### 3.37 polymorphs

**polymorphs** <mineral>

**Config** → **Show...** → Minerals → *select mineral*



Use the “polymorphs” command to list each polymorph (mineral identical in composition to the specified mineral) in the current database. Example:

```
polymorphs Albite
```

### 3.38 pressure

```
pressure = <value> <unit>
```

```
Basis → pressure →  ▼ bars ▼
```

Use the “pressure” command (also: “P”) to set total pressure. Pressure affects the position of water stability limits and the fugacities assigned to gaseous species, but not mineral or species stability. Examples:

```
pressure 20 atm
pressure 1.01 bar
P 0.1 MPa
```

The default setting is the temperature expansion for pressure given at the head of the thermo dataset in use. In many datasets, pressure is set to 1 atm below 100°C and the vapor pressure of water above this temperature. The default unit for the “pressure” command is bar.

### 3.39 print

```
print <off | on>
```

```
Config → Output...
```

The command “print” (also: “printout”) with the argument “off” causes **Act2** to bypass writing calculation results to the “Act2\_output.txt” dataset. By default, the program writes to this dataset. The command “print on” (or just “print”) re-enables the output.

### 3.40 pwd

```
pwd
```

```
File → Working Directory...
```

The “pwd” command returns the name of the current working directory. The command has the same effect as typing “show directory”. See the “chdir” command.

### 3.41 quit

**quit**

**File** → **Quit**

Use the “quit” command (also: “end”, “q”) to exit the program. Touching **Ctrl+Q** serves the same purpose as the “quit” command.

### 3.42 ratio

**ratio** <act./fug. ratio> = <value | ?>

**Basis** →  ▼ ratio ▼

Use the “ratio” command to constrain an activity or fugacity ratios over the diagram. Example:

```
swap Ca++/Na+^2 for Ca++
ratio Ca++/Na+^2 = 0.2
```

Use “?” to unset a ratio value: See also the “activity”, “pH”, “Eh”, “pe”, and “fugacity” commands.

### 3.43 read

**read** <dataset>

**File** → **Open** → **Read Script...**

Use the “read” command to begin reading commands from a script stored in a dataset. Example:

```
read Cu_solubility
```

Control returns to the user after the script has been read, unless the script contains a “quit” command. You can also use the “read” command in place of the “data” or “surface\_data” command to read a thermo or surface reaction dataset.

When typing a “read” command, you can use the spelling completion feature to complete dataset names: touch “[tab]” or “[esc]” to cycle through the possible completions, or **Ctrl+D** to list possible completions.

### 3.44 remove

```
remove <basis specie(s)> <solid solution(s)>
remove basis <basis specie(s)>
remove solid_solution <solid solution(s)>
```

**Basis** or  
**Config** → **Solid Solutions...** →  
*select entry(ies)* →

Use the “remove” command (also: “rm”) to eliminate one or more basis entries from consideration in the calculation. Example:

```
remove Na+
remove Quartz Calcite
```

Components can be reentered into the basis using the “swap”, “add”, “activity”, and “fugacity” commands. You can also use the “remove” command to remove solid solutions configured in a run (but not those set in the thermo dataset):

```
remove solid_solution mySS
```

### 3.45 reset

```
reset
```

**File** → **Reset Configuration**

Use the “reset” command to begin defining the chemical system again with a clean slate. Your current settings will be lost, and all options will be returned to their default states. The command, however, does not alter the setting for the thermo dataset.

### 3.46 resume

```
resume
```

**File** → **Resume**

Use the “resume” command to restore the configuration from when you last exited the program. The program configuration is stored in a file “act2\_resume.ac2” in your profile directory (e.g., “c:\Documents and Settings\jones\Application Data\GWB”). Your current configuration is discarded.

### 3.47 save

```
save <dataset> <hex>
```

**File** → **Save As...**

Use the “save” command to write the current chemical system into a dataset in **Act2** format commands. The dataset can be used as an **Act2** input script. Examples:

```
save  
save kspar.ac2
```

If you don't specify a dataset name, **Act2** will ask for one. The optional keyword “hex” causes the program to output numbers as hexadecimal values.

### 3.48 scatter

```
scatter <dataset>
```

**File** → **Open** → **Scatter Data...** or **Format** → **Scatter Data...**

Use the “scatter” command to project data points onto the diagram. Data points are read from the specified dataset; “scatter off” turns off the option. The preferred way to add scatter data is to import it from a **GSS** spreadsheet (“.gss” file, see the **Using GSS** and **Using Act2** chapters of the **GWB Essentials Guide**). The old method of importing a specially formatted table from a text file still works, however (see the **Scatter Data** chapter in the **GWB Reference Manual**).

### 3.49 script

```
script  
script end
```

Use the “script” command to mark the beginning, and optionally the end, of a control script. Control scripts differ from standard input files in that they can contain not only **Act2** commands, but control structures such as loops and if-else branches. Control scripts follow the Tcl syntax, described in [www.tcl.tk](http://www.tcl.tk) and [mini.net/tcl](http://mini.net/tcl), as well as several widely available textbooks.

Within a control script, filenames are written with double rather than single backslashes. For example, a “read” command might appear as

```
read GWB_files\\My_file.ac2
```

within a control script.

### 3.50 show

```
show <option>
show <aqueous | minerals | solid_solutions | gases> \
  <with | w/> <basis entry | string>

Config → Show...
```

Use the “show” command to display specific information about the current system or database. The options are:

show	show all	show altered
show basis	show colors	show commands
show couples	show directory	show elements
show gases	show lines	show minerals
show options	show oxides	show paper
show pens	show species	show solid_solutions
show suppressed	show symbols	show system
show show		

The command “show show” gives a list of show command options. When you type “show aqueous” (or “minerals”, “solid\_solutions”, or “gases”), the program lists all entries of that type in the thermo database. The “solid\_solutions” option additionally includes solutions defined locally. A long form lets you limit the query to entries composed of a particular basis species or containing a text string in the name:

```
show aqueous with Al+++
show minerals w/ chal
```

There is also a compound form of the “show couples” command:

```
show coupling reactions
```

This command produces a complete list of the redox couples, in reaction form.

### 3.51 solid\_solution

```

solid_solution <name = <label>> <type> \
  <<mineral> tag = <label>> <contains> <<mineral> tag = <label>> \
  <from <value>> <to <value>> <step = <value>> <<variable> = <value>>
solid_solution <solid solution> <options>
solid_solution remove <solid solution>

```

**Config** → **Solid Solutions...** →

**Config** → **Solid Solutions...** → *select entry(ies)* →

Use the “solid\_solution” command (abbrev.: “solid\_soln”) to define binary solid solutions in addition to any provided in the current thermo database. Supply a name with the “name” keyword, or simply set one as the first argument.

You can select the type of solid solution, either “ideal” (the default), “site\_mixing”, “guggenheim”, “regular”, “cubic”, or “third\_order”. For a site-mixing solution, set the site number “bsite”, which defaults to 1. For a Guggenheim solution, set dimensionless parameters “a0”, “a1”, and “a2” or a temperature expansion for their J/mol counterparts using coefficients “p1”-“p9”, each of which defaults to zero. For the other nonideal models, similarly, set coefficients “p1”, “p2”, etc., as appropriate. For more information, see **Solid solutions** under **Configuring the programs** in the **GWB Essentials Guide** and the **Thermo Datasets** chapter in the **GWB Reference Manual**.

You need to specify two end members for a solid solution, each associated with an abbreviated “tag”. The tag cannot be the name of another entry in the database. You can set a composition range in terms of the mole fraction of the most recently referenced end member, using keywords “from” and “to”, as well as a compositional “step” separating tranches of a discrete solution; by default, composition ranges from zero to one and the step is one-twentieth of the range. Keyword “contains” may be inserted for clarity. Examples:

```

solid_solution name = my_ss Calcite contains Strontianite from 0% to 10% a0= 0.05
solid_solution my_ss Calcite Strontianite from 0 to .1 a0= 0.05
solid_solution Biotite site_mixing Annite Phlogopite bsite= 3

```

If you don’t specify a solid solution’s name, the app will create one automatically using the tags for the two end members, separated by a hyphen. If you don’t set a tag for an end member, the app will pick up the first six characters of the corresponding mineral’s name. For example, the command

```
solid_solution Calcite contains Strontianite
```

produces a solid solution named “Calcit-Stront”, whereas typing

```
solid_solution Calcite tag Cc contains Strontianite tag Str
```

yields the name “Cc-Str”. To redefine the properties of a solid solution, enter the name followed by the properties to be changed:

```
solid_solution my_ss a0 = 1 a1 = 2
```

(You may prefer to use the “alter” command to change solutions defined within the thermo database; the two commands are functionally equivalent.) The command

```
solid_solution remove my_ss
```

discards a previously defined solution.

### 3.52 span

```
span <value | ?> <C> <to> <value | ?> <C>
span <off | on | reset>
```

**Config → Options...**

The “span” command lets you set a temperature range across which stability must be defined for a species to be considered in the calculation.

For example, you may wish to compare results calculated at 25°C with those figured at 30°C. Since  $\log K$  for a number of species in the thermo dataset is given at only 25°C, unless the “extrap” option is set, the former calculation may well take into account a somewhat larger set of species than the latter. Entering the command

```
span 25 C to 30 C
```

assures the same set of species is loaded in each case.

The lower limit may be identified with keyword “from” or “Tmin”, and the upper denoted by “to” or “Tmax”. A value of “?” unsets the lower or upper limit. Keyword “off” disables the feature, leaving values for the limits intact, and keyword “on” re-enables the feature. The “reset” keyword clears the limits, disabling the feature. By default, the span feature is disabled.

### 3.53 speciate

**speciate** <basis species> <over> <X | Y | X-Y>

Use the “speciate” command to cause a basis species of the chemical system to react with the axis species to form complex ions and minerals. Example:  $\text{SO}_4^{--}$  can speciate on an Eh-pH diagram to form  $\text{HSO}_4^-$ ,  $\text{H}_2\text{S}(\text{aq})$ ,  $\text{HS}^-$ , and native sulfur. The program will calculate subdiagrams within each area of predominance resulting from the speciation and assemble the subdiagrams into a complete diagram.

Examples:

```
speciate SO4--
speciate HCO3- over X
speciate NO3- over X-Y
```

Each basis species can speciate over both axes, the default if the speciate command is given, or over only the  $x$  or the  $y$  axis. The “unspeciate” command reverses the process.

### 3.54 suffix

**suffix** <string>

**Config** → **Output...**

Use the “suffix” command to alter the names of the output datasets (“Act2\_output.txt”) by adding a trailing string. Example:

```
suffix _run2
```

produces output datasets with names such as “Act2\_output\_run2.txt”.



### 3.55 suppress

**suppress** <species, minerals, solid solutions, gases | **ALL**>

**Config** → **Suppress...**

Use the “suppress” command (also: “kill”) to prevent certain aqueous species, minerals, solid solutions, or gases from being considered in a calculation. Example:

```
suppress H3SiO4- Quartz "Maximum Microcline"
```

prevents the three entries listed from being loaded from the database. Typing “suppress ALL” suppresses all of the minerals and solid solutions in the thermodynamic database, as well as any solid solutions defined locally. The “unsuppress” command reverses the process. To suppress all but a few minerals, you could type

```
suppress ALL
unsuppress Quartz Muscovite Kaolinite
```

### 3.56 swap

**swap** <new basis> <for> <basis species>

**Basis** → ↺

Use the “swap” command to change the set of basis entries. All reactions are written internally in terms of an initial list of basis species. The basis can be altered over the course of the calculation to reflect the chemical constraints to be imposed. An aqueous species, mineral, gas, or activity ratio can be swapped into the basis in place of one of the original basis species listed in the database. Examples:

```
swap CO3--      for HCO3-
swap Quartz     for SiO2(aq)
swap CO2(g)     for H+
swap O2(g)      for O2(aq)
swap Ca++/Na+^2 for Ca++
```

The new species must contain in its composition the original basis species being swapped out (you can’t swap lead for gold). For example,  $\text{CO}_2(\text{g})$  is composed of  $\text{HCO}_3^-$ ,  $\text{H}^+$ , and water in “thermo.tdat”. The reactions in the thermo dataset (once reduced to the set of basis and redox species and modified to reflect enabled redox couples) show the basis entries for which a species may be swapped. For a list of original basis species, type “show basis”. To reverse a swap, type “unswap <species>”.

### 3.57 system

```
system <system command>
```

Use the “system” command to execute a command on the computer’s operating system (a “DOS command”) without leaving **Act2**. Example:

```
system "copy Act2_output.txt Act2_output2.txt"
```

Enclose multiword system commands in quotes. An alternative way to fork a system command is to begin the **Act2** command with a “\$”:

```
$copy Act2_output.txt Act2_output2.txt
```

In this case, no quotation marks are needed.

When a “system” command is executed, a “Command Prompt” window will appear briefly on your screen. Due to limitations of the Windows operating system, you cannot fork a command that requires user input, and you will not be able to see any output (including error messages) that might be generated by the command.

### 3.58 temperature

```
temperature = <value> <unit>
```

```
Basis → temperature  ▼ C ▼
```

Use the “temperature” command (also: “T”) to set temperature in °C, °F, or K. Examples:

```
temperature 25 C  
temperature 300 F  
T = 298 K
```

Temperature values can range over the span of the thermo dataset, from 0°C to 300°C for “thermo.tdat”; 25°C is the default; centigrade is the default scale. Setting the temperature automatically sets pressure (see the “pressure” command).

### 3.59 title

```
title <character string | empty>
title x <value | ?> y <value | ?>
```

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Title**

Use the “title” command to set a title to be written above the plot. Keyword “empty” eliminates the title. Keywords “x” and “y” set the position of the title. Examples:

```
title "Al solubility vs pH"
title x 2 y 7
```

Enclose multiword titles in quotes.

### 3.60 trace

```
trace <dataset>
trace appearance <global | individual>
```

**File** → **Open** → **Reaction Trace...** or **Format** → **Trace...**

**File** → **Update Trace**

Use the “trace” command to project the results of a **React** run onto the diagram. The **React** results are read from a “React\_plot.gtp” dataset in the working directory or from a specified dataset. Repeat the “trace” command to specify additional datasets. Examples:

```
trace
trace ..\React_plot_run1.gtp
trace ..\React_plot_run2.gtp
```

Delete a trace by including the “off” keyword at the end of the “trace” command. Examples:

```
trace ..\React_plot_run1.gtp off
trace off
```

The first command removes the specified dataset and the second turns off the projection option entirely.

Use the “appearance individual” keywords to allow details of a trace’s appearance (e.g., line thickness and color) to be set separately for each dataset. The “appearance global” keywords revert the option to the default setting, in which the appearance applies to all loaded reaction traces. The “line” command sets the appearance.

### 3.61 truncate

**truncate** <off>

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Diagram**

The truncate option controls whether the diagram is truncated at the stability limits of water, whenever the limits are displayed, or extends across the axis ranges. By default, the diagram is truncated at the water limits, but the command “truncate off” extends the diagram over the full plot.

### 3.62 unalter

**unalter** <species | mineral | solid solution | gas | **ALL**>

**Config** → **Alter Log Ks...** → *select entry(ies)* → **delete**

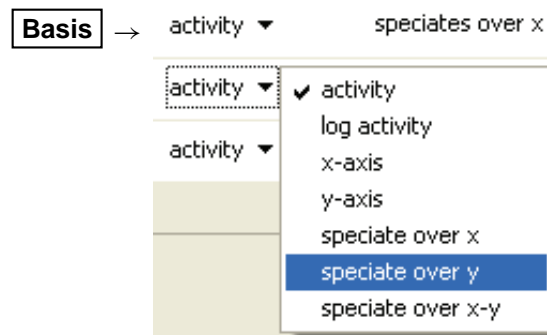
Use the “unalter” command to reverse the effect of having changed the log  $K$ ’s for a species, mineral, or gas, or the properties of a solid solution in the current dataset. Example:

unalter Quartz

In this case, the log  $K$  values for quartz revert to those in the current thermodynamic dataset. The argument “ALL” resets the log  $K$ ’s for all species, minerals, and gases, as well as the properties of all solid solutions in the dataset.

### 3.63 unspciate

**unspciate** <basis species>



Use the “unspeciate” command to reverse the effects of the “speciate” command.  
Example:

```
unspeciate SO4--
```

### 3.64 unsuppress

```
unsuppress <species, minerals, solid solutions, gases | ALL>
```

**Config** → **Suppress...**

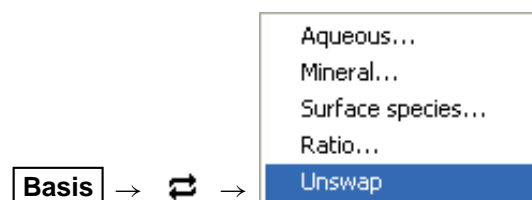
Use the “unsuppress” command (also: “include”) to include in the calculation aqueous species, minerals, solid solutions, or gases that have previously been suppressed.  
Examples:

```
unsuppress Quartz Albite "Albite low"  
unsuppress ALL
```

The argument “ALL” clears any species, minerals, solid solutions, or gases that have been suppressed.

### 3.65 unswap

```
unswap <species | ALL>
```



Use the “unswap” command to reverse a basis swap. Example:

```
unswap Quartz (or unswap SiO2(aq))
```

to reverse the effect of the previous command

```
swap Quartz for SiO2(aq)
```

At this point, SiO<sub>2</sub>(aq) is back in the basis. The “ALL” argument reverses all basis swaps.

### 3.66 usgovt

**usgovt**

**Help** → **About Act2...**

Display the restricted rights legend that governs use, duplication, or disclosure by the U.S. Government.

### 3.67 water\_limits

**water\_limits** <off>

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Diagram**

Water, depending on its oxidation state, can decompose to form oxygen or hydrogen gas. The “water\_limits” command controls whether **Act2** will determine the stability limits of water at the current pressure (see the “pressure” command) and constrain the diagram to within these limits. If this option is not selected, the program draws a diagram to fill the axis range, regardless of whether water is stable over the diagram. **Act2**, by default, honors the stability limits of water in chemical systems where water can decompose. The optional argument “off” de-selects the option. Type “show options” to display the current option setting.

### 3.68 x-axis

**x-axis** <from value> <to value> <increment value>  
<length value> <margin value>

**Format** → **Appearance...** → **Axis Range**

**Format** → **Appearance...** → **Plot Area**

**Basis** → from  ▼ to  ▼ increment  ▼

Use the “x-axis” command (abbrev.: “x”) to set the range, tick increment, length and margin for the horizontal axis. Examples:

```
x-axis from -6 to 0 increment .5
x 0 14 2
x length 6 margin 2
```

Axis ranges are set on a log scale, and the axis length and margin are set in inches. The keywords “from”, “to”, and “increment” may be omitted, and any keyword may be abbreviated to two letters.

### 3.69 y-axis

**y-axis** <from value> <to value> <increment value>  
<length value> <margin value>

**Format** → **Appearance...** → **Axis Range**

**Format** → **Appearance...** → **Plot Area**

**Basis** → from  ▼ to  ▼ increment  ▼

Use the “y-axis” command (abbrev.: “y”) to set the range, tick increment, length and margin for the vertical axis. Examples:

```
y-axis from -6 to 0 increment .5
y 0 14 2
y length 6 margin 2
```





# Tact Commands

---

## 4.1 activity

**activity** <species> = <value | ?>

**Basis** →  ▼ activity ▼

Use the “activity” command (abbrev.: “a”) to constrain the activity of an aqueous species or water over the diagram. Examples:

```
activity Na+ = 0.3
activity main = 10^-5
a H2O = 0.9
log a H+ = -5
```

The label “main” refers to the species being diagrammed. Use “?” to unset an activity value. See also the “pH”, “Eh”, “pe”, “fugacity”, and “ratio” commands.

## 4.2 add

**add** <basis species>

**Basis** →

Use the “add” command to include a basis species in the calculation. Example:

```
add HCO3-
```

See also the “swap”, “activity”, “fugacity”, “ratio”, “pH”, “pe”, and “Eh” commands.

## 4.3 alter

```

alter <species | mineral | gas> <log Ks>
alter <species | mineral | gas> <poly coefs> TminK = <value> TmaxK = <value>
alter <solid solution> <type> \
  <from <value> to <value> step <value>> <<variable> = <value>>

Config → Alter Log Ks...
Config → Solid Solutions...

```

Use the “alter” command to change the temperature expansion for a species’, mineral’s, or gas’ log  $K$  or to change the properties of a solid solution.

Temperature expansions for log  $K$ s are given in the thermo dataset either as  $T$ -tables, or by up to six coefficients of a polynomial. When the current thermo dataset uses  $T$ -tables, you list replacement values at each of the principal temperatures specified in the dataset, most commonly 0°C, 25°C, 60°C, 100°C, 150°C, 200°C, 250°C, and 300°C. Example:

```
alter Anhydrite -4.3009 -4.4199 -4.7126 -5.1758 -6.2299 500 500 500
```

Values of “500” represent a lack of data at the corresponding temperature.

For a thermo dataset constructed from polynomial expansions, on the other hand, list instead up to six polynomial coefficients

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4
```

You can optionally append a temperature range of validity, in Kelvins, for the polynomial:

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4 TminK= 293 TmaxK= 383
```

Absent a range, the polynomial is taken to span the principal temperatures. You may adjust a reaction’s temperature range without specifying polynomial coefficients, but setting only coefficients resets the temperature range.

For a solid solution, use the “alter” command to set the activity coefficient model, the composition range in terms of the mole fraction of the most recently specified end member, and parameters for the chosen activity model, as described for the “solid\_solution” command. A solid solution “my\_ss” might be set as a subregular Guggenheim solution with the command

```
alter my_ss guggenheim a0 = 1 a1 = 2
```

You may prefer to use the “alter” command to adjust an entry from the thermo database, and the “solid\_solution” command to reconfigure a user-defined solution, but the two commands work interchangeably for this purpose.

Type “show alter” to list altered species or solutions and their settings; the “unalter” command reverses the process.

## 4.4 background

**background** <color name>

**Format** → **Appearance...** → **Plot Area**

Use the “background” command to set the background color of the plot. Example:

```
background "sky blue"
```

Type “show lines” for current color settings; “show colors” for color choices. Enclose multiword colors in quotes.

## 4.5 banner

**banner** <off>

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Title**

The “banner” command controls whether **Tact** will plot a banner along the side of the diagram summarizing the calculation constraints. The optional argument “off” de-selects the option. Type “show options” to display the current option setting.

## 4.6 chdir

**chdir** <directory> <ALL>

**File** → **Working Directory...**

Use the “chdir” command (abbrev.: “work\_dir”, “cd”) to change the working directory. The program reads input scripts relative to the current working directory and writes output into it. Typing the command “chdir” without an argument causes the program to display the name of the working directory. The “ALL” argument changes the working directories of all of the GWB apps. The command

```
chdir ~
```

changes to the user’s home directory, if one is defined by the operating system.

## 4.7 clear

**clear**

**View** → **Clear Screen**

Use the “clear” command (abbrev.: “c”) or right-click in the text window and choose **Clear Screen** to clear it.

## 4.8 clipboard

**clipboard**  
**clipboard bitmap**

**Edit** → **Copy**

**Edit** → **Copy As** → **Bitmap**

The “clipboard” command causes the contents of the **Plot** pane to be copied to the clipboard, from where it can be pasted into various types of documents. The “bitmap” keyword causes the plot to be copied as a bitmap, rather than a vector graphic.

## 4.9 color\_by\_sample

**color\_by\_sample** <off>

**Format** → **Scatter Data...**

The “color\_by\_sample” command causes **Tact** to show each scatter data sample label in the color of its related marker. The optional argument “off” causes all of the sample labels to be shown in the color set in **Format** → **Scatter Data...**

## 4.10 couple

**couple** <redox species | element(s) | **ALL**>

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “couple” command to enable any redox coupling reactions that have been disabled with the “decouple” command. You specify one or more redox species or elements. For example, the command

**couple** Carbon

couples all redox reactions involving the element carbon. Argument “ALL” enables all of the coupling reactions in the thermo dataset.

## 4.11 custom

```
custom <color number> <name> <red green blue>
```

**Color** → **Define Custom Colors**

The “custom” command allows you to change the name and/or RGB values of the eleven custom colors. Specify the custom color number (1 - 11), the name in quotes, and the RGB values (0 - 255). Examples:

```
custom 2 "Med Blue" 128 128 255
custom 2 "Med Blue"
custom 2 128 128 255
```

## 4.12 data

```
data <thermo dataset> <verify>
```

**File** → **Open** → **Thermo Data...**

Use the “data” command to change the input file of thermodynamic data. Example:

```
data "..\my_thermo.tdat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. The “verify” option causes the program to read the named dataset only if it has not already been read.

## 4.13 decouple

```
decouple <redox species | element(s) | ALL>
```

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “decouple” command to disable the coupling reactions for one or more redox species, in order to calculate a model assuming redox disequilibrium. The redox species then become available for use as basis species and may be constrained independently of the original basis entries. You can disable as many coupling reactions as you want.

You specify either one or more redox species or elements. For example, the command

decouple Carbon

decouples all redox reactions involving the element carbon. Argument "ALL" disables all of the coupling reactions in the thermo dataset. Use the "couple" command to enable coupling reactions, once they have been disabled.

### 4.14 diagram

**diagram** <main species> <vs axis> <activity = value>

Use the "diagram" command (also: "plot") to set the species to be diagrammed, its activity, and the diagram axis. Examples:

diagram Fe++ vs pH activity = 1e-4  
 diagram U++++ log activity = -10  
 diagram x= Cl-

The keyword "x" may be substituted for "vs". The species to be diagrammed, as well as the species appearing on the axis, must have been swapped into the basis if they do not appear in the original basis.

### 4.15 Eh

**Eh** = <value in volts | ?>

**Basis** →  Eh ▼

Use the “Eh” command to set Eh over the diagram. Example:

```
Eh = 1/2
```

sets the system’s oxidation state to correspond to an Eh of 0.5 volts. Use “?” to unset an Eh value. See also the “activity”, “pH”, “pe”, “fugacity”, and “ratio” commands.

## 4.16 explain

```
explain <species | mineral(s) | solid solution(s) | gas(es)>
```

**Config** → **Show...** → *select species*

Use the “explain” command to get more information (such as the mole weight of a species, a mineral’s formula, mole volume, and density, or a solid solution’s end-member minerals) about species, minerals, solid solutions, and gases in the dataset. Example:

```
explain Analcime
```

## 4.17 extrapolate

```
extrapolate <on | off>
```

**Config** → **Options...**

Use the “extrapolate” option to cause the program to extrapolate log  $K$ s for reactions forming aqueous species, minerals, and gases to temperatures beyond the data’s known range of validity. The option should be used with considerable care.

The temperature range of validity for a species’ log  $K$  is defined by its temperature expansion, taken from the thermo dataset. For  $T$ -table expansions, it is the span of principal temperatures at which log  $K$ s are not 500, whereas for polynomial expansions, the range is listed explicitly.

In normal operation, the program loads only species with ranges of validity encompassing the calculation’s temperature range. When the “extrapolate” option is “on”, in contrast, the program projects to the temperature of interest the log  $K$ s of species that would not otherwise be loaded.

## 4.18 font

```
font <font name>
```

Use the “font” command to select the font to be used to label the diagram. Example:

```
font "courier new"
```

You may select any font name available in the **Choose Font...** dialog. Enclose multiple word names in quotes. To change the font for individual aspects of the diagram, right-click on the aspect and choose **Font** →.

### 4.19 fugacity

```
fugacity <gas> = <value | ?>
```

```
Basis →  ▼ fugacity ▼
```

Use the “fugacity” command (abbrev.: “f”) to set gas fugacities (on an atm scale) over the diagram. Examples:

```
fugacity O2(g) = .2  
f CO2(g) = 0.0003  
log f S2(g) = -30
```

Use “?” to unset a fugacity value: See also the “activity”, “pH”, “Eh”, “pe”, and “ratio” commands.

### 4.20 gases

```
gases <off>
```

```
Format → Quick Toggle...
```

```
Format → Appearance... → Diagram
```

The “gases” command controls whether **Tact** will consider gaseous species when it calculates the full diagram. By default, the program includes any gases that contain the species being diagrammed and that can be made up from the basis entries; it assigns each a fugacity equal to the total pressure. Use this command with the “off” argument to cause gases to be excluded from the calculation. See also the “pressure” command.



## 4.21 go

**go**

**Run** → **Go**

**Plot**

Use the “go” command to initiate calculation of a diagram based on the current chemical system.

## 4.22 grep

**grep** <string>

**File** → **View** → **.\Tact\_output.txt** → **Edit** → **Find**

Use the “grep” command to search for each occurrence of the given character string in the current output dataset. For example, use

grep Albite

to obtain a list of each line in “Tact\_output.txt” containing the string “Albite”.

## 4.23 gridlines

**gridlines** <off>

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Axes and Ticks**

The “gridlines” command controls whether **Tact** will draw gridlines on the diagram. The optional argument “off” de-selects the option.

## 4.24 help

**help** <help | subject>

**Help**

Use the “help” command (also: “man”) to display online documentation for the GWB package. The command “help help” shows a list of the types of documentation available.

## 4.25 history

**history**

**Run** → **History...**

Use the “history” command to view a list of previously executed commands. You can re-execute commands in the **History...** dialog box by double-clicking on a command, selecting one or more commands, and then choosing **Execute**, or with a history substitution command, as explained in the **Command Line Interface** chapter in the **GWB Reference Manual**.

## 4.26 labels

**labels** <off>

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Diagram**

The “labels” command controls whether **Tact** will label the diagram with the names of predominant aqueous species and stable minerals and gases. Other types of lettering on the plot, such as axis labels, may be eliminated by using the “line” command to omit the aspect. **Tact**, by default, includes species labels on the diagram. The optional argument “off” de-selects the option. Type “show options” to display the current option setting.

## 4.27 legend

**legend** <off>

**legend** x <value | ?> y <value | ?>

**Format** → **Quick Toggle...**

**Format** → **Scatter Data...**

Use the “legend” command to control whether a legend related to scatter data is shown. The legend is only available when scatter data is currently loaded. Keyword “off” hides the legend. Keywords “x” and “y” set the position of the legend. Examples:

```
legend
legend off
legend x 2 y 7
```

## 4.28 line

```

line <aspect> <pen thickness> <color> <size = character size>
      <font = <noitalic | italic>> <font = <nobold | bold>>
      <noomit | include | omit>
line <aspect> <index> <pen thickness> <color> <size = character size>
      <symbol = symbol> <noomit | include | omit>

```

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Diagram**

**Format** → **Appearance...** → **Axes and Ticks**

**Format** → **Appearance...** → **Plot Area**

**Format** → **Appearance...** → **Title**

**Format** → **Scatter Data...**

**Format** → **Trace...**

Use the “line” command to change the boldness and color of lines used to construct the diagram, set the point size for lettering, set bold and/or italic lettering, or omit an aspect of the lettering. Examples:

```

line axes fine cyan
line axis_labels size = 18
line axis_labels font = bold
line axis_labels omit

```

The command also sets the boldness and color of reaction traces and the symbol, color, and point size of trace markers. If multiple reaction traces are loaded and the “trace appearance individual” option has been enabled (see the “trace” command), set the appearance of the trace, markers, marker\_start, or marker\_end aspects for a specific dataset by specifying its index immediately after the aspect argument. Of the currently loaded datasets, that loaded first has an index of 1, the second an index of 2, and so on. Examples:

```

line trace 1 "Grey (90%)" xx-bold
line trace 2 "Grey (60%)" med-bold
line marker_start 1 noomit symbol blot

```

Set the lettering and marker size in points (values  $\geq 1$ ) or inches (values  $< 1$ ). Type “show lines” for aspects that can be changed and their current settings. Type “show pens”, “show colors”, and “show symbols” for available pens, colors, and symbols.

## 4.29 log

```
log <variable> = <value>
```

Use the “log” command to set variables on a logarithmic scale. Examples:

```
log fugacity O2(g) = -65  
log activity U++++ = -10  
diagram Cu+ log activity = -8
```

## 4.30 minerals

```
minerals <off>
```

```
Format → Quick Toggle...
```

```
Format → Appearance... → Diagram
```

The “minerals” command controls whether **Tact** will consider mineral species when it calculates the full diagram. By default, the program includes each mineral that contains the species being diagrammed and that can be made up from the basis entries. Use this command with the “off” argument to cause minerals to be excluded from the calculation.

## 4.31 mosaic\_bounds

```
mosaic_bounds <off>
```

```
Format → Quick Toggle...
```

```
Format → Appearance... → Diagram
```

When calculating a mosaic diagram (see the “speciate” command), **Tact** normally draws the boundaries among sub-diagrams using a finely dashed line. The “mosaic\_bounds” command controls whether these lines are drawn. **Tact**, by default, draws the mosaic boundaries. The optional argument “off” de-selects the option. Type “show options” to display the current option setting.

### 4.32 mosaic\_labels

```
mosaic_labels <off>
```

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Diagram**

The “mosaic\_labels” command controls whether **Tact** labels the subdiagrams in a mosaic diagram (see the “speciate” command). The program, by default, does not label these fields. The command “mosaic\_labels” causes the labels to be drawn; the optional argument “off” de-selects the option. Type “show options” to display the current option setting.

### 4.33 paper

```
paper height = <value | ?> width = <value | ?> \
  factor = <value | ?> \
  orientation = <portrait | landscape | ?> \
  units = <in | cm | ?> scale = <on | off | ?>
paper reset
```

**File** → **Page Setup...**

The “paper” command controls printing from **Tact**, including the paper size and diagram orientation. Keywords “height” and “width” specify paper dimensions, “factor” is an optional scaling factor applied to the plot upon printing, “orientation” controls whether the plot prints as portrait or landscape, “units” tells whether the page dimensions are in inches or centimeters, and “scale” controls whether the plot is scaled automatically to the size of the printed page. Keywords “selection” and “paper” are for internal use by the program.

Type

```
show paper
```

to see the current settings, and

```
paper reset
```

to restore settings to their default values.

### 4.34 pause

```
pause
```

Use the “pause” command to cause **Tact** to pause temporarily during input. This command is useful when you are debugging scripts.

### 4.35 pe

**pe** = <value | ?>

**Basis** →  ▼ pe ▼

Use the “pe” command to set oxidation state over the diagram in terms of pe. Example:

pe = 10

is equivalent to

log activity e- = -10

where “e-” is the electron. Use “?” to unset a pe value: See also the “activity”, “Eh”, “pH”, “fugacity”, and “ratio” commands.

### 4.36 pH

**pH** = <value | ?>

**Basis** →  ▼ pH ▼

Use the “pH” command to set pH over the diagram. Example:

pH = 5

is equivalent to

log activity H+ = -5

Use “?” to unset a pH value: See also the “activity”, “Eh”, “pe”, “fugacity”, and “ratio” commands.

### 4.37 polymorphs

**polymorphs** <mineral>

**Config** → **Show...** → Minerals → *select mineral*

Use the “polymorphs” command to list each polymorph (mineral identical in composition to the specified mineral) in the current database. Example:

```
polymorphs Albite
```

### 4.38 pressure

```
pressure = <value | vapor> <unit>
```

```
Basis → pressure →  ▼ bars ▼
```

Use the “pressure” command (also: “P”) to set total pressure. Pressure affects the position of water stability limits and the fugacities assigned to gaseous species, but not mineral or species stability. Examples:

```
pressure 20 atm
pressure 1.01 bar
P 0.1 MPa
pressure vapor
```

The default setting is the temperature expansion for pressure given at the head of the thermo dataset in use. In many datasets, pressure is set to 1 atm below 100°C and the vapor pressure of water above this temperature. Keyword “vapor” restores the default setting; the default unit for the “pressure” command is bar.

### 4.39 print

```
print <off | on>
```

```
Config → Output...
```

The command “print” (also: “printout”) with the argument “off” causes **Tact** to bypass writing calculation results to the “Tact\_output.txt” dataset. By default, the program writes to this dataset. The command “print on” (or just “print”) re-enables the output.

### 4.40 pwd

```
pwd
```

```
File → Working Directory...
```

The “pwd” command returns the name of the current working directory. The command has the same effect as typing “show directory”. See the “chdir” command.

## 4.41 quit

**quit**

**File** → **Quit**

Use the “quit” command (also: “end”, “q”) to exit the program. Touching **Ctrl+Q** serves the same purpose as the “quit” command.

## 4.42 ratio

**ratio** <act./fug. ratio> = <value | ?>

**Basis** →  ▼ ratio ▼

Use the “ratio” command to constrain an activity or fugacity ratios over the diagram. Example:

```
swap Ca++/Na+^2 for Ca++
ratio Ca++/Na+^2 = 0.2
```

Use “?” to unset a ratio value: See also the “activity”, “pH”, “Eh”, “pe”, and “fugacity” commands.

## 4.43 read

**read** <dataset>

**File** → **Open** → **Read Script...**

Use the “read” command to begin reading commands from a script stored in a dataset. Example:

```
read Cu_solubility
```

Control returns to the user after the script has been read, unless the script contains a “quit” command. You can also use the “read” command in place of the “data” or “surface\_data” command to read a thermo or surface reaction dataset.

When typing a “read” command, you can use the spelling completion feature to complete dataset names: touch “[tab]” or “[esc]” to cycle through the possible completions, or **Ctrl+D** to list possible completions.



## 4.44 remove

```
remove <basis specie(s)> <solid solution(s)>
remove basis <basis specie(s)>
remove solid_solution <solid solution(s)>
```

**Basis** or  
**Config** → **Solid Solutions...** →  
*select entry(ies)* →

Use the “remove” command (also: “rm”) to eliminate one or more basis entries from consideration in the calculation. Example:

```
remove Na+
remove Quartz Calcite
```

Components can be reentered into the basis using the “swap”, “add”, “activity”, and “fugacity” commands. You can also use the “remove” command to remove solid solutions configured in a run (but not those set in the thermo dataset):

```
remove solid_solution mySS
```

## 4.45 reset

```
reset
```

**File** → **Reset Configuration**

Use the “reset” command to begin defining the chemical system again with a clean slate. Your current settings will be lost, and all options will be returned to their default states. The command, however, does not alter the setting for the thermo dataset.

## 4.46 resume

```
resume
```

**File** → **Resume**

Use the “resume” command to restore the configuration from when you last exited the program. The program configuration is stored in a file “tact\_resume.tac” in your profile directory (e.g., “c:\Documents and Settings\jones\Application Data\GWB”). Your current configuration is discarded.

## 4.47 save

```
save <dataset> <hex>
```

**File** → **Save As...**

Use the “save” command to write the current chemical system into a dataset in **Tact** format commands. The dataset can be used as an **Tact** input script. Examples:

```
save  
save kspar.tac
```

If you don't specify a dataset name, **Tact** will ask for one. The optional keyword “hex” causes the program to output numbers as hexadecimal values.

## 4.48 scatter

```
scatter <dataset>
```

**File** → **Open** → **Scatter Data...** or **Format** → **Scatter Data...**

Use the “scatter” command to project data points onto the diagram. Data points are read from the specified dataset; “scatter off” turns off the option. The preferred way to add scatter data is to import it from a **GSS** spreadsheet (“.gss” file, see the **Using GSS** and **Using Act2** chapters of the **GWB Essentials Guide**). The old method of importing a specially formatted table from a text file still works, however (see the **Scatter Data** chapter in the **GWB Reference Manual**).

## 4.49 script

```
script  
script end
```

Use the “script” command to mark the beginning, and optionally the end, of a control script. Control scripts differ from standard input files in that they can contain not only **Tact** commands, but control structures such as loops and if-else branches. Control scripts follow the Tcl syntax, described in [www.tcl.tk](http://www.tcl.tk) and [mini.net/tcl](http://mini.net/tcl), as well as several widely available textbooks.

Within a control script, filenames are written with double rather than single backslashes. For example, a “read” command might appear as

```
read GWB_files\\My_file.tac
```

within a control script.

## 4.50 show

```
show <option>
show <aqueous | minerals | solid_solutions | gases> \
  <with | w/> <basis entry | string>

Config → Show...
```

Use the “show” command to display specific information about the current system or database. The options are:

show	show all	show altered
show basis	show colors	show commands
show couples	show directory	show elements
show gases	show lines	show minerals
show options	show oxides	show paper
show pens	show species	show solid_solutions
show suppressed	show symbols	show system
show show		

The command “show show” gives a list of show command options. When you type “show aqueous” (or “minerals”, “solid\_solutions”, or “gases”), the program lists all entries of that type in the thermo database. The “solid\_solutions” option additionally includes solutions defined locally. A long form lets you limit the query to entries composed of a particular basis species or containing a text string in the name:

```
show aqueous with Al+++
show minerals w/ chal
```

There is also a compound form of the “show couples” command:

```
show coupling reactions
```

This command produces a complete list of the redox couples, in reaction form.

## 4.51 solid\_solution

```

solid_solution <name = <label>> <type> \
  <<mineral> tag = <label>> <contains> <<mineral> tag = <label>> \
  <from <value>> <to <value>> <step = <value>> <<variable> = <value>>
solid_solution <solid solution> <options>
solid_solution remove <solid solution>

```

**Config** → **Solid Solutions...** →

**Config** → **Solid Solutions...** → *select entry(ies)* →

Use the “solid\_solution” command (abbrev.: “solid\_soln”) to define binary solid solutions in addition to any provided in the current thermo database. Supply a name with the “name” keyword, or simply set one as the first argument.

You can select the type of solid solution, either “ideal” (the default), “site\_mixing”, “guggenheim”, “regular”, “cubic”, or “third\_order”. For a site-mixing solution, set the site number “bsite”, which defaults to 1. For a Guggenheim solution, set dimensionless parameters “a0”, “a1”, and “a2” or a temperature expansion for their J/mol counterparts using coefficients “p1”-“p9”, each of which defaults to zero. For the other nonideal models, similarly, set coefficients “p1”, “p2”, etc., as appropriate. For more information, see **Solid solutions** under **Configuring the programs** in the **GWB Essentials Guide** and the **Thermo Datasets** chapter in the **GWB Reference Manual**.

You need to specify two end members for a solid solution, each associated with an abbreviated “tag”. The tag cannot be the name of another entry in the database. You can set a composition range in terms of the mole fraction of the most recently referenced end member, using keywords “from” and “to”, as well as a compositional “step” separating tranches of a discrete solution; by default, composition ranges from zero to one and the step is one-twentieth of the range. Keyword “contains” may be inserted for clarity. Examples:

```

solid_solution name = my_ss Calcite contains Strontianite from 0% to 10% a0= 0.05
solid_solution my_ss Calcite Strontianite from 0 to .1 a0= 0.05
solid_solution Biotite site_mixing Annite Phlogopite bsite= 3

```

If you don’t specify a solid solution’s name, the app will create one automatically using the tags for the two end members, separated by a hyphen. If you don’t set a tag for an end member, the app will pick up the first six characters of the corresponding mineral’s name. For example, the command

```
solid_solution Calcite contains Strontianite
```

produces a solid solution named “Calcit-Stront”, whereas typing

```
solid_solution Calcite tag Cc contains Strontianite tag Str
```

yields the name “Cc-Str”. To redefine the properties of a solid solution, enter the name followed by the properties to be changed:

```
solid_solution my_ss a0 = 1 a1 = 2
```

(You may prefer to use the “alter” command to change solutions defined within the thermo database; the two commands are functionally equivalent.) The command

```
solid_solution remove my_ss
```

discards a previously defined solution.

## 4.52 span

```
span <value | ?> <C> <to> <value | ?> <C>  
span <off | on | reset>
```

**Config** → **Options...**

The “span” command lets you set a temperature range across which stability must be defined for a species to be considered in the calculation.

For example, you may wish to compare results calculated at 25°C with those figured at 30°C. Since  $\log K$  for a number of species in the thermo dataset is given at only 25°C, unless the “extrap” option is set, the former calculation may well take into account a somewhat larger set of species than the latter. Entering the command

```
span 25 C to 30 C
```

assures the same set of species is loaded in each case.

The lower limit may be identified with keyword “from” or “Tmin”, and the upper denoted by “to” or “Tmax”. A value of “?” unsets the lower or upper limit. Keyword “off” disables the feature, leaving values for the limits intact, and keyword “on” re-enables the feature. The “reset” keyword clears the limits, disabling the feature. By default, the span feature is disabled.

## 4.53 speciate



Use the “speciate” command to cause a basis species of the chemical system to react with the  $x$ -axis species to form complex ions and minerals. Example:  $\text{SO}_4^{--}$  can speciate over a pH axis to form  $\text{HSO}_4^-$ ;  $\text{H}_2\text{S}(\text{aq})$  can form  $\text{HS}^-$  and  $\text{S}^{--}$ . The program will calculate subdiagrams within each area of predominance resulting from the speciation and assemble the subdiagrams into a complete diagram.

Examples:

```
speciate HCO3-
speciate SO4--
speciate NO3-
```

The “unspeciate” command reverses the process.

## 4.54 suffix

**suffix** <string>

**Config** → **Output...**

Use the “suffix” command to alter the names of the output datasets (“Tact\_output.txt”) by adding a trailing string. Example:

```
suffix _run2
```

produces output datasets with names such as “Tact\_output\_run2.txt”.

## 4.55 suppress

**suppress** <species, minerals, solid solutions, gases | **ALL**>

**Config** → **Suppress...**

Use the “suppress” command (also: “kill”) to prevent certain aqueous species, minerals, solid solutions, or gases from being considered in a calculation. Example:

```
suppress H3SiO4- Quartz "Maximum Microcline"
```

prevents the three entries listed from being loaded from the database. Typing “suppress ALL” suppresses all of the minerals and solid solutions in the thermodynamic database, as well as any solid solutions defined locally. The “unsuppress” command reverses the process. To suppress all but a few minerals, you could type

```
suppress ALL
unsuppress Quartz Muscovite Kaolinite
```

## 4.56 swap

```
swap <new basis> <for> <basis species>
```

```
Basis → ↺
```

Use the “swap” command to change the set of basis entries. All reactions are written internally in terms of an initial list of basis species. The basis can be altered over the course of the calculation to reflect the chemical constraints to be imposed. An aqueous species, mineral, gas, or activity ratio can be swapped into the basis in place of one of the original basis species listed in the database. Examples:

```
swap CO3--      for HCO3-
swap Quartz     for SiO2(aq)
swap CO2(g)     for H+
swap O2(g)      for O2(aq)
swap Ca++/Na+^2 for Ca++
```

The new species must contain in its composition the original basis species being swapped out (you can't swap lead for gold). For example, CO<sub>2</sub>(g) is composed of HCO<sub>3</sub><sup>-</sup>, H<sup>+</sup>, and water in “thermo.tdat”. The reactions in the thermo dataset (once reduced to the set of basis and redox species and modified to reflect enabled redox couples) show the basis entries for which a species may be swapped. For a list of original basis species, type “show basis”. To reverse a swap, type “unswap <species>”.

## 4.57 system

```
system <system command>
```

Use the “system” command to execute a command on the computer's operating system (a “DOS command”) without leaving **Tact**. Example:

```
system "copy Tact_output.txt Tact_output2.txt"
```

Enclose multiword system commands in quotes. An alternative way to fork a system command is to begin the **Tact** command with a "\$":

```
$copy Tact_output.txt Tact_output2.txt
```

In this case, no quotation marks are needed.

When a "system" command is executed, a "Command Prompt" window will appear briefly on your screen. Due to limitations of the Windows operating system, you cannot fork a command that requires user input, and you will not be able to see any output (including error messages) that might be generated by the command.

### 4.58 title

```
title <character string | empty>  
title x <value | ?> y <value | ?>
```

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Title**

Use the "title" command to set a title to be written above the plot. Keyword "empty" eliminates the title. Keywords "x" and "y" set the position of the title. Examples:

```
title "Al solubility vs pH"  
title x 2 y 7
```

Enclose multiword titles in quotes.

### 4.59 trace

```
trace <dataset>  
trace appearance <global | individual>
```

**File** → **Open** → **Reaction Trace...** or **Format** → **Trace...**

**File** → **Update Trace**

Use the "trace" command to project the results of a **React** run onto the diagram. The **React** results are read from a "React\_plot.gtp" dataset in the working directory or from a specified dataset. Repeat the "trace" command to specify additional datasets. Examples:



```
trace
trace ..\React_plot_run1.gtp
trace ..\React_plot_run2.gtp
```

Delete a trace by including the “off” keyword at the end of the “trace” command. Examples:

```
trace ..\React_plot_run1.gtp off
trace off
```

The first command removes the specified dataset and the second turns off the projection option entirely.

Use the “appearance individual” keywords to allow details of a trace’s appearance (e.g., line thickness and color) to be set separately for each dataset. The “appearance global” keywords revert the option to the default setting, in which the appearance applies to all loaded reaction traces. The “line” command sets the appearance.

## 4.60 truncate

```
truncate <off>
```

**Format** → **Quick Toggle...**

**Format** → **Appearance...** → **Diagram**

The truncate option controls whether the diagram is truncated at the stability limits of water, whenever the limits are displayed, or extends across the axis ranges. By default, the diagram is truncated at the water limits, but the command “truncate off” extends the diagram over the full plot.

## 4.61 unalter

```
unalter <species | mineral | solid solution | gas | ALL>
```

**Config** → **Alter Log Ks...** → *select entry(ies)* → **delete**

Use the “unalter” command to reverse the effect of having changed the log  $K$ ’s for a species, mineral, or gas, or the properties of a solid solution in the current dataset. Example:

```
unalter Quartz
```

In this case, the log  $K$  values for quartz revert to those in the current thermodynamic dataset. The argument "ALL" resets the log  $K$ 's for all species, minerals, and gases, as well as the properties of all solid solutions in the dataset.

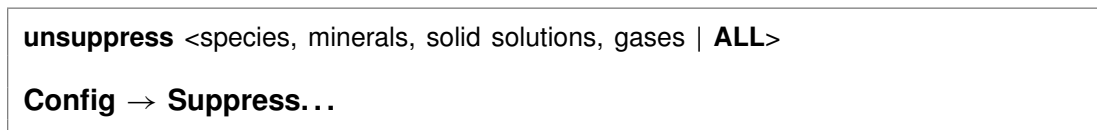
## 4.62 unspciate



Use the "unspciate" command to reverse the effects of the "speciate" command. Example:

```
unspciate SO4--
```

## 4.63 unsuppress

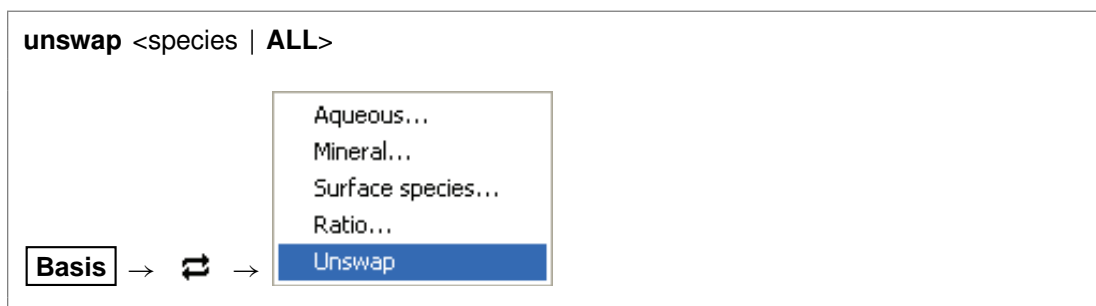


Use the "unsuppress" command (also: "include") to include in the calculation aqueous species, minerals, solid solutions, or gases that have previously been suppressed. Examples:

```
unsuppress Quartz Albite "Albite low"  
unsuppress ALL
```

The argument "ALL" clears any species, minerals, solid solutions, or gases that have been suppressed.

## 4.64 unswap



Use the “unswap” command to reverse a basis swap. Example:

```
unswap Quartz (or unswap SiO2(aq))
```

to reverse the effect of the previous command

```
swap Quartz for SiO2(aq)
```

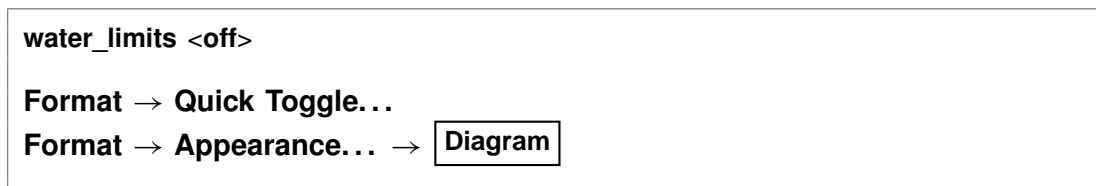
At this point, SiO<sub>2</sub>(aq) is back in the basis. The “ALL” argument reverses all basis swaps.

## 4.65 usgovt



Display the restricted rights legend that governs use, duplication, or disclosure by the U.S. Government.

## 4.66 water\_limits



Water, depending on its oxidation state, can decompose to form oxygen or hydrogen gas. The “water\_limits” command controls whether **Tact** will determine the stability limits of water at the current pressure (see the “pressure” command) and constrain the diagram to within these limits. If this option is not selected, the program draws a diagram to fill the axis range, regardless of whether water is stable over the diagram.

**Tact**, by default, honors the stability limits of water in chemical systems where water can decompose. The optional argument “off” de-selects the option. Type “show options” to display the current option setting.

## 4.67 x-axis

**x-axis** <from value> <to value> <increment value>  
<length value> <margin value>

**Format** → **Appearance...** → **Axis Range**

**Format** → **Appearance...** → **Plot Area**

**Basis** → from  ▼ to  ▼ increment  ▼

Use the “x-axis” command (abbrev.: “x”) to set the range, tick increment, length and margin for the horizontal axis. Examples:

```
x-axis from -6 to 0 increment .5
x 0 14 2
x length 6 margin 2
```

Axis ranges are set on a log scale, and the axis length and margin are set in inches. The keywords “from”, “to”, and “increment” may be omitted, and any keyword may be abbreviated to two letters.

## 4.68 y-axis

**y-axis** <from value> <to value> <increment value>  
<length value> <margin value>

**Format** → **Appearance...** → **Axis Range**

**Format** → **Appearance...** → **Plot Area**

**Basis** → from  ▼ to  ▼ increment  ▼

Use the “y-axis” command (abbrev.: “y”) to set the range, tick increment, length and margin for the vertical axis. Examples:

```
y-axis from 0 to 200 increment 25
y 100 200 10
y length 6 margin 2
```

# SpecE8 Commands

## 5.1 <unit>

<value> <free> <unit> <as element symbol> <basis entry>

**Basis** →  ▼ free kg ▼

To constrain the initial system, enter a command containing only the above entries. Entries may appear in any order. The qualifier “free” specifies that the constraint applies to the free rather than to the bulk basis entry. **SpecE8** recognizes the following units for constraining the initial system:

*By mass or volume:*

mol	mmol	umol	nmol	
kg	g	mg	ug	ng
eq	meq	ueq	neq	
cm3	m3	km3	l	

*By concentration:*

mol/kg	mmol/kg	umol/kg	nmol/kg
molal	mmolal	umolal	nmolal
mol/l	mmol/l	umol/l	nmol/l
g/kg	mg/kg	ug/kg	ng/kg
wt%	"wt fraction"		
g/l	mg/l	ug/l	ng/l
eq/kg	meq/kg	ueq/kg	neq/kg
eq/l	meq/l	ueq/l	neq/l

*By carbonate alkalinity (applied to bicarbonate component):*

eq_acid	meq_acid	ueq_acid	neq_acid
eq_acid/kg	meq_acid/kg	ueq_acid/kg	neq_acid/kg
eq_acid/l	meq_acid/l	ueq_acid/l	neq_acid/l
g/kg_as_CaCO3	mg/kg_as_CaCO3	ug/kg_as_CaCO3	ng/kg_as_CaCO3
wt%_as_CaCO3			
g/l_as_CaCO3	mg/l_as_CaCO3	ug/l_as_CaCO3	ng/l_as_CaCO3
mol/kg_as_Ca...	mmol/kg_as_Ca...	umol/kg_as_Ca...	nmol/kg_as_CaCO3
mol/l_as_CaCO3	mmol/l_as_CaCO3	umol/l_as_CaCO3	nmol/l_as_CaCO3

*Per volume of the system:*

## GWB Command Reference

mol/cm3	mmol/cm3	umol/cm3	nmol/cm3
kg/cm3	g/cm3	mg/cm3	ug/cm3
ng/cm3			
mol/m3	mmol/m3	umol/m3	nmol/m3
kg/m3	g/m3	mg/m3	ug/m3
ng/m3			
volume%	"vol. fract."		
<i>By activity:</i>			
activity	fugacity	ratio	
pH	V	pe	
<i>By partial pressure:</i>			
Pa	MPa	atm	bar
psi			

Activity and fugacity may be abbreviated to "a" or "f". Keyword "total" reverses a setting of "free".

Use the "as" keyword to constrain mass in terms of elemental equivalents. For example, the command

```
CH3COO- = 10 umol/kg as C
```

specifies 5 umol/kg of acetate ion, since each acetate contains two carbons, whereas

```
20 mg/kg SO4-- as S
```

would specify 59.9 mg/kg of sulfate, since the ion's mole weight is about 3 times that of sulfur itself.

Examples:

```
55 mg/kg HCO3-
Na+ = 1 molal
1 ug/kg U++++
100 free cm3 Dolomite
pH = 8
Eh = .550 V
log f O2(g) = -60
HCO3- = 30 mg/kg as C
```

## 5.2 <isotope>

```
<isotope | symbol> <fluid | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the name of any isotope system loaded in the isotope dataset (also, the isotope's symbol) to set the isotopic composition of the fluid or any segregated minerals. The composition may be set on any scale (e.g., SMOW, PDB, ...), but you must be consistent throughout the calculation.

For example, if the  $^{17}\text{O}$  isotope system has been added to the isotope dataset, you could enter:

```
oxygen-17 fluid = -10, Quartz = +15
  or
17-O fluid = -10, Quartz = +15
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
oxygen-17 remove
oxygen-17 off
```

clear all settings for  $^{17}\text{O}$  isotopes from the calculation.

### 5.3 activity

```
activity <species> = <value | ?>
```

**Basis** →  ▼ activity ▼

Use the “activity” command (abbrev.: “a”) to constrain the activity of an aqueous species in the system. Examples:

```
activity Na+ = 0.3
log a H+ = -5
```

See also the “pH”, “Eh”, “pe”, “ratio”, “fugacity”, “fix”, and “slide” commands.

### 5.4 add

```
add <basis species>
```

**Basis** →  add

Use the “add” command to include a basis species in the calculation. Example:

```
add HCO3-
```

See also the “swap”, “activity”, “fugacity”, “pH”, “pe”, and “Eh” commands.

## 5.5 alkalinity

**alkalinity** = <value> <unit>

**Basis** → HCO<sub>3</sub><sup>-</sup>  ▼ mg/kg\_as\_CaCO<sub>3</sub> ▼

Use the “alkalinity” command to constrain the total concentration of HCO<sub>3</sub><sup>-</sup> to reflect the solution’s carbonate alkalinity. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “mg/kg\_as\_CaCO<sub>3</sub>” is the default. To use this option, the solution pH must be set explicitly.

## 5.6 alter

**alter** <species | mineral | gas> <log Ks>

**alter** <species | mineral | gas> <poly coefs> **TminK** = <value> **TmaxK** = <value>

**alter** <surface species> <poly coefs> **TminK** = <value> **TmaxK** = <value>

**alter** <surface species> **logK** = <value> **dlogK/dT** = <value>

**alter** <exchange species> **beta** = <value>

**alter** <sorbed species> **Kd** = <value>

**alter** <sorbed species> **Kf** = <value> **nf** = <value>

**alter** <solid solution> <type> <**discrete** | **continuous**> \  
 <from <value> to <value> **step** <value>> <<variable> = <value>>

**Config** → **Alter Log Ks...**

**Config** → **Solid Solutions...**

Use the “alter” command to change the temperature expansion for a species’, mineral’s, or gas’ log *K*, to adjust the stability of a surface species, or to change the properties of a solid solution.

Temperature expansions for log *K*s are given in the thermo dataset either as *T*-tables, or by up to six coefficients of a polynomial. When the current thermo dataset uses *T*-tables, you list replacement values at each of the principal temperatures specified in the dataset, most commonly 0°C, 25°C, 60°C, 100°C, 150°C, 200°C, 250°C, and 300°C. Example:

```
alter Anhydrite -4.3009 -4.4199 -4.7126 -5.1758 -6.2299 500 500 500
```

Values of “500” represent a lack of data at the corresponding temperature.

For a thermo dataset constructed from polynomial expansions, on the other hand, list instead up to six polynomial coefficients

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4
```



You can optionally append a temperature range of validity, in Kelvins, for the polynomial:

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4 TminK= 293 TmaxK= 383
```

Absent a range, the polynomial is taken to span the principal temperatures. You may adjust a reaction's temperature range without specifying polynomial coefficients, but setting only coefficients resets the temperature range.

For a surface complex, use the “alter” command to set  $\log K$ , its temperature derivative, or both

```
alter >(w)FeOCa+ logK = 6.0 dlogK/dT = 0.02
```

whereas for an ion exchange reaction, you set the selectivity coefficient  $\beta$

```
alter >X2:Ca beta = .033
```

directly, rather than as a logarithm. For a sorbing species, set  $K_d$  or, for a Freundlich species,  $K_f$ ,  $n_f$ , or both, as follows

```
alter >Pb++ Kd = .03
alter >Sr++ Kf = .015 nf = .8
```

For a solid solution, use the “alter” command to set the activity coefficient model, discrete or continuous behavior, the composition range in terms of the mole fraction of the most recently specified end member, and parameters for the chosen activity model, as described for the “solid\_solution” command. A solid solution “my\_ss” might be set as a subregular Guggenheim solution with the command

```
alter my_ss guggenheim a0 = 1 a1 = 2
```

You may prefer to use the “alter” command to adjust an entry from the thermo database, and the “solid\_solution” command to reconfigure a user-defined solution, but the two commands work interchangeably for this purpose.

Type “show alter” to list altered species or solutions and their settings; the “unalter” command reverses the process.

## 5.7 b-dot

**b-dot**

**File** → **Open** → **Thermo Data...** → thermo.tdat

The “b-dot” command (formerly: “debye-huckel”) causes the program to read the “thermo.tdat” thermodynamic dataset, which invokes the “B-dot” form of the extended Debye-Hückel equation to calculate activity coefficients for aqueous species.

## 5.8 balance

```
balance <on> <basis entry>
balance <off>
```

**Basis** →  ▼ unit ▼ → Balance species

Use the “balance” command to specify the basis entry to be used for electrical charge balancing. The basis entry must be a charged aqueous species. By default, **SpecE8** balances on  $\text{Cl}^-$ .

The command “balance off” disables **SpecE8’s** charge balancing feature. In this case, the user is responsible for prescribing charge-balanced input constraints.

## 5.9 carbon-13

```
carbon-13 <fluid | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “carbon-13” command (also, “13-C”) to set the  $^{13}\text{C}$  isotopic composition of the fluid or any segregated minerals. The composition may be set on any scale (e.g., PDB), but you must be consistent throughout the calculation. Example:

```
carbon-13 fluid = -10, Calcite = +4
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
carbon-13 remove
carbon-13 off
```

clear all settings for  $^{13}\text{C}$  isotopes from the calculation.

See also the “<isotope>” section above, and the “hydrogen-2”, “oxygen-18”, and “sulfur-34” commands.

## 5.10 chdir

```
chdir <directory> <ALL>
```

**File** → **Working Directory...**

Use the “chdir” command (abbrev.: “work\_dir”, “cd”) to change the working directory. The program reads input scripts relative to the current working directory and writes

output into it. Typing the command “chdir” without an argument causes the program to display the name of the working directory. The “ALL” argument changes the working directories of all of the GWB apps. The command

```
chdir ~
```

changes to the user’s home directory, if one is defined by the operating system.

## 5.11 clear

```
clear
```

**View** → **Clear Screen**

Use the “clear” command (abbrev.: “c”) or right-click in the text window and choose **Clear Screen** to clear it.

## 5.12 clipboard

```
clipboard
```

**Edit** → **Copy**

The “clipboard” command causes the contents of the **Results** pane to be copied to the clipboard, from where it can be pasted into various types of documents.

## 5.13 conductivity

```
conductivity <conductivity dataset>
```

**File** → **Open** → **Conductivity Data...**

Use the “conductivity” command to change the input file of coefficients used to calculate electrical conductivity. Example:

```
conductivity "..\my_conductivity.dat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. Beginning with GWB11, the applications compute electrical conductivity using either of two different approaches, the USGS and APHA methods; the USGS method is the default. The required coefficients are defined in the files “conductivity-USGS.dat” and “conductivity-APHA.dat”, respectively, which are installed in the same directory as the thermo datasets (commonly “\Program Files\GWB\Gtdata”).

## 5.14 couple

```
couple <redox species | element(s) | ALL>
```

```
Config → Redox Couples... → select couple(s) → select with...
```

Use the “couple” command to enable any redox coupling reactions that have been disabled with the “decouple” command. You specify one or more redox species or elements. For example, the command

```
couple Carbon
```

couple all redox reactions involving the element carbon. Argument “ALL” enables all of the coupling reactions in the thermo dataset.

## 5.15 data

```
data <thermo dataset> <verify>
```

```
File → Open → Thermo Data...
```

Use the “data” command to change the input file of thermodynamic data. Example:

```
data "..\my_thermo.tdat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. The “verify” option causes the program to read the named dataset only if it has not already been read.

## 5.16 decouple

```
decouple <redox species | element(s) | ALL>
```

```
Config → Redox Couples... → select couple(s) → select with...
```

Use the “decouple” command to disable the coupling reactions for one or more redox species, in order to calculate a model assuming redox disequilibrium. The redox species then become available for use as basis species and may be constrained independently of the original basis entries. You can disable as many coupling reactions as you want.

You specify either one or more redox species or elements. For example, the command

```
decouple Carbon
```

decouples all redox reactions involving the element carbon. Argument “ALL” disables all of the coupling reactions in the thermo dataset. Use the “couple” command to enable coupling reactions, once they have been disabled.

## 5.17 density

```

density = <value | ?>
density = <TDS | chlorinity>
density = <batzle-wang | phillips>

Basis →  advanced → density  ▼ g/cm3
Config → Options...

```

You can use the “density” command to set in g/cm<sup>3</sup> the fluid density the program uses to convert compositional constraints to molality, the concentration unit it carries internally. If you set the initial Na<sup>+</sup> composition in mg/l, for example, the program needs to know the density of the initial fluid to determine Na<sup>+</sup> molality.

The program by default converts units using a density value it calculates automatically, as discussed below. This value is sufficient for most purposes, and hence it is generally not necessary to set fluid density explicitly. You might, however, want to set the density if you are working at high temperature, but your analysis is expressed per liter of solution at room temperature.

You can also use the “density” command to tell the program how to calculate the default density it uses to convert units, and the fluid density it reports in the simulation results. The program by default uses the Batzle-Wang equation to figure density, but you can use the command “density = phillips” to select the method of Phillips et al., instead.

As well, the program normally figures density as that of an NaCl solution with the same TDS as the fluid in question, at the temperature of interest. With the command “density = chlorinity” you can tell the program to instead use the density of an NaCl solution of equivalent chlorinity.

To restore automatic calculation, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 5.18 dual\_porosity

```
dual_porosity = <on | off> <spheres | blocks | fractures> \
  <geometry = <spheres | blocks | fractures | ?>> \
  <volfrac = value | ?>> \
  <<radius | half-width> = <value | ?> <unit>>
```

**Config** → **Dual Porosity...**

Use the “dual\_porosity” command (abbrev: “dual”) to configure stagnant zones in the simulation, using the dual porosity feature. Enable and disable the feature with the “on” and “off” keywords. Disabling the feature does not affect other settings, so re-enabling the feature returns the model to its most recent configuration.

With the “geometry” keyword, you configure the stagnant zone into spheres, blocks, or a fractured domain, the latter being slabs separated by fractures arrayed along the  $x$  direction. Alternatively, you can set the three configurations directly with keywords “spheres”, “blocks”, and “fractures”. Keyword “volfrac” sets the fraction of the nodal block’s bulk volume occupied by the stagnant zone.

The “radius” (or “half-width”, for blocks and fractures) keyword sets the zone’s characteristic dimension, in units of distance (see the **Units Recognized** chapter in the **GWB Reference Manual**; default is cm). You may also set any of the keywords recognized by **React**, **X1t**, and **X2t**, but these have no effect in **SpecE8**, since that program does not model reaction processes.

As an example, the command

```
dual_porosity geometry = spheres, radius = 50 cm, volfrac = 75%
```

configures the stagnant zone into spheres of half-meter radius that occupy three-quarters of the domain.

## 5.19 Eh

```
Eh = <value | ?> <unit>
```

**Basis** →  ▼ V ▼

Use the “Eh” command to set Eh in the initial system. Example:

```
Eh = 200 mV
```

sets the system’s oxidation state to correspond to an Eh of 0.2 volt. See also the “activity”, “pH”, “pe”, “fugacity”, “fix”, and “slide” commands.

## 5.20 epsilon

```
epsilon = <value | ?>
```

**Config** → **Iteration...**

Use the “epsilon” command to set the convergence criterion (dimensionless) for iterating to a solution of the equations representing the distribution of chemical mass. By default, this variable is set to  $5 \times 10^{-11}$ . To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 5.21 exchange\_capacity

```
exchange_capacity = <value | ?> <units>
exchange_capacity on <type> = <value | ?> <units>
```

**Config** → **Sorbing Surfaces...** → exchange capacity  ▾ eq ▾  
or **File** → **Open** → **Sorbing Surfaces...**

Use the “exchange\_capacity” command (abbrev.: “ex\_capacity” or “exch\_capacity”) to set the exchange capacity (i.e., the CEC) of the system when modeling ion exchange reactions or sorption according to Langmuir isotherms. For ion exchange reactions, you set units of electrical equivalents (“eq”, “meq”, and so on) or equivalents per gram of dry sediment (“eq/g”, “meq/g”, ...). For Langmuir reactions, you similarly set a value in mole units: “mol”, “mmol”, “mol/g”, “mmol/g”. If you set units per gram of sediment, the program multiplies the value entered by the mass of rock in the system (including equilibrium and kinetic minerals as well as inert volume) to get the system’s capacity.

If you read in a surface that sorbs by ion exchange or Langmuir isotherms, you must set a value for its exchange capacity. If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example

```
exchange_capacity on IonEx = .0008 eq/g
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the surface represented by the sample dataset “IonEx.sdat”, for example, is “IonEx”. You can use the “show” command to display the “type” of each active surface. See also the “surface\_data” and “inert” commands.

## 5.22 explain

**explain** <species | mineral(s) | solid solution(s) | gas(es) | surface species>

**Config** → **Show...** → *select species*

Use the “explain” command to get more information (such as the mole weight of a species, a mineral’s formula, mole volume, and density, or a solid solution’s end-member minerals) about species, minerals, solid solutions, and gases in the dataset. Example:

```
explain Analcime
```

## 5.23 extrapolate

**extrapolate** <on | off>

**Config** → **Options...**

Use the “extrapolate” option to cause the program to extrapolate log  $K$ s for reactions forming aqueous species, minerals, and gases to temperatures beyond the data’s known range of validity. The option should be used with considerable care.

The temperature range of validity for a species’ log  $K$  is defined by its temperature expansion, taken from the thermo dataset. For  $T$ -table expansions, it is the span of principal temperatures at which log  $K$ s are not 500, whereas for polynomial expansions, the range is listed explicitly.

In normal operation, the program loads only species with ranges of validity encompassing the calculation’s temperature range. When the “extrapolate” option is “on”, in contrast, the program projects to the temperature of interest the log  $K$ s of species that would not otherwise be loaded.

Temperature ranges of validity can also be set in the thermo dataset for the virial coefficients used to calculate “Pitzer” and SIT activity coefficients. The “extrapolate on” option works in a similar manner in such cases, allowing virial coefficients to be used at temperatures beyond the coefficients’ known ranges of validity.

## 5.24 fugacity

**fugacity** <gas> = <value | ?>

**Basis** →  ▼ fugacity ▼

Use the “fugacity” command (abbrev.: “f”) to set gas fugacities (on an atm scale) in the initial system. Examples:



```
fugacity O2(g) = .2
f CO2(g) = 0.0003
log f S2(g) = -30
```

Use “?” to unset a fugacity value: See also the “activity”, “ratio”, “pH”, “Eh”, “pe”, “fix”, and “slide” commands.

## 5.25 go

```
go
```

**Run** → **Go**

**Results** → **Run**

Use the “go” command to initiate calculation of a model based on the current configuration.

## 5.26 grep

```
grep <string>
```

**File** → **View** → **.\SpecE8\_output.txt** → **Edit** → **Find**

Use the “grep” command to search for each occurrence of the given character string in the current output dataset. For example, use

```
grep Albite
```

to obtain a list of each line in “SpecE8\_output.txt” (or the current output dataset, as modified by the “suffix” command) containing the string “Albite”.

## 5.27 gtpplot

```
gtpplot <gtpplot options>
```

**Run** → **Gtpplot**

The “gtpplot” command launches program **Gtpplot** from the **SpecE8** command line, invoking input from the current graphics interface dataset (by default, “SpecE8\_plot.gtp”; see the “suffix” command). For example,

```
gtplot -graph piper
```

starts **Gtplot** with a Piper diagram as the initial plot.

### 5.28 h-m-w

```
h-m-w
```

```
File → Open → Thermo Data... → thermo_hmw.tdat
```

Use the “h-m-w” command (abbrev.: “hmw”) to set the program to calculate species’ activity coefficients by using the Harvie-Møller-Weare equations. Executing this command automatically sets the input dataset of thermodynamic data to “thermo\_hmw.tdat”. Note that dataset “thermo\_hmw.tdat” supports calculations at 25°C only.

### 5.29 help

```
help <help | subject>
```

```
Help
```

Use the “help” command (also: “man”) to display online documentation for the GWB package. The command “help help” shows a list of the types of documentation available.

### 5.30 history

```
history
```

```
Run → History...
```

Use the “history” command to view a list of previously executed commands. You can re-execute commands in the **History...** dialog box by double-clicking on a command, selecting one or more commands, and then choosing **Execute**, or with a history substitution command, as explained in the **Command Line Interface** chapter in the **GWB Reference Manual**.

### 5.31 hydrogen-2

```
hydrogen-2 <fluid | segregated mineral> = <value>
```

```
Config → Isotopes...
```

Use the “hydrogen-2” command (also, “2-H”) to set the  $^2\text{H}$  isotopic composition of the fluid or any segregated minerals. The composition may be set on any scale (e.g., SMOW), but you must be consistent throughout the calculation. Example:

```
hydrogen-2 fluid = -120, Muscovite = -40
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
hydrogen-2 remove
hydrogen-2 off
```

clear all settings for  $^2\text{H}$  isotopes from the calculation.

See also the “<isotope>” section above, and the “carbon-13”, “oxygen-18”, and “sulfur-34” commands.

## 5.32 inert

```
inert = <value | ?> <units>
```

**Medium** → inert volume  ▼ cm3 ▼

Use the “inert” command to set the volume of non-reacting space in the system. You may set a value in units of volume, including  $\text{cm}^3$ ,  $\text{m}^3$ , and  $l$ , as well as volume% and “vol. fract.”. The default setting is zero and the default unit is  $\text{cm}^3$ .

Assuming you have not set a value for the initial fluid fraction in the system using the “porosity” command, the program figures the porosity over the course of the calculation as a derived variable. Specifically, it divides the fluid volume by the sum of the fluid volume, mineral volume, and inert volume, and reports this value as a result.

When you have set a value for initial porosity with the “porosity” command, on the other hand, the program works in the contrary sense. In this case, it calculates the inert volume as that required to form a system of the specified initial porosity; the program now ignores any entry you may have set using the “inert” command.

To restore the default state, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show”.

## 5.33 isotope\_data

```
isotope_data <dataset>
```

**File** → **Open** → **Isotope Data...**

Use the “isotope\_data” command (also: “idata”) to set the name of the database containing isotope fractionation factors for species, minerals, and gases. Example:

```
isotope_data Isotope.mydata
```

### 5.34 itmax

```
itmax = <value | ?>
```

**Config** → **Iteration...**

Use the “itmax” command to set the maximum number of iterations that may be taken in an attempt to converge to a solution for the equations representing the distribution of chemical mass. By default, this variable is set to 999. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

### 5.35 Kd

```
Kd <off>
```

**Config** → **Options...**

The “Kd” command controls whether the program calculates  $K_d$  distribution coefficients for sorbing components, in units of liters per kg sediment mass. This calculation requires that the mineral mass in the system (as specified for individual minerals and/or in terms of inert volume) be set correctly.

### 5.36 log

```
log <variable> = <value>
```

Use the “log” command to set variables on a logarithmic scale. Examples:

```
log fugacity O2(g) = -65  
log activity U++++ = -10
```

## 5.37 mobility

```
mobility = <surface_type> <value>
```

**Config** → **Sorbing Surfaces...** → mobility  ▼

or **File** → **Open** → **Sorbing Surfaces...**

Use the “mobility” command to set up a complexing surface in your model as a mobile colloid. A mobile colloid is composed of the mineral (or minerals) associated with a complexing surface, as well as the ion complexes present on that surface. Only datasets with model type “two-layer”, “three-layer”, or “cd-music” as set in the dataset header are surface complexation models, and hence only those datasets can be used to form a mobile colloid.

Since SpecE8 is a static model, the program reports the composition of mobile colloids in the system, but does not move them. To model transport of the colloidal fraction, you would typically use program **X1t** or **X2t** to trace a reactive transport simulation.

Mobility refers to the fraction of the surface in question that can move in the model by advection and dispersion. A surface with a mobility of one moves freely, whereas a mobility of zero sets the surface to be stationary. Intermediate values arise, for example, when some of the surface is attached to the medium, or when colloid motion is impeded by electrostatic interactions. By default in the software the mobility of any surface is zero.

To set a mobile colloid, begin by reading in a surface complexation dataset using the “surface\_data” command. Then, use the “mobility” command, referencing the surface’s label, to set the colloid’s mobility. The label is given at the head of the surface dataset, on a line beginning “Surface type”. The label in dataset “FeOH.sdat”, for example, is “HFO”. If you omit the label, the program will assume you are referring to the surface complexation dataset most recently read.

You can define the mobility as a field variable, which means you can have the program calculate mobility using an equation, script, or compiled function you provide. The program evaluates mobility at each block just once, at the start of the run.

Example:

```
surface_data FeOH.sdat
mobility HFO = 100%
```

where “HFO” is the label for the surface defined by dataset “FeOH.sdat”.

Restore the default behavior of immobility by entering a command such as

```
mobility HFO ?
```

## 5.38 oxygen-18

```
oxygen-18 <fluid | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “oxygen-18” command (also, “18-O”) to set the  $^{18}\text{O}$  isotopic composition of the fluid or any segregated minerals. The composition may be set on any scale (e.g., SMOW), but you must be consistent throughout the calculation. Example:

```
oxygen-18 fluid = -10, Quartz = +15
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
oxygen-18 remove
oxygen-18 off
```

clear all settings for  $^{18}\text{O}$  isotopes from the calculation.

See also the “<isotope>” section above, and the “carbon-13”, “hydrogen-2”, and “sulfur-34” commands.

## 5.39 pause

```
pause
```

Use the “pause” command to cause **SpecE8** to pause temporarily during input. This command is useful when you are debugging scripts.

## 5.40 pe

```
pe = <value | ?>
```

**Basis** →  ▼ pe ▼

Use the “pe” command to set oxidation state in the initial system in terms of pe. Example:

```
pe = 10
```

is equivalent to

```
log activity e- = -10
```

where “e-” is the electron. Use “?” to unset a pe value: See also the “activity”, “Eh”, “pH”, “fugacity”, “fix”, and “slide” commands.

## 5.41 permeability

```
permeability <intercept = value | ?> <unit> \
  <porosity = value | ?> \
  <mineral = value | ?>
```

**Medium** → A (porosity)  ▼  
 B (intercept)  ▼ darcy ▼

You use the “permeability” command to set the correlation by which the program calculates sediment permeability. **SpecE8** calculates this value as a reported variable consistent with **X1t** and **X2t**, but does not use it in its calculations.

The correlation gives log permeability in any of the units listed in the **Units Recognized** chapter of the **GWB Reference Manual** (darcys by default) as a linear function of the porosity (expressed as a volume fraction) of a nodal block and, optionally, the volume fractions of one or more minerals.

Examples:

```
permeability intercept = -11 cm2 porosity = 15
permeability Kaolinite = -8
```

The latter command adds a term for the mineral Kaolinite to the existing correlation. To remove a term from the correlation, set a value of “?”. The entry

```
permeability Kaolinite = ?
```

for example, removes the correlation entry for that mineral.

The default correlation is

$$\log k = -5 + 15\phi$$

where  $k$  is permeability in darcys and  $\phi$  is porosity (expressed as a fraction).

## 5.42 pH

pH = <value | ?>

**Basis** →  ▼ pH ▼

Use the “pH” command to set pH in the initial system. Example:

pH = 5

is equivalent to

log activity H<sub>+</sub> = -5

Use “?” to unset a pH value: See also the “activity”, “Eh”, “pe”, “fugacity”, “fix”, and “slide” commands.

## 5.43 phrqpitz

**phrqpitz**

**File** → **Open** → **Thermo Data...** → thermo\_phrqpitz.tdat

Use the “phrqpitz” command to set the program to calculate species’ activity coefficients using the Harvie-Møller-Weare equations, as implemented in the USGS program PHRQPITZ. Executing this command automatically sets the input dataset of thermodynamic data to “thermo\_phrqpitz.tdat”. Note that dataset “thermo\_phrqpitz.tdat” is primarily intended to support calculations at or near 25°C.

## 5.44 pitz\_dgamma

pitz\_dgamma = <value | ?>

**Config** → **Iteration...**

Use the “pitz\_dgamma” command to control the relative change in an activity coefficient’s value the program allows during each Newton-Raphson iteration, when the Harvie-Møller-Weare activity model has been invoked. By default, the program allows a 10% change, which corresponds to a value of 0.1.



## 5.45 pitz\_precon

```
pitz_precon = <value | ?>
```

**Config** → **Iteration...**

Use the “pitz\_precon” command to control the maximum number of passes the program takes through the pre-conditioning loop before beginning a Newton-Raphson iteration, when a virial activity model (Harvie-Møller-Weare or SIT) has been invoked. By default, the program makes up to 10 passes. In cases of difficult convergence, counter-intuitively, it can sometimes be beneficial to decrease this value.

## 5.46 pitz\_relax

```
pitz_relax = <value | ?>
```

**Config** → **Iteration...**

The “pitz\_relax” command controls under-relaxation when evaluating the Harvie-Møller-Weare equations. The program at each Newton-Raphson iteration assigns activity coefficients as a weighted average of the newly calculated value and the corresponding value at the previous iteration level. Setting pitz\_relax to zero eliminates under-relaxation, so the newly calculated values are used directly; a value of one, in contrast, should be avoided because it would prevent the activity coefficients from being updated. By default, the program carries an under-relaxation factor of 0.5.

## 5.47 plot

```
plot <character | binary> <on | off>
```

**Config** → **Output...**

Use the “plot” command to set the format of the “SpecE8\_plot.gtp” dataset. The dataset is written in XML, a standard format that is easy to parse for use with alternative plotting programs. Numerical data in the dataset can be represented in either standard decimal notation (keyword “character”) for user readability or a binary encoding (keyword “binary”) that maintains full precision of data. The default format, XML with binary encoded data, also zips the output file to reduce output size and improve file opening speed. The command “plot off” causes **SpecE8** to bypass writing calculation results to the “SpecE8\_plot.gtp” dataset, which is used to pass input to **Gtplot**. By default, the program writes output to the dataset. The command “plot on” (or just “plot”) re-enables the output. To see the current setting, type “show print”.

## 5.48 polymorphs

**polymorphs** <mineral>

**Config** → **Show...** → Minerals → *select mineral*

Use the “polymorphs” command to list each polymorph (mineral identical in composition to the specified mineral) in the current database. Example:

polymorphs Albite

## 5.49 porosity

**porosity** = <value | ?>

**Medium** → porosity  ▼

Use the “porosity” command to set (as a volume fraction) the initial porosity of the system. Porosity, the fraction of the system occupied by fluid, is the ratio of fluid volume to the sum of fluid, mineral, and inert volume.

The examples

porosity = 0.30  
porosity = 30%

are equivalent.

When you specify the porosity, the program will figure the difference between the volume of a system of the given porosity and fluid volume, and the volume taken up initially in the system by minerals and fluid. The program assigns this difference as inert, non-reactive volume (see the “inert” command). In this case, the program ignores any settings that may have been made with the “inert” command.

When you do not specify an initial porosity with the “porosity” command, on the other hand, the program calculates it from volumes in the system of fluid, minerals, and inert space. To restore this default behavior, enter the command with an argument of “?”.

## 5.50 press\_model

**press\_model** <Tsonopoulos | Peng-Robinson | Spycher-Reed | default | off>

**Config** → **Options...**

The “press\_model” command (also: “pressure\_model”) lets you control the method used to calculate fugacity coefficients and gas partial pressures. Three nonideal pressure models are coded in the software: Tsonopoulos, Peng-Robinson, and Spycher-Reed, as described in the **GWB Essentials Guide**.

By default, the pressure model is taken from the header lines of the thermo dataset in use, but you can use the “press\_model” command to override the default setting. Keywords “Tsonopoulos”, “Peng-Robinson”, and “Spycher-Reed” set the pressure model directly (you need only enter the first three letters), whereas “default” returns to the setting in the thermo dataset, and “off” (also: “ideal”) disables the feature, forcing all fugacity coefficients to one.

Examples:

```
press_model Peng-Robinson
press_model default
```

## 5.51 print

```
print <option> = <long | short | none>
print <off | on>
print <numeric | alphabetic>
```

**Config** → **Output...**

Use the “print” command (also: “printout”) to control the amount of detail to be written into the “SpecE8\_output.txt” dataset. For example, the dataset can contain information about each aqueous species, information on only species with concentrations greater than  $10^{-8}$  molal, or no species information.

Options, which may be abbreviated to three letters, and their default settings are:

species	long
surfaces	long
saturations	long
gases	long
basis	none
orig_basis	long
elements	long
reactions	none
stagnant	none

The “print” command can also be used to arrange entries in the output dataset either numerically or alphabetically:

```
print numeric
print alphabetic
```

To see the current print settings, type “show print”. Finally, the command “print off” causes **SpecE8** to bypass writing calculation results to the “SpecE8\_output.txt” dataset. By default, the program writes to the dataset. The command “print on” (or just “print”) re-enables the output.

### 5.52 pwd

```
pwd
File → Working Directory...
```

The “pwd” command returns the name of the current working directory. The command has the same effect as typing “show directory”. See the “chdir” command.

### 5.53 quit

```
quit
File → Quit
```

Use the “quit” command (also: “end”, “q”) to exit the program. Touching **Ctrl+Q** serves the same purpose as the “quit” command.

### 5.54 ratio

```
ratio <species ratio> = <value | ?>
```

```
Basis →  ▼ ratio ▼
```

Use the “ratio” command to constrain an activity ratio in the initial system. Example:

```
swap Ca++/Na+2 for Ca++
ratio Ca++/Na+2 = 0.2
```

Use “?” to unset a ratio value: See also the “activity”, “pH”, “Eh”, “pe”, “fugacity”, “fix”, and “slide” commands.

## 5.55 read

```
read <dataset>
```

**File** → **Open** → **Read Script...**

Use the “read” command to begin reading commands from a script stored in a dataset. Example:

```
read Seawater
```

Control returns to the user after the script has been read, unless the script contains a “quit” command. You can also use the “read” command in place of the “data” or “surface\_data” command to read a thermo or surface reaction dataset.

When typing a “read” command, you can use the spelling completion feature to complete dataset names: touch “[tab]” or “[esc]” to cycle through the possible completions, or **Ctrl+D** to list possible completions.

## 5.56 remove

```
remove <basis specie(s)> <solid solution(s)>
```

```
remove basis <basis specie(s)>
```

```
remove solid_solution <solid solution(s)>
```

**Basis** or

**Config** → **Solid Solutions...** →

*select entry(ies)* → **delete**

Use the “remove” command (also: “rm”) to eliminate one or more basis entries from consideration in the calculation. Example:

```
remove Na+
remove Quartz Calcite
```

Components can be reentered into the basis using the “swap”, “add”, “activity”, and “fugacity” commands. You can also use the “remove” command to remove solid solutions configured in a run (but not those set in the thermo dataset):

```
remove solid_solution mySS
```

## 5.57 report

```
report <option>  
report set_digits <value>
```

Once the program has completed a calculation, you can use the “report” command to return aspects of the calculation results. You can use this command from the **Command** pane to query the program interactively. More significantly, the “report” command provides a means of communicating calculation results to control scripts and external programs running GWB programs by remote control.

For arguments available, see the **Report Command** chapter in the **GWB Reference Manual**. For details about control scripts and the plug-in feature, see the **Control Scripts** and **Plug-in Feature** chapters of the **GWB Reference Manual**. Use the “set\_digits” argument to set the number of significant digits in the values reported, and the “set\_node” argument to identify the node of interest.

## 5.58 reset

```
reset  
reset system  
reset variables
```

**File** → **Reset Configuration**

Use the “reset” command to begin defining the chemical system again with a clean slate. Your current settings will be lost, and all options will be returned to their default states. The command, however, does not alter the setting for the thermo dataset. The “reset system” command resets only the initial system. Similarly, typing “reset variables” sets each settable variable to its default value.

## 5.59 resume

```
resume
```

**File** → **Resume**

Use the “resume” command to restore the configuration from when you last exited the program. The program configuration is stored in a file “spece8\_resume.sp8” in your profile directory (e.g., “c:\Documents and Settings\jones\Application Data\GWB”). Your current configuration is discarded.

## 5.60 save

```
save <dataset> <hex>
```

```
File → Save As...
```

Use the “save” command to write the current chemical system into a dataset in **SpecE8** format commands. The dataset can be used as an **SpecE8** input script. Examples:

```
save  
save kspar.sp8
```

If you don't specify a dataset name, **SpecE8** will ask for one. The optional keyword “hex” causes the program to output numbers as hexadecimal values.

## 5.61 script

```
script  
script end
```

Use the “script” command to mark the beginning, and optionally the end, of a control script. Control scripts differ from standard input files in that they can contain not only **SpecE8** commands, but control structures such as loops and if-else branches. Control scripts follow the Tcl syntax, described in [www.tcl.tk](http://www.tcl.tk) and [mini.net/tcl](http://mini.net/tcl), as well as several widely available textbooks.

Within a control script, filenames are written with double rather than single backslashes. For example, a “read” command might appear as

```
read GWB_files\\My_file.sp8
```

within a control script.

## 5.62 segregate

```

segregate <mineral(s)>
segregate <mineral> <value>
segregate <mineral> <initial value> <final value>
segregate <mineral> <value> Xi = <value> <value> Xi = <value>

```

**Config** → **Isotopes...** →

**Config** → **Isotopes...** → *select entry(ies)* →

The “segregate” command causes minerals to be isolated from isotopic exchange over the course of a reaction path. By default, a mineral in the equilibrium system remains in isotopic equilibrium with the fluid and other minerals. A segregated mineral, on the other hand, changes in isotopic composition only when it precipitates from solution; it alters the system’s composition only if it dissolves. Example:

```
segregate Quartz Calcite "Maximum Microcline"
```

Optionally, a fraction of a mineral’s mass may be isotopically segregated, and that fraction may vary linearly with reaction progress. Examples:

```

segregate Quartz 100%, Muscovite 7/10
segregate Ca-Saponite 100% 0%
segregate Ca-Saponite 80% Xi = .3, 20% Xi = .7

```

In the latter example, the program segregates 80% of the mass of Ca-Saponite until the reaction progress variable Xi reaches .3, decreases the segregated fractionation until it reaches 20% when Xi equals .7, and then holds the value constant until the end of the path. To display the isotopically segregated minerals, type “show isotopes”.

Note, you use the name of the corresponding mineral to segregate a solid solution end member.

## 5.63 show

```

show <option>
show <aqueous | minerals | solid_solutions | gases | surfaces> \
  <with | w/> <basis entry | string>

```

**Config** → **Show...**



Use the “show” command to display specific information about the current system or database. The options are:

show	show all	show altered
show aqueous	show basis	show commands
show couples	show directory	show elements
show gases	show initial	show isotopes
show minerals	show oxides	show printout
show show	show solid_solutions	show suppressed
show surfaces	show system	show variables

The command “show show” gives a list of show command options. When you type “show aqueous” (or “minerals”, “solid\_solutions”, “gases”, “oxides”, or “surfaces”), the program lists all entries of that type in the thermo database. The “solid\_solutions” option additionally includes solutions defined locally. A long form lets you limit the query to entries composed of a particular basis species or containing a text string in the name:

```
show aqueous with Al+++
show minerals w/ chal
```

There is also a compound form of the “show couples” command:

```
show coupling reactions
```

This command produces a complete list of the redox couples, in reaction form.

## 5.64 simax

```
simax = <value | ?>
```

```
Config → Iteration...
```

The “simax” command sets in molal units the maximum value of the stoichiometric ionic strength used in calculating water activity when Helgeson’s B-dot Debye-Hückel model is employed. By default, this variable is set to 3 molal. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 5.65 solid\_solution

```

solid_solution <name = <label>> <type> <discrete | continuous> \
  <<mineral> tag = <label>> <contains> <<mineral> tag = <label>> \
  <from <value>> <to <value>> <step = <value>> <<variable> = <value>>
solid_solution <solid solution> <options>
solid_solution remove <solid solution>

```

**Config** → **Solid Solutions...** →

**Config** → **Solid Solutions...** → *select entry(ies)* →

Use the “solid\_solution” command (abbrev.: “solid\_soln”) to define binary solid solutions in addition to any provided in the current thermo database. Supply a name with the “name” keyword, or simply set one as the first argument.

You can select the type of solid solution, either “ideal” (the default), “site\_mixing”, “guggenheim”, “regular”, “cubic”, or “third\_order”. For a site-mixing solution, set the site number “bsite”, which defaults to 1. For a Guggenheim solution, set dimensionless parameters “a0”, “a1”, and “a2” or a temperature expansion for their J/mol counterparts using coefficients “p1”-“p9”, each of which defaults to zero. For the other nonideal models, similarly, set coefficients “p1”, “p2”, etc., as appropriate. For more information, see **Solid solutions** under **Configuring the programs** in the **GWB Essentials Guide** and the **Thermo Datasets** chapter in the **GWB Reference Manual**. You can as well choose whether the solid solution is to be treated as continuous, the default option, or as a sequence of “discrete” minerals.

You need to specify two end members for a solid solution, each associated with an abbreviated “tag”. The tag cannot be the name of another entry in the database. You can set a composition range in terms of the mole fraction of the most recently referenced end member, using keywords “from” and “to”, as well as a compositional “step” separating tranches of a discrete solution; by default, composition ranges from zero to one and the step is one-twentieth of the range. Keyword “contains” may be inserted for clarity. Examples:

```

solid_solution name = my_ss Calcite contains Strontianite from 0% to 10% a0= 0.05
solid_solution my_ss Calcite Strontianite from 0 to .1 a0= 0.05
solid_solution Biotite site_mixing Annite Phlogopite bsite= 3

```

If you don’t specify a solid solution’s name, the app will create one automatically using the tags for the two end members, separated by a hyphen. If you don’t set a tag for an end member, the app will pick up the first six characters of the corresponding mineral’s name. For example, the command

```

solid_solution Calcite contains Strontianite

```

produces a solid solution named “Calcit-Stront”, whereas typing

```
solid_solution Calcite tag Cc contains Strontianite tag Str
```

yields the name “Cc-Str”. To redefine the properties of a solid solution, enter the name followed by the properties to be changed:

```
solid_solution my_ss a0 = 1 a1 = 2
```

(You may prefer to use the “alter” command to change solutions defined within the thermo database; the two commands are functionally equivalent.) The command

```
solid_solution remove my_ss
```

discards a previously defined solution.

## 5.66 sorbate

```
sorbate <exclude | include>
```

**Config** → **Options...**

Use the “sorbate” command to tell the program, when considering sorption onto surfaces (see the “surface\_data” command), whether to include or exclude sorbed species in figuring the composition of the initial system. By default, the program does not include sorbed species in this calculation. If you set the  $\text{Ca}^{++}$  concentration to 15 mg/kg, for example, the initial system would contain that amount in the fluid and an additional amount sorbed onto mineral surfaces. If you type the command “sorbate include”, however, that amount would apply to the sum of the  $\text{Ca}^{++}$  sorbed and in solution.

## 5.67 span

```
span <value | min | ?> <unit> <to> <value | max | ?> <unit>  
span <off | on | reset>
```

**Config** → **Options...**

The “span” command lets you set a temperature range across which stability must be defined for each species considered in the calculation. The lower limit may be identified with keyword “from” or “Tmin”, and the upper denoted by “to” or “Tmax”. A value of “?” unsets the bound in question.

With  $T$ -table datasets, simply setting “span = on” without specifying a range forces the program to load species’ log  $K$ s in polynomial mode. In other words, it will

take log  $K$ s from polynomial fits against temperature to the stability constants in the thermo dataset, even if the calculation is performed entirely at one of the dataset's principal temperatures. Otherwise, an isothermal calculation run at one of the principal temperatures (commonly 0°C, 25°C, 60°C, ...) takes log  $K$ s for aqueous species, minerals, and so on directly from the values given in the thermo dataset.

Temperature ranges of validity can also be set in the thermo dataset for the virial coefficients used to calculate "Pitzer" and SIT activity coefficients. The "span" command works in a similar manner in such cases, loading only virial coefficients with validity that spans the calculation's temperature range.

As an example of using the "span" command, you may wish to compare results calculated at 25°C with those figured at 30°C. Since log  $K$  for a number of species in the thermo dataset is given at only 25°C, unless the "extrap" option is set, the former calculation may well take into account a somewhat larger set of species than the latter. Entering the command

```
span 25 C to 30 C
```

assures the same set of species is loaded in each case, and that both runs take log  $K$ s from polynomial fits.

Set the bounds to specific temperatures, or to "min" or "max", which respectively represent the lowest and highest temperatures considered in the thermo dataset, as loaded at run time. If you set the temperature range directly, it will be bracketed to that of the thermo dataset. For example, the command

```
span -1000 C to 1000 C
```

is functionally the same as

```
span 0 C to 300 C
```

when the "thermo.tdat" dataset is loaded, since the dataset's range is 0°C to 300°C.

Keyword "off" disables the feature, leaving values for the limits intact, and keyword "on" re-enables the feature. The "reset" keyword clears the limits, disabling the feature. By default, the span feature is disabled.

### 5.68 suffix

```
suffix <string>
```

```
Config → Output...
```

Use the "suffix" command to alter the names of the output datasets ("SpecE8\_output.txt", "SpecE8\_isotope.gtp", and "SpecE8\_plot.gtp") by adding a trailing string. Example:

```
suffix _run2
```

produces output datasets with names such as “SpecE8\_output\_run2.txt”.

## 5.69 sulfur-34

```
sulfur-34 <fluid | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “sulfur-34” command (also, “34-S”) to set the  $^{34}\text{S}$  isotopic composition of the fluid or any segregated minerals. The composition may be set on any scale (e.g., CDT), but you must be consistent throughout the calculation. Example:

```
sulfur-34 fluid = +45, Gypsum = +2
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
sulfur-34 remove
sulfur-34 off
```

clear all settings for  $^{34}\text{S}$  isotopes from the calculation.

See also the “<isotope>” section above, and the “carbon-13”, “hydrogen-2”, and “oxygen-18” commands.

## 5.70 suppress

```
suppress <species, surface species>
```

**Config** → **Suppress...**

Use the “suppress” command (also: “kill”) to prevent certain aqueous species or surface species from being considered in a calculation. Example:

```
suppress H3SiO4- >(s)FeOPb+
```

prevents the entries listed from being loaded from the database. The “unsuppress” command reverses the process.

## 5.71 surface\_capacitance

```
surface_capacitance = <value | ?>
surface_capacitance C1 = <value | ?> C2 = <value | ?>
surface_capacitance on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant capacitance  ▼ F/m<sup>2</sup>  
 or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_capacitance”) to set, in units of F/m<sup>2</sup>, the capacitance or capacitances of a sorbing surface.

For a two-layer complexation model, when you set a capacitance with the command, or if a value for capacitance is set in the header section of the surface reaction dataset, **SpecE8** will model surface complexation for the surface in question using the constant capacitance model, rather than the full two-layer model.

A triple-layer or CD-MUSIC complexation model, on the other hand, requires two capacitances,  $C_1$  and  $C_2$ . Default capacitances are set in the header of the surface dataset, but you may use the “C1” and “C2” arguments to override the default settings from the command line.

If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_capacitance on HFO = 2
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface.

## 5.72 surface\_data

```
surface_data <sorption dataset>
surface_data remove <sorption dataset | surface type>
surface_data OFF
```

**File** → **Open** → **Sorbing Surfaces...** →

**File** → **Open** → **Sorbing Surfaces...** → *select surface(s)* →

or **Config** → **Sorbing Surfaces...**

Use the “surface\_data” command (abbrev.: “surf\_data”) to specify an input dataset of surface sorption reactions to be considered in the calculation. The dataset name should be enclosed in quotes if it contains any unusual characters. Use the “remove” argument to eliminate a surface dataset, specified by name or surface type (e.g.,

“HFO”), from consideration. The argument “OFF” disables consideration of all surface complexes.

You can specify more than one sorbing surface in a model by repeating the “surface\_data” command for different datasets (a dataset of surface reactions for sorption onto hydrous ferric oxide, several triple-layer and CD-MUSIC datasets, as well as example datasets for the ion exchange,  $K_d$ , Freundlich, and Langmuir models are distributed with the software). To remove a dataset of surface reactions from consideration, you use commands such as

```
surface_data remove FeOH.sdat
surface_data remove HFO
surface_data OFF
```

The latter command removes all of the surface datasets that have been loaded.

### 5.73 surface\_potential

```
surface_potential = <value | ?>
surface_potential on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant potential  ▾ mV  
or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_potential”) to set, in units of mV, the electrical potential for a two-layer sorbing surface. When you set this value (or if a value is set in the header section of the surface reaction dataset), **SpecE8** will model surface complexation for the surface in question using the constant potential method, rather than invoking the full two-layer model.

If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_potential on HFO = 0
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface.

### 5.74 swap

```
swap <new basis> <for> <basis species>
```

**Basis** → ↺

## GWB Command Reference

---

Use the “swap” command to change the set of basis entries. All reactions are written internally in terms of a set of basis species that you can alter to constrain the composition of the initial system. An aqueous species, mineral, gas, or activity ratio can be swapped into the basis in place of one of the original basis species listed in the database. Examples:

```
swap CO3--      for HCO3-
swap Quartz     for SiO2(aq)
swap CO2(g)     for H+
swap O2(g)      for O2(aq)
swap Ca++/Na+^2 for Ca++
```

Each end member of a continuous solid solution can similarly be swapped into the basis. Use the syntax <solid solution>::<end member tag>. For example, the sodium end member Albite (tag “Ab”) of the plagioclase feldspar solid solution (name “Plag”) can be swapped into the basis simultaneously with the calcium end member Anorthite (tag “An”) as follows:

```
swap Plag::Ab for Na+
swap Plag::An for Al+++
```

The new species must contain in its composition the original basis species being swapped out (you can’t swap lead for gold). For example, CO<sub>2</sub>(g) is composed of HCO<sub>3</sub><sup>-</sup>, H<sup>+</sup>, and water in “thermo.tdat”. The reactions in the thermo dataset (once reduced to the set of basis and redox species and modified to reflect enabled redox couples) show the basis entries for which a species may be swapped. For a list of original basis species, type “show basis”. To reverse a swap, type “unswap <species>”.

### 5.75 system

```
system <system command>
```

Use the “system” command to execute a command on the computer’s operating system (a “DOS command”) without leaving **SpecE8**. Example:

```
system "copy SpecE8_output.txt SpecE8_output2.txt"
```

Enclose multiword system commands in quotes. An alternative way to fork a system command is to begin the **SpecE8** command with a “\$”:

```
$copy SpecE8_output.txt SpecE8_output2.txt
```

In this case, no quotation marks are needed.



When a “system” command is executed, a “Command Prompt” window will appear briefly on your screen. Due to limitations of the Windows operating system, you cannot fork a command that requires user input, and you will not be able to see any output (including error messages) that might be generated by the command.

## 5.76 TDS

**TDS** = <value | ?>

**Basis** →  advanced → TDS  ▼ mg/kg

Use the “TDS” command to set in mg/kg the total dissolved solids for the initial fluid, if you don’t want the program to calculate this value automatically. The program uses the TDS when needed to convert input constraints into molal units.

To restore automatic calculation of the TDS, type the command with no argument or with an argument of “?”. To see the variable’s current setting, type “show variables”.

## 5.77 temperature

**temperature** = <value> <unit>

**Basis** → temperature  ▼ C ▼

Use the “temperature” command (also: “T”) to set the temperature of the system. The **Units Recognized** chapter in the **GWB Reference Manual** lists possible units, which default to “C”. Examples:

temperature 300 C  
T = 473 K

Temperature values can range over the span of the thermo dataset, from 0°C to 300°C for “thermo.tdat”; 25°C is the default.

## 5.78 timax

**timax** = <value | ?>

**Config** → **Iteration...**

The “timax” command sets in molal units the maximum value of ionic strength used in calculating species’ activity coefficients when Debye-Hückel methods are employed. The limiting value is also used when water activity is calculated according to Wolery

(1992b). By default, this variable is set to 3 molal. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 5.79 title

**title** <character string>

**Config** → **Output...**

Use the “title” command to set a title to be passed to the graphics program. Example:

```
title "Yucca Mountain Groundwater"
```

Be sure to put multiword titles in quotes.

## 5.80 unalter

**unalter** <species | mineral | solid solution | gas | surface species | **ALL**>

**Config** → **Alter Log Ks...** → *select entry(ies)* →

Use the “unalter” command to reverse the effect of having changed the log  $K$ ’s for a species, mineral, gas, or surface reaction, the selectivity coefficient for an exchange species, the  $K_d$  for a sorbed species, the  $K_f$  and  $n_f$  for a Freundlich species, or the properties of a solid solution in the current dataset. Example:

```
unalter Quartz
```

In this case, the log  $K$  values for quartz revert to those in the current thermodynamic dataset. The argument “ALL” resets the log  $K$ ’s, selectivity coefficients,  $K_d$ ’s, or  $K_f$ ’s and  $n_f$ ’s for all species, minerals, gases, and surface species, as well as the properties of all solid solutions in the dataset.

## 5.81 unsegregate

**unsegregate** <mineral(s) | **ALL**>

**Config** → **Isotopes...** → *select entry(ies)* →

Use the “unsegregate” command to remove minerals from the list of minerals to be segregated isotopically.

## 5.82 unsuppress

**unsuppress** <species, surface species | **ALL**>

**Config** → **Suppress...**

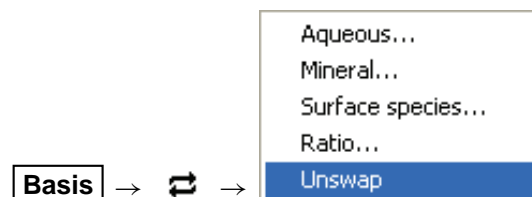
Use the “unsuppress” command (also: “include”) to include in the calculation aqueous species or surface species that have previously been suppressed. Examples:

```
unsuppress H3SiO4- >(s)FeOPb+
unsuppress ALL
```

The argument “ALL” clears any species that have been suppressed.

## 5.83 unswap

**unswap** <species | **ALL**>



Use the “unswap” command to reverse a basis swap. Example:

```
unswap Quartz (or unswap SiO2(aq))
```

to reverse the effect of the previous command

```
swap Quartz for SiO2(aq)
```

At this point, SiO<sub>2</sub>(aq) is back in the basis. The “ALL” argument reverses all basis swaps.

## 5.84 usgovt

**usgovt**

**Help** → **About SpecE8...**

Display the restricted rights legend that governs use, duplication, or disclosure by the U.S. Government.

## 5.85 volume

**volume** = <value | ?> <unit>

**Medium** → bulk volume  ▼ cm3 ▼

Use the “volume” (also “bulk\_volume”) command to cause the program to scale the system to a specific size. By default, the program does not scale the system. In this case, the bulk volume  $V_b$

$$V_b = V_f + V_{mn} + V_{in} + V_{stag} \quad (5.1)$$

is computed as the sum of the fluid volume  $V_f$ , mineral volume  $V_{mn}$ , inert volume  $V_{in}$ , and the volume  $V_{stag}$  of the stagnant zone, if one has been specified using the “dual\_porosity” command.

If  $X_{stag}$  is the stagnant fraction of the system, which is zero if a dual porosity model has not been invoked, the above equation can be rewritten

$$(1 - X_{stag}) V_b = V_f + V_{mn} + V_{in} \quad (5.2)$$

When you use the “volume” command to set the bulk volume  $V_b$ , the program scales the fluid volume  $V_f$  and optionally the mineral and inert volumes,  $V_{mn}$  and  $V_{in}$ , so the system fills the specified space.

The command

volume = 2 m3

for example, causes the program to scale the fluid volume  $V_f$  and optionally the mineral and inert volumes,  $V_{mn}$  and  $V_{in}$ , so the system occupies 2 m<sup>3</sup>. Default behavior, in which no scaling is performed, can be restored at any time with the command

volume = ?

The unit you use to constrain the amount of each mineral in the system controls whether or not the program adjusts that mineral’s volume. If you constrain Quartz to “10 cm<sup>3</sup>”, for example, Quartz will occupy 10 cm<sup>3</sup> after scaling. Setting Quartz in relative units such “volume%” or “mmol/m<sup>3</sup>”, in contrast, causes the program to fix the mineral’s volume relative to  $V_b$ .

Similarly if you set inert volume in an absolute unit, such as

inert = 50 cm3

the program will honor that constraint and hold  $V_{in}$  steady. The inert portion of the system in this example will occupy 50 cm<sup>3</sup> after scaling. The command

```
inert = 10 volume%
```

however, causes the program to scale the inert volume to 10% of the system's bulk volume.



# React Commands

## 6.1 <unit>

<value> <free> <unit> <as element symbol> <basis entry>

**Basis** →  ▼ free kg ▼

To constrain the initial system, enter a command containing only the above entries. Entries may appear in any order. The qualifier “free” specifies that the constraint applies to the free rather than to the bulk basis entry. **React** recognizes the following units for constraining the initial system:

*By mass or volume:*

mol	mmol	umol	nmol	
kg	g	mg	ug	ng
eq	meq	ueq	neq	
cm <sup>3</sup>	m <sup>3</sup>	km <sup>3</sup>	l	

*By concentration:*

mol/kg	mmol/kg	umol/kg	nmol/kg	
molal	mmolal	umolal	nmolal	
mol/l	mmol/l	umol/l	nmol/l	
g/kg	mg/kg	ug/kg	ng/kg	
wt%	"wt fraction"			
g/l	mg/l	ug/l	ng/l	
eq/kg	meq/kg	ueq/kg	neq/kg	
eq/l	meq/l	ueq/l	neq/l	

*By carbonate alkalinity (applied to bicarbonate component):*

eq_acid	meq_acid	ueq_acid	neq_acid	
eq_acid/kg	meq_acid/kg	ueq_acid/kg	neq_acid/kg	
eq_acid/l	meq_acid/l	ueq_acid/l	neq_acid/l	
g/kg_as_CaCO <sub>3</sub>	mg/kg_as_CaCO <sub>3</sub>	ug/kg_as_CaCO <sub>3</sub>	ng/kg_as_CaCO <sub>3</sub>	
wt%_as_CaCO <sub>3</sub>				
g/l_as_CaCO <sub>3</sub>	mg/l_as_CaCO <sub>3</sub>	ug/l_as_CaCO <sub>3</sub>	ng/l_as_CaCO <sub>3</sub>	
mol/kg_as_Ca...	mmol/kg_as_Ca...	umol/kg_as_Ca...	nmol/kg_as_CaCO <sub>3</sub>	
mol/l_as_CaCO <sub>3</sub>	mmol/l_as_CaCO <sub>3</sub>	umol/l_as_CaCO <sub>3</sub>	nmol/l_as_CaCO <sub>3</sub>	

*Per volume of the system:*

## GWB Command Reference

mol/cm3	mmol/cm3	umol/cm3	nmol/cm3
kg/cm3	g/cm3	mg/cm3	ug/cm3
ng/cm3			
mol/m3	mmol/m3	umol/m3	nmol/m3
kg/m3	g/m3	mg/m3	ug/m3
ng/m3			
volume%	"vol. fract."		
<i>By activity:</i>			
activity	fugacity	ratio	
pH	V	pe	
<i>By partial pressure:</i>			
Pa	MPa	atm	bar
psi			

Activity and fugacity may be abbreviated to "a" or "f". Keyword "total" reverses a setting of "free".

Use the "as" keyword to constrain mass in terms of elemental equivalents. For example, the command

```
CH3COO- = 10 umol/kg as C
```

specifies 5 umol/kg of acetate ion, since each acetate contains two carbons, whereas

```
20 mg/kg SO4-- as S
```

would specify 59.9 mg/kg of sulfate, since the ion's mole weight is about 3 times that of sulfur itself.

Examples:

```
55 mg/kg HCO3-
Na+ = 1 molal
1 ug/kg U++++
100 free cm3 Dolomite
pH = 8
Eh = .550 V
log f O2(g) = -60
HCO3- = 30 mg/kg as C
```

You may in a similar fashion constrain the concentration of a kinetic aqueous or surface complex. For example, the commands

```
kinetic AIF++
AIF++ = 1 umol/kg
```

set the concentration of the kinetic complex AIF<sup>++</sup> to 1  $\mu\text{mol kg}^{-1}$ .



## 6.2 <isotope>

```
<isotope | symbol> <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the name of any isotope system loaded in the isotope dataset (also, the isotope's symbol) to set the isotopic composition of the initial fluid, reactant species (aqueous species, minerals, end members, gases, or oxides) or segregated minerals. The composition may be set on any scale (e.g., SMOW, PDB, ...), but you must be consistent throughout the calculation.

For example, if the  $^{17}\text{O}$  isotope system has been added to the isotope dataset, you could enter:

```
oxygen-17 fluid = -10, Quartz = +15
or
17-O fluid = -10, Quartz = +15
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
oxygen-17 remove
oxygen-17 off
```

clear all settings for  $^{17}\text{O}$  isotopes from the calculation.

## 6.3 activity

```
activity <species> = <value | ?>
```

**Basis** →  ▼ activity ▼

Use the “activity” command (abbrev.: “a”) to constrain the activity of an aqueous species in the initial system. Examples:

```
activity Na+ = 0.3
log a H+ = -5
```

See also the “pH”, “Eh”, “pe”, “ratio”, “fugacity”, “fix”, and “slide” commands.

## 6.4 add

**add** <basis species>

**Basis** → add

Use the “add” command to include a basis species in the calculation. Example:

add HCO3-

See also the “swap”, “activity”, “fugacity”, “pH”, “pe”, and “Eh” commands.

## 6.5 alkalinity

**alkalinity** = <value> <unit>

**Basis** → HCO3-  ▼ mg/kg\_as\_CaCO3 ▼

Use the “alkalinity” command to constrain the total concentration of  $\text{HCO}_3^-$  to reflect the solution’s carbonate alkalinity. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “mg/kg\_as\_CaCO3” is the default. To use this option, the solution pH must be set explicitly.

## 6.6 alter

**alter** <species | mineral | gas> <log  $K$ s>

**alter** <species | mineral | gas> <poly coefs> **TminK** = <value> **TmaxK** = <value>

**alter** <surface species> <poly coefs> **TminK** = <value> **TmaxK** = <value>

**alter** <surface species> **logK** = <value> **dlogK/dT** = <value>

**alter** <exchange species> **beta** = <value>

**alter** <sorbed species> **Kd** = <value>

**alter** <sorbed species> **Kf** = <value> **nf** = <value>

**alter** <solid solution> <type> <**discrete** | **continuous**> \

<from <value> to <value> **step** <value>> <<variable> = <value>>

**Config** → **Alter Log Ks...**

**Config** → **Solid Solutions...**

Use the “alter” command to change the temperature expansion for a species’, mineral’s, or gas’ log  $K$ , to adjust the stability of a surface species, or to change the properties of a solid solution.

Temperature expansions for log  $K$ s are given in the thermo dataset either as  $T$ -tables, or by up to six coefficients of a polynomial. When the current thermo dataset uses

*T*-tables, you list replacement values at each of the principal temperatures specified in the dataset, most commonly 0°C, 25°C, 60°C, 100°C, 150°C, 200°C, 250°C, and 300°C. Example:

```
alter Anhydrite -4.3009 -4.4199 -4.7126 -5.1758 -6.2299 500 500 500
```

Values of “500” represent a lack of data at the corresponding temperature.

For a thermo dataset constructed from polynomial expansions, on the other hand, list instead up to six polynomial coefficients

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4
```

You can optionally append a temperature range of validity, in Kelvins, for the polynomial:

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4 TminK= 293 TmaxK= 383
```

Absent a range, the polynomial is taken to span the principal temperatures. You may adjust a reaction’s temperature range without specifying polynomial coefficients, but setting only coefficients resets the temperature range.

For a surface complex, use the “alter” command to set  $\log K$ , its temperature derivative, or both

```
alter >(w)FeOCa+ logK = 6.0 dlogK/dT = 0.02
```

whereas for an ion exchange reaction, you set the selectivity coefficient  $\beta$

```
alter >X2:Ca beta = .033
```

directly, rather than as a logarithm. For a sorbing species, set  $K_d$  or, for a Freundlich species,  $K_f$ ,  $n_f$ , or both, as follows

```
alter >Pb++ Kd = .03
alter >Sr++ Kf = .015 nf = .8
```

For a solid solution, use the “alter” command to set the activity coefficient model, discrete or continuous behavior, the composition range in terms of the mole fraction of the most recently specified end member, and parameters for the chosen activity model, as described for the “solid\_solution” command. A solid solution “my\_ss” might be set as a subregular Guggenheim solution with the command

```
alter my_ss guggenheim a0 = 1 a1 = 2
```

You may prefer to use the “alter” command to adjust an entry from the thermo database, and the “solid\_solution” command to reconfigure a user-defined solution, but the two commands work interchangeably for this purpose.

Type “show alter” to list altered species or solutions and their settings; the “unalter” command reverses the process.

## 6.7 audit

**audit** <off>

**Config** → **Stepping...**

The program performs a global mass balance audit over the course of each simulation. By default, the program is silent unless it detects a mass imbalance, in which case it writes a warning message to the **Results** pane, as well as the bottom of the print-format output, if the print output stream is open.

You can use the “audit” option to cause the program to report at the end of the simulation the complete results of the mass balance audit for each original basis entry.

To turn audit reporting on and off, use:

```
audit
audit off
```

## 6.8 b-dot

**b-dot**

**File** → **Open** → **Thermo Data...** → thermo.tdat

The “b-dot” command (formerly: “debye-huckel”) causes the program to read the “thermo.tdat” thermodynamic dataset, which invokes the “B-dot” form of the extended Debye-Hückel equation to calculate activity coefficients for aqueous species.

## 6.9 balance

**balance** <on> <basis entry>

**balance** <off>

**Basis** →  ▼ unit ▼ → Balance species

Use the “balance” command to specify the basis entry to be used for electrical charge balancing. The basis entry must be a charged aqueous species. By default, **React** balances on  $\text{Cl}^-$ .

The command “balance off” disables **React’s** charge balancing feature. In this case, the user is responsible for prescribing charge-balanced input constraints.

## 6.10 carbon-13

```
carbon-13 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “carbon-13” command (also, “13-C”) to set the  $^{13}\text{C}$  isotopic composition of the initial fluid, reactant species (aqueous species, minerals, end members, gases, or oxides) or segregated minerals. The composition may be set on any scale (e.g., PDB), but you must be consistent throughout the calculation. Example:

```
carbon-13 fluid = -10, Calcite = +4
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
carbon-13 remove
carbon-13 off
```

clear all settings for  $^{13}\text{C}$  isotopes from the calculation.

See also the “<isotope>” section above, and the “hydrogen-2”, “oxygen-18”, and “sulfur-34” commands.

## 6.11 chdir

```
chdir <directory> <ALL>
```

**File** → **Working Directory...**

Use the “chdir” command (abbrev.: “work\_dir”, “cd”) to change the working directory. The program reads input scripts relative to the current working directory and writes output into it. Typing the command “chdir” without an argument causes the program to display the name of the working directory. The “ALL” argument changes the working directories of all of the GWB apps. The command

```
chdir ~
```

changes to the user’s home directory, if one is defined by the operating system.

## 6.12 clear

**clear**

**View → Clear Screen**

Use the “clear” command (abbrev.: “c”) or right-click in the text window and choose **Clear Screen** to clear it.

## 6.13 clipboard

**clipboard**

**Edit → Copy**

The “clipboard” command causes the contents of the **Results** pane to be copied to the clipboard, from where it can be pasted into various types of documents.

## 6.14 conductivity

**conductivity** <conductivity dataset>

**File → Open → Conductivity Data...**

Use the “conductivity” command to change the input file of coefficients used to calculate electrical conductivity. Example:

```
conductivity "..\my_conductivity.dat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. Beginning with GWB11, the applications compute electrical conductivity using either of two different approaches, the USGS and APHA methods; the USGS method is the default. The required coefficients are defined in the files “conductivity-USGS.dat” and “conductivity-APHA.dat”, respectively, which are installed in the same directory as the thermo datasets (commonly “\Program Files\GWB\Gtdata”).

## 6.15 couple

**couple** <redox species | element(s) | **ALL**>

**Config → Redox Couples... → *select couple(s)* → **select with...****

Use the “couple” command to enable any redox coupling reactions that have been disabled with the “decouple” command. You specify one or more redox species or elements. For example, the command

```
couple Carbon
```

ouples all redox reactions involving the element carbon. Argument “ALL” enables all of the coupling reactions in the thermo dataset.

## 6.16 cpr

```
cpr = <field variable | ?> <unit> <steady | transient>
```

**Medium** → cpr (minerals)  +  cal/g/C ▼

Use the “cpr” command to set the heat capacity of the rock (mineral) framework. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/g°C” is the default. This value is used during polythermal mixing simulations. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function.

By default, this variable is set to 0.2 cal/g°C. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.17 cpu\_max

```
cpu_max = <value | ?>
```

**Config** → **Stepping...**

Use the “cpu\_max” command to limit the amount of computing time a simulation may take. You set the maximum computing time in seconds, or use a “?” to restore the default state, which is no prescribed limit. To see the current setting, type “show variables”.

## 6.18 cpw

```
cpw = <field variable | ?> <unit> <steady | transient>
```

**Medium** → cpw (fluid)  +  cal/g/C ▼

Use the “cpw” command to set in cal/g°C the heat capacity of the fluid. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/g°C” is the default. This value is used during polythermal mixing simulations. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function.

By default, this variable is set to 1.0 cal/g°C. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

### 6.19 data

```
data <thermo dataset> <verify>
```

**File** → **Open** → **Thermo Data...**

Use the “data” command to change the input file of thermodynamic data. Example:

```
data "..\my_thermo.tdat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. The “verify” option causes the program to read the named dataset only if it has not already been read.

### 6.20 decouple

```
decouple <redox species | element(s) | ALL>
```

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “decouple” command to disable the coupling reactions for one or more redox species, in order to calculate a model assuming redox disequilibrium. The redox species then become available for use as basis species and may be constrained independently of the original basis entries. You can disable as many coupling reactions as you want.

You specify either one or more redox species or elements. For example, the command

```
decouple Carbon
```

decouples all redox reactions involving the element carbon. Argument “ALL” disables all of the coupling reactions in the thermo dataset. Use the “couple” command to enable coupling reactions, once they have been disabled.



## 6.21 delQ

```
delQ = <value | ?>
```

**Config** → **Stepping...**

Use the “delQ” command to control the lengths of time steps taken in a simulation accounting for reaction kinetics. The program limits how much the ion activity product  $Q$  can change over a step, for each kinetic reaction considered. The setting for “delQ” is the projected change  $\Delta Q/Q$  allowed in the relative value of the activity product. You can set a larger value to permit longer time steps, or a smaller value to improve stability. The default setting is 0.1. Type the command with no argument or with an argument of “?” to restore the default. To see the current setting, type “show variables”.

## 6.22 delxi

```
delxi = <value | ?> <linear | log>
```

**Config** → **Stepping...**

Use the “delxi” command to set the maximum length (in terms of reaction progress, which varies from zero to one over the course of the simulation) of the reaction step, and to specify reaction stepping on a linear or logarithmic scale.

The command

```
delxi = .1 linear
```

for example, causes the program, in the absence of other constraints on the reaction step, to take steps through reaction progress of .1, .2, .3, ..., 1.0. Alternatively, the commands

```
dx_init = .001
delxi = .5 log
```

produce a path with steps .001, .003, .01, .03, .1, .3, and 1.0 (see the “dx\_init” command).

By default, this variable is set to 0.01 and “linear”. To restore the default settings, type the command with no argument or with an argument of “?”. To see the current settings, type “show variables”.

## 6.23 density

```
density = <value | ?>
density = <TDS | chlorinity>
density = <batzle-wang | phillips>
```

**Basis** →  advanced → density  ▼ g/cm<sup>3</sup>  
**Config** → **Options...**

You can use the “density” command to set in g/cm<sup>3</sup> the fluid density the program uses to convert compositional constraints to molality, the concentration unit it carries internally. If you set the initial Na<sup>+</sup> composition in mg/l, for example, the program needs to know the density of the initial fluid to determine Na<sup>+</sup> molality.

The program by default converts units using a density value it calculates automatically, as discussed below. This value is sufficient for most purposes, and hence it is generally not necessary to set fluid density explicitly. You might, however, want to set the density if you are working at high temperature, but your analysis is expressed per liter of solution at room temperature.

You can also use the “density” command to tell the program how to calculate the default density it uses to convert units, and the fluid density it reports in the simulation results. The program by default uses the Batzle-Wang equation to figure density, but you can use the command “density = phillips” to select the method of Phillips et al., instead.

As well, the program normally figures density as that of an NaCl solution with the same TDS as the fluid in question, at the temperature of interest. With the command “density = chlorinity” you can tell the program to instead use the density of an NaCl solution of equivalent chlorinity.

To restore automatic calculation, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.24 dual\_porosity

```

dual_porosity = <on | off> <spheres | blocks | fractures> \
  <geometry = <spheres | blocks | fractures | ?>> \
  <volfrac = <field variable | ?>> \
  <Nsubnode = <value | ?>> \
  <<radius | half-width> = <field variable | ?> <unit>> \
  <diff_length = <field variable | ?> <unit>> \
  <porosity = <field variable | ?>> \
  <retardation = <field variable | ?>> <steady | transient> \
  <diff_coef = <field variable | ?> <unit>> <steady | transient> \
  <thermal_con = <field variable | ?> <unit>> <steady | transient> \
  <theta = <value | ?>> <reset | ?>

```

**Config** → **Dual Porosity...**

Use the “dual\_porosity” command (abbrev: “dual”) to configure stagnant zones in the simulation, using the dual porosity feature. Enable and disable the feature with the “on” and “off” keywords. Disabling the feature does not affect other settings, so re-enabling the feature returns the model to its most recent configuration.

With the “geometry” keyword, you configure the stagnant zone into spheres, blocks, or a fractured domain, the latter being slabs separated by fractures arrayed along the  $x$  direction. Alternatively, you can set the three configurations directly with keywords “spheres”, “blocks”, and “fractures”. Keyword “volfrac” sets the fraction of the nodal block’s bulk volume occupied by the stagnant zone.

The “Nsubnode” (or “nx”) keyword sets the number of nodes into which the stagnant zone within each node will be divided when solving for solute and temperature distributions. The “radius” (or “half-width”, for blocks and fractures) keyword sets the zone’s characteristic dimension, in units of distance (see the **Units Recognized** chapter in the **GWB Reference Manual**; default is cm), and keyword “diff\_length” sets the distance (same units) from the contact with the free-flowing zone over which the model will account for solute diffusion and heat conduction. Use keywords “porosity”, “retardation”, “diff\_coef”, and “thermal\_con” to set values for the porosity, retardation factor, diffusion coefficient, and thermal conductivity of the stagnant zone. The default unit for the diffusion coefficient is  $\text{cm}^2/\text{s}$ , and thermal conductivity defaults to  $\text{cal}/\text{cm}/\text{s}/^\circ\text{C}$  units; see the **Units Recognized** chapter for a list of options.

By default, the stagnant zone is configured in spheres divided into 5 subnodes. You must specify a value for the volume fraction of the stagnant zone, as well as one for the radius (or half-width); the diffusion length defaults to the latter value. The program uses whatever values are set for the free-flowing zone in the node in question as default values for the porosity, diffusion coefficient, and thermal conductivity of the stagnant zone; the retardation factor defaults to a value of one.

The “theta” keyword sets time weighting ( $0 \leq \theta \leq 1$ ) for the numerical solution of diffusive transport within the stagnant zone. A weight  $\theta = 0$  assigns the explicit method, and larger values invoke an implicit solution. By default, the program chooses  $\theta$

automatically, using the explicit method unless it would force too many more time steps than would otherwise be necessary. In that case, the program uses the implicit method ( $\theta = 0.6$ ), which requires more computing effort per time step, but can take long steps without becoming numerically unstable.

You can append the “transient” keyword when setting several of the parameters: the diffusion coefficient, thermal conductivity, and retardation factor. If the variable is defined by an equation, script, or external function, it will then be re-evaluated continuously over the course of the run.

As an example, the command

```
dual_porosity geometry = spheres, radius = 50 cm, volfrac = 75%
```

configures the stagnant zone into spheres of half-meter radius that occupy three-quarters of the domain. The command

```
dual_porosity reset
```

enables the feature after restoring default settings for each keyword.

### 6.25 dump

```
dump <off>
```

**Config** → **Stepping...**

Use the “dump” command to eliminate minerals present at the beginning of the reaction before the program begins to trace the path. Use

```
dump  
dump off
```

### 6.26 dx\_init

```
dx_init = <value | ?>
```

**Config** → **Stepping...**

Use the “dx\_init” command to set the length of the initial time step. You set this value in terms of reaction progress, which varies from zero to one over the course of the simulation.

By default, the variable is ignored. To restore the default state, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.27 dxplot

**dxplot** = <value | ?>

**Config** → **Output...**

Use the “dxplot” command to set the interval in reaction progress (which varies from zero to one over the course of the simulation) between entries in the “React\_plot.gtp” dataset. A value of zero causes the program to write the results after each step in reaction progress. This variable setting does not apply to reactions paths with logarithmic reaction stepping (see the “delxi” command), in which case all points in reaction progress are written to the plot dataset.

By default, this variable is set to 0.005. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.28 dxprint

**dxprint** = <value | ?>

**Config** → **Output...**

Use the “dxprint” command to set the interval in reaction progress (which varies from zero to one over the course of the simulation) between entries in the “React\_output.txt” dataset. A value of zero causes the program to write the results after each step in reaction progress, which can produce large amounts of output. This variable setting does not apply to reactions paths with logarithmic reaction stepping (see the “delxi” command), in which case all points in reaction progress are written to the plot dataset.

By default, this variable is set to 0.01. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.29 Eh

**Eh** = <value | ?> <unit>

**Basis** →  ▼ V ▼

Use the “Eh” command to set Eh in the initial system. Example:

Eh = 200 mV

sets the system’s oxidation state to correspond to an Eh of 0.2 volt. See also the “activity”, “pH”, “pe”, “fugacity”, “fix”, and “slide” commands.

## 6.30 end-dump

**end-dump** <off>

**Config** → **Stepping...**

Use the “end-dump” command (also: “enddump”) to eliminate minerals present at the end of the reaction path after the path has been traced. Use

```
end-dump
end-dump off
```

This operation can also be accomplished using the command “pickup fluid”.

## 6.31 epsilon

**epsilon** = <value | ?>

**Config** → **Iteration...**

Use the “epsilon” command to set the convergence criterion (dimensionless) for iterating to a solution of the equations representing the distribution of chemical mass. By default, this variable is set to  $5 \times 10^{-11}$ . To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.32 exchange\_capacity

**exchange\_capacity** = <value | ?> <units>  
**exchange\_capacity on** <type> = <value | ?> <units>

**Config** → **Sorbing Surfaces...** → exchange capacity  ▼ eq ▼  
 or **File** → **Open** → **Sorbing Surfaces...**

Use the “exchange\_capacity” command (abbrev.: “ex\_capacity” or “exch\_capacity”) to set the exchange capacity (i.e., the CEC) of the system when modeling ion exchange reactions or sorption according to Langmuir isotherms. For ion exchange reactions, you set units of electrical equivalents (“eq”, “meq”, and so on) or equivalents per gram of dry sediment (“eq/g”, “meq/g”, ...). For Langmuir reactions, you similarly set a value in mole units: “mol”, “mmol”, “mol/g”, “mmol/g”. If you set units per gram of sediment, the program multiplies the value entered by the mass of rock in the system (including equilibrium and kinetic minerals as well as inert volume) to get the system’s capacity.

If you read in a surface that sorbs by ion exchange or Langmuir isotherms, you must set a value for its exchange capacity. If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example

```
exchange_capacity on IonEx = .0008 eq/g
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the surface represented by the sample dataset “IonEx.sdat”, for example, is “IonEx”. You can use the “show” command to display the “type” of each active surface. See also the “surface\_data” and “inert” commands.

### 6.33 explain

```
explain <species | mineral(s) | solid solution(s) | gas(es) | surface species>
```

**Config** → **Show...** → *select species*

Use the “explain” command to get more information (such as the mole weight of a species, a mineral’s formula, mole volume, and density, or a solid solution’s end-member minerals) about species, minerals, solid solutions, and gases in the dataset. Example:

```
explain Analcime
```

### 6.34 explain\_step

```
explain_step <off>
```

**Config** → **Stepping...**

The “explain\_step” option causes the program to report on the Results pane the factor controlling the length of each time step, whenever the step size is limited by the need to maintain numerical stability.

### 6.35 extrapolate

```
extrapolate <on | off>
```

**Config** → **Options...**

Use the “extrapolate” option to cause the program to extrapolate log  $K$ s for reactions forming aqueous species, minerals, and gases to temperatures beyond the data’s known range of validity. The option should be used with considerable care.

The temperature range of validity for a species’ log  $K$  is defined by its temperature expansion, taken from the thermo dataset. For  $T$ -table expansions, it is the span of principal temperatures at which log  $K$ s are not 500, whereas for polynomial expansions, the range is listed explicitly.

In normal operation, the program loads only species with ranges of validity encompassing the calculation’s temperature range. When the “extrapolate” option is “on”, in contrast, the program projects to the temperature of interest the log  $K$ s of species that would not otherwise be loaded.

Temperature ranges of validity can also be set in the thermo dataset for the virial coefficients used to calculate “Pitzer” and SIT activity coefficients. The “extrapolate on” option works in a similar manner in such cases, allowing virial coefficients to be used at temperatures beyond the coefficients’ known ranges of validity.

### 6.36 fix

**fix** <unit> <species | gas>

Use the “fix” command to hold the activity of a species, fugacity of a gas, or an activity ratio constant over the course of a run. The <unit> can be “activity” or “fugacity” (“a” or “f” for short), “ratio”, “pH”, “pe”, or “Eh”, or it can be omitted. Examples:

```
fix pH
fix a H+
fix f O2(g)
fix ratio Ca++/Na+^2
```

### 6.37 flash

**flash** <on | off | fluid | system>

**Config** → **Stepping...**



Use the “flash” command to set a “flash” model in which the original fluid (or fluid and minerals, if keyword “system” is used) is removed over the course of the reaction path. Keywords “on” and “fluid” are synonymous. The command “flash” without an argument is the same as “flash fluid”. Generally, the fluid is replaced by a reactant fluid. Use

```
flash
flash system
flash off
```

### 6.38 flow-through

```
flow-through <off>
```

**Config** → **Stepping...**

Use the “flow-through” command to turn on or off the flow-through option by which mineral precipitates are isolated from back-reaction. Use

```
flow-through
flow-through off
```

### 6.39 flush

```
flush <off>
```

**Config** → **Stepping...**

Use the “flush” command to turn on or off the flush option by which fluid reactants displace existing fluid from the system over the course of the reaction path. Use

```
flush
flush off
```

### 6.40 fugacity

```
fugacity <gas> = <value | ?>
```

**Basis** →  ▼ fugacity ▼

## GWB Command Reference

---

Use the “fugacity” command (abbrev.: “f”) to set gas fugacities (on an atm scale) in the initial system. Examples:

```
fugacity O2(g) = .2  
f CO2(g) = 0.0003  
log f S2(g) = -30
```

Use “?” to unset a fugacity value: See also the “activity”, “ratio”, “pH”, “Eh”, “pe”, “fix”, and “slide” commands.

### 6.41 go

```
go
```

```
Run → Go
```

```
Results → Run
```

Use the “go” command to initiate calculation of a model based on the current configuration.

### 6.42 grep

```
grep <string>
```

```
File → View → .\React_output.txt → Edit → Find
```

Use the “grep” command to search for each occurrence of the given character string in the current output dataset. For example, use

```
grep Albite
```

to obtain a list of each line in “React\_output.txt” (or the current output dataset, as modified by the “suffix” command) containing the string “Albite”.

### 6.43 gtpplot

```
gtpplot <gtpplot options>
```

```
Run → Gtpplot
```

The “gtpplot” command launches program **Gtplot** from the **React** command line, invoking input from the current graphics interface dataset (by default, “React\_plot.gtp”; see the “suffix” command). For example,

```
gtpplot -graph piper
```

starts **Gtplot** with a Piper diagram as the initial plot.

## 6.44 h-m-w

**h-m-w**

**File** → **Open** → **Thermo Data...** → **thermo\_hmw.tdat**

Use the “h-m-w” command (abbrev.: “hmw”) to set the program to calculate species’ activity coefficients by using the Harvie-Møller-Weare equations. Executing this command automatically sets the input dataset of thermodynamic data to “thermo\_hmw.tdat”. Note that dataset “thermo\_hmw.tdat” supports calculations at 25°C only.

## 6.45 heat\_source

```
heat_source = <field variable | ?> <unit> <steady | transient> \
              <Tmin = <value | min | ?> <unit>> <Tmax = <value | max | ?> <unit>>
heat_source = <off | on | reset>
```

**Medium** →  internal heat source   cal/cm<sup>3</sup>/s ▼  
 min  C ▼ max  C ▼

Use the “heat\_source” command (also: “heat\_src”) to set the rate of internal heat production within the medium. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/cm<sup>3</sup>/s” is the default. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function. By default, the program does not account for internal heat production. The “?” argument resets the default value of zero.

The “Tmin” and “Tmax” keywords (also: “temp\_max”, “temp\_min”) prescribe the allowable temperature range for the simulation. Set the keywords to specific temperatures, or to “min” or “max”, which respectively represent the lowest and highest temperatures considered in the thermo dataset, as loaded at run time. If you set the temperature range directly, it will be bracketed to that of thermo dataset. For example, the command

```
heat_source Tmin = -1000 C, Tmax = 1000 C
```

is functionally the same as

```
heat_source Tmin = 0 C, Tmax = 300 C
```

when the “thermo.tdat” dataset is loaded, since the dataset’s range is 0°C to 300°C.

The upper and lower temperature bounds serve two purposes. First, the simulation will give an error message and stop if temperature at any point in the domain falls more than 5°C less than the minimum value, or exceeds the maximum value by more than this amount. Second, unless the “extrap” option is set, the model will load for the simulation only those species for which log  $K$  values are available in the thermodynamic dataset over the specified temperature range.

The temperature bounds specified with the “Tmin” and “Tmax” keywords are the same as those set with the “span” command: The command

```
heat_source Tmin = 20 C, Tmax = 100 C
```

may be equivalently expressed

```
span 20 C to 100 C
```

as long as the heat source option is enabled.

Values for the keywords default to the temperature span of the thermodynamic database, as set in the database header. Keyword “off” disables the heat source, leaving the source rate and temperature range intact, and keyword “on” re-enables the heat source; “reset” disables the source and discards any settings for the source rate and temperature range.

## 6.46 help

```
help <help | subject>
```

**Help**

Use the “help” command (also: “man”) to display online documentation for the GWB package. The command “help help” shows a list of the types of documentation available.

## 6.47 history

```
history
```

**Run → History...**

Use the “history” command to view a list of previously executed commands. You can re-execute commands in the **History...** dialog box by double-clicking on a command,

selecting one or more commands, and then choosing **Execute**, or with a history substitution command, as explained in the **Command Line Interface** chapter in the **GWB Reference Manual**.

## 6.48 hydrogen-2

```
hydrogen-2 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “hydrogen-2” command (also, “2-H”) to set the  $^2\text{H}$  isotopic composition of the initial fluid, reactant species (aqueous species, minerals, end members, gases, or oxides) or segregated minerals. The composition may be set on any scale (e.g., SMOW), but you must be consistent throughout the calculation. Example:

```
hydrogen-2 fluid = -120, Muscovite = -40
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
hydrogen-2 remove
hydrogen-2 off
```

clear all settings for  $^2\text{H}$  isotopes from the calculation.

See also the “<isotope>” section above, and the “carbon-13”, “oxygen-18”, and “sulfur-34” commands.

## 6.49 inert

```
inert = <field variable | ?> <units>
```

**Medium** → inert volume     cm3

Use the “inert” command to set the volume of non-reacting space in the system. You may set a value in units of volume, including  $\text{cm}^3$ ,  $\text{m}^3$ , and  $l$ , as well as volume% and “vol. fract.”. The default setting is zero and the default unit is  $\text{cm}^3$ .

Assuming you have not set a value for the initial fluid fraction in the system using the “porosity” command, the program figures the porosity over the course of the calculation as a derived variable. Specifically, it divides the fluid volume by the sum of the fluid volume, mineral volume, and inert volume, and reports this value as a result.

When you have set a value for initial porosity with the “porosity” command, on the other hand, the program works in the contrary sense. In this case, it calculates the

inert volume as that required to form a system of the specified initial porosity; the program now ignores any entry you may have set using the “inert” command.

To restore the default state, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show”.

### 6.50 isotope\_data

```
isotope_data <dataset>
```

**File** → **Open** → **Isotope Data...**

Use the “isotope\_data” command (also: “idata”) to set the name of the database containing isotope fractionation factors for species, minerals, and gases. Example:

```
isotope_data Isotope.mydata
```

### 6.51 itmax

```
itmax = <value | ?>
```

**Config** → **Iteration...**

Use the “itmax” command to set the maximum number of iterations that may be taken in an attempt to converge to a solution for the equations representing the distribution of chemical mass. By default, this variable is set to 999. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

### 6.52 Kd

```
Kd <off>
```

**Config** → **Options...**

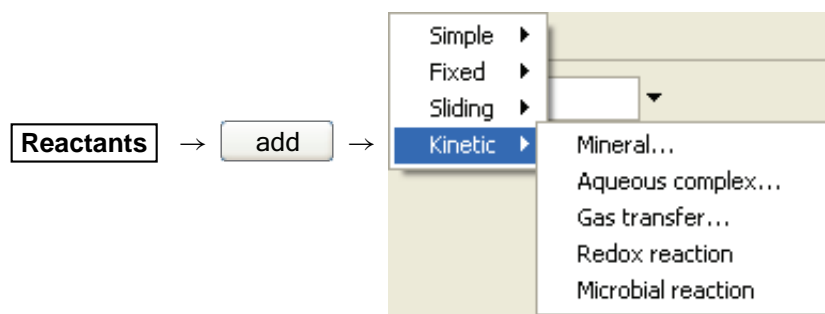
The “Kd” command controls whether the program calculates  $K_d$  distribution coefficients for sorbing components, in units of liters per kg sediment mass. This calculation requires that the mineral mass in the system (as specified for individual minerals and/or in terms of inert volume) be set correctly.

## 6.53 kinetic

```

kinetic <species | mineral | end member | gas> <variable> = <value>
kinetic <species | mineral | end member | gas> <variable> = <field variable> \
    <steady | transient>
kinetic <species | mineral | end member | gas> <apower | mpower(species)> = <value>
kinetic <redox(label)> <variable> = <value>
kinetic <redox(label)> <variable> = <field variable> <steady | transient>
kinetic <redox(label)> <apower | mpower(species)> = <value>
kinetic <microbe(label)> <variable> = <value>
kinetic <microbe(label)> <variable> = <field variable> <steady | transient>
kinetic <microbe(label)> <apower | mpower(species)> = <value>
kinetic <microbe(label)> <apower | mpower(species)> = <value> \
    <apowerA | mpowerA(species)> = <value> \
    <apowerD | mpowerD(species)> = <value>

```



Use the “kinetic” command to set variables defining a kinetic rate law for (1) dissolution or precipitation of any mineral or end member in the initial system or reactant list, (2) the association or dissociation of any aqueous or surface complex in the system modeled, (3) the transfer of gases into or out of an external reservoir, (4) a redox reaction, including those promoted by catalysis or enzymes, or (5) a microbial metabolism.

In the first three cases, you identify the kinetic reaction by the name of the mineral, end member, species, or gas involved. In the case of a redox reaction, you set a label that begins with the characters “redox”, such as “redox-1” or “redox-Fe”. For a microbial reaction, set a label that starts with “microbe”, such as “microbe-Ecoli”.

The rate law you specify in a “kinetic” command, by default, applies to the dissolution of a mineral or end member, dissociation of a complex, dissolution of a gas, or forward progress of a redox or microbial reaction. The synonymous keywords “forward”, “dissolution”, and “dissociation” set this behavior. Including in a “kinetic” command the keyword “reverse” or its synonyms “precipitation”, “complexation”, “association”, or “exsolution” invokes the opposite behavior. In this case, the rate law applies to the reverse reaction: mineral or end member precipitation, complex association, or gas exsolution.

You can append the “transient” keyword when setting the following field variables: rate constant, specific surface area, activation energy, pre-exponential factor, nucleus

area, and critical saturation index. If the variable is defined by an equation, script, or external function, it will then be re-evaluated continuously over the course of the run.

See also the “react” and “remove reactant” commands.

The following paragraphs apply to all types of kinetic reactions. You set the rate constant either directly using the “rate\_con” keyword, or by setting an activation energy and pre-exponential factor with keywords “act\_en” and “pre-exp”. In the absence of promoting and inhibiting species (see next paragraph), you set the rate constant and preexponential factor in (1) mol/cm<sup>2</sup> s for mineral, end member, and gas transfer reactions, (2) molal/s or molal/cm<sup>2</sup> s (the latter when accounting for heterogeneous catalysis) for complexation and redox reactions, and (3) mol/mg s for microbial reactions. The activation energy is specified in J/mol. Example:

```
kinetic "Albite low" rate_con = 1e-15
```

You can set “rate\_con”, “act\_en”, and “pre-exp” as field variables (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

You use the “apower” or “mpower” (also “apow” or “mpow”) keyword to specify any promoting or inhibiting species in the kinetic rate law. Keyword “apower” sets the exponent of a species activity, and “mpower” the exponent of a species molality. Promoting species have positive powers, and the powers of inhibiting species are negative. For example, the command

```
kinetic "Albite low" apower(H+) = 1
```

sets H<sup>+</sup> as a promoting species, the activity of which is raised to a power of one.

You can use aqueous species, minerals (represented by activity, which is one, or molality), end members (activity or mole fraction), gas species (fugacity or partial pressure), surface complexes (molal concentration), and solvent water (activity) as promoting and inhibiting species. When setting an end member, use keyword “xpower” to set the rate law in terms of mole fraction. When setting a gas, use keyword “fpower” to set the rate law in terms of fugacity, and “ppower” to use partial pressure, instead. The generic keyword “power” sets the activity of the solvent or an aqueous species, the activity of an end member, the fugacity of a gas, and the molality of a surface species.

The “order1” and “order2” keywords set nonlinear rate laws. Keyword “order1” represents the power of the  $Q/K$  term, and “order2” represents the power of the  $(1 - Q/K)$  term.

Use the “rate\_law” keyword to set the form of the kinetic rate law for a specific mineral, end member, species, gas, redox reaction, or microbial metabolism. You may set the keyword equal to (1) a character string containing the rate law, (2) the name of a file containing a basic-like script, or (3) the name of a function in a library. The name of a file containing a rate law script must end in “.bas”. To specify a function from a library, set the name of a dynamic link library (DLL) separated from the function name



by a colon (":"), such as "rate\_laws.dll:my\_ratelaw"; the library file must end in ".dll". To return to the program's built-in rate law, enter "rate\_law = off" or "rate\_law = ?".

The following paragraphs apply to dissolution and precipitation reactions. You set the specific surface area of a kinetic mineral or end member (in  $\text{cm}^2/\text{g}$ ) with the "surface" keyword. For example,

```
kinetic "Albite low" surface = 1000
```

The "cross-affinity" option lets you use the saturation state of one mineral to model the reaction rate of another (or of an end member), as is sometimes useful for example in studying glass dissolution. To do so, you use the "xaffin" option. For (a hypothetical) example, the command

```
kinetic Quartz xaffin = Cristobalite
```

causes the program to calculate the reaction rate of quartz according to the fluid's saturation state with respect to cristobalite. The command

```
kinetic Quartz xaffin = OFF
```

turns off the option.

Finally, you use the "nucleus" and "critSI" keywords to set the area available for nucleation (in  $\text{cm}^2/\text{cm}^3$  fluid volume) and the critical saturation index above which the mineral or end member can nucleate. Each of these values, by default, is zero.

Keywords "surface", "nucleus", and "critSI" can be set as field variables (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

The following paragraphs apply to reactions for aqueous and surface complexes.

When you specify a kinetic reaction for the association of an aqueous complex or surface complex, or its dissociation, you can set the complex's initial concentration directly. The concentration can be set heterogeneously, as a field variable. If you do not specify an initial concentration, or set an entry of "?", the program takes the complex at the start of the simulation to be in equilibrium with the initial fluid.

You specify the initial concentration within a "kinetic" command or as a separate command line. For example, the commands

```
kinetic AIF++ rate_con = 3.3e-6, mpow(AIF++) = 1
AIF++ = 1 umol/kg
```

are equivalent to

```
kinetic AIF++ 1 umol/kg rate_con = 3.3e-6, mpow(AIF++) = 1
```

Either case defines a kinetic reaction for decomposition of the  $\text{AlF}^{++}$  ion pair, setting it initially to a free concentration of  $1 \mu\text{mol kg}^{-1}$ .

The following paragraphs apply to gas transfer reactions. Use the “f\_ext” keyword to specify the fugacity of the gas in question in the external reservoir, or keyword “P\_ext” to set its partial pressure. In the latter case, you may append a pressure unit; the default is bar. Keyword “contact” sets the contact area between fluid and external reservoir, in  $\text{cm}^2/\text{kg}$  of water. Example:

```
kinetic CO2(g) f_ext = 10^-3.5, contact = 10
```

Both values can be set as field variables, as described in the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**.

If you do not set a value for the gas’ external fugacity, or set “f\_ext = ?”, the program uses the fugacity in the initial fluid, at the start of the simulation, and the external fugacity.

The following paragraphs apply to redox reactions. You set the form of the redox reaction to be considered as a character string, using the “rxn” (or “reaction”) keyword. For example,

```
kinetic redox-1 rxn = "Fe++ + 1/4 O2(aq) + H+ -> Fe+++ + 1/2 H2O"
```

To specify that the reaction be promoted by a heterogeneous catalyst, set keyword “catalyst” to the name of the catalyzing mineral, or simply to “on”. In the former case, you use keyword “surface” to set the specific surface area of the catalytic mineral (in  $\text{cm}^2/\text{g}$ ). If you have set “catalyst = on”, however, you use the “surface” keyword to set total catalytic area, in  $\text{cm}^2$ . Setting “catalyst = off” disables the catalysis feature.

To set an enzymatically promoted reaction, set keyword “me” to the name of the aqueous species serving as the enzyme, or simply to the value to be used as the enzyme’s molality. In the former case, the program tracks the enzyme molality  $m_E$  over the course of the simulation from the calculated distribution of species. If you have set a numeric value for  $m_E$  using the “mE” keyword, the program uses this value directly. You may alternatively specify the enzyme species or its activity  $a_E$  using keyword “aE”, in which case variables  $m_E$ ,  $m_A$ , and  $m_P$  in the rate law are replaced by the activities  $a_E$ ,  $a_A$ , and  $a_P$ .

For an enzymatic reaction, you further set the half-saturation constants  $K_A$  and  $K_P$  for the forward and reverse reactions in molal with the “KA” and “KP” keywords. You must set a value for  $K_A$ , but may omit  $K_P$ , in which case the  $m_P/K_P$  term in the rate law will be ignored. Setting “enzyme = off” disables the enzyme feature.

The following paragraphs apply to microbial reactions. You set the form of the metabolic reaction using the “rxn” (or “reaction”) keyword, in the same manner as with redox reactions. For example,

```
kinetic microbe-1 rxn = "CH4(aq) + 2 O2(aq) -> HCO3- + H+ + H2O"
```

Set the half-saturation constants  $K_D$  and  $K_A$  for the electron donating and accepting reactions with the “KD” and “KA” keywords. These values default to zero.

You set the powers of species in the numerator of the rate law with the “mpower” keyword, as with other types of kinetic reactions. Use keywords “mpowerD” and “mpowerA” (or “mpowD” and “mpowA”) to set the powers  $p_D$ , etc., of species from the electron donating and accepting reactions, respectively, within the product functions in the rate law’s denominator. For example,

```
kinetic microbe-1 mpower(CH4(aq)) = 1, mpowerD(CH4(aq)) = 1
```

sets the power of the electron-donating species  $\text{CH}_4(\text{aq})$  to one in both the rate law numerator and denominator. Keywords “PKD” and “PKA” set the overall powers  $p_{KD}$  and  $p_{KA}$  of the electron donating and accepting terms in the denominator of the rate law; by default, these are one.

You set the free energy  $\Delta G_{\text{ATP}}$  of ATP hydrolysis (in kJ/mol) with the “ATP\_energy” keyword, and the value of  $n_{\text{ATP}}$  with keyword “ATP\_number”. These values default to zero.

Use the “biomass” keyword to set the initial biomass concentration, in mg/kg. You can set this value as a field variable (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

The “growth\_yield” keyword sets the microbe’s growth yield in mg biomass/mol of reaction progress, and “decay\_con” sets its decay constant in  $\text{s}^{-1}$ ; both values default to zero.

## 6.54 log

```
log <variable> = <value>
```

Use the “log” command to set variables on a logarithmic scale. Examples:

```
log fugacity O2(g) = -65
log activity U++++ = -10
```

## 6.55 mobility

**mobility** = <surface type> <field variable> <**steady** | **transient**>

**Config** → **Sorbing Surfaces...** → mobility

or **File** → **Open** → **Sorbing Surfaces...**

Use the “mobility” command to set up a complexing surface in your model as a mobile colloid. A mobile colloid is composed of the mineral (or minerals) associated with a complexing surface, as well as the ion complexes present on that surface. Only datasets with model type “two-layer”, “three-layer”, or “cd-music” as set in the dataset header are surface complexation models, and hence only those datasets can be used to form a mobile colloid.

Mobility refers to the fraction of the surface in question that can move in the model by advection and dispersion. A surface with a mobility of one moves freely, whereas a mobility of zero sets the surface to be stationary. Intermediate values arise, for example, when some of the surface is attached to the medium, or when colloid motion is impeded by electrostatic interactions. By default in the software the mobility of any surface is zero.

To set a mobile colloid, begin by reading in a surface complexation dataset using the “surface\_data” command. Then, use the “mobility” command, referencing the surface’s label, to set the colloid’s mobility. The label is given at the head of the surface dataset, on a line beginning “Surface type”. The label in dataset “FeOH.sdat”, for example, is “HFO”. If you omit the label, the program will assume you are referring to the surface complexation dataset most recently read.

You can define the mobility as a field variable, which means you can have the program calculate mobility using an equation, script, or compiled function you provide. When you set the “transient” keyword, the program upon undertaking each time step in the simulation evaluates mobility at each nodal block. In the “steady” case, which is the default, the program evaluates mobility at each block just once, at the start of the run.

Example:

```
surface_data FeOH.sdat
mobility HFO = 100%
```

where “HFO” is the label for the surface defined by dataset “FeOH.sdat”.

Restore the default behavior of immobility by entering a command such as

```
mobility HFO ?
```

## 6.56 no-precip

```
no-precip <off>
```

**Config** → **Options...**

Use the “no-precip” command (also: “noprecip”) to prevent new minerals from precipitating over the course of a simulation. By default, they are allowed to precipitate. Use:

```
no-precip
no-precip off
```

See also the “precip” command.

## 6.57 nswap

```
nswap = <value | ?>
```

**Config** → **Iteration...**

Use the “nswap” command to set the maximum number of times that the program may swap entries in the basis in an attempt to converge to a stable mineral assemblage. By default, this variable is set to 30. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.58 oxygen-18

```
oxygen-18 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “oxygen-18” command (also, “18-O”) to set the  $^{18}\text{O}$  isotopic composition of the initial fluid, reactant species (aqueous species, minerals, end members, gases, or oxides) or segregated minerals. The composition may be set on any scale (e.g., SMOW), but you must be consistent throughout the calculation. Example:

```
oxygen-18 fluid = -10, Quartz = +15
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
oxygen-18 remove  
oxygen-18 off
```

clear all settings for  $^{18}\text{O}$  isotopes from the calculation.

See also the “<isotope>” section above, and the “carbon-13”, “hydrogen-2”, and “sulfur-34” commands.

### 6.59 pause

```
pause
```

Use the “pause” command to cause **React** to pause temporarily during input. This command is useful when you are debugging scripts.

### 6.60 pe

```
pe = <value | ?>
```

```
Basis →  ▼ pe ▼
```

Use the “pe” command to set oxidation state in the initial system in terms of pe. Example:

```
pe = 10
```

is equivalent to

```
log activity e- = -10
```

where “e-” is the electron. Use “?” to unset a pe value: See also the “activity”, “Eh”, “pH”, “fugacity”, “fix”, and “slide” commands.

## 6.61 permeability

```
permeability <intercept = field variable | ?> <unit> <steady | transient> \
  <porosity = field variable | ?> <steady | transient> \
  <mineral = field variable | ?> <steady | transient>
```

→ A (porosity)  +  ▼  
 B (intercept)  +  ▼ darcy ▼

You use the “permeability” command to set the correlation by which the program calculates sediment permeability. **React** calculates this value as a reported variable consistent with **X1t** and **X2t**, but does not use it in its calculations.

The correlation gives log permeability in any of the units listed in the **Units Recognized** chapter of the **GWB Reference Manual** (darcys by default) as a linear function of the porosity (expressed as a volume fraction) of a nodal block and, optionally, the volume fractions of one or more minerals. The “transient” keyword causes the model to evaluate the coefficient in question continuously over the course of the simulation, if it is set with an equation, script, or function.

Examples:

```
permeability intercept = -11 cm2 porosity = 15
permeability Kaolinite = -8
```

The latter command adds a term for the mineral Kaolinite to the existing correlation. To remove a term from the correlation, set a value of “?”. The entry

```
permeability Kaolinite = ?
```

for example, removes the correlation entry for that mineral.

The default correlation is

$$\log k = -5 + 15\phi$$

where  $k$  is permeability in darcys and  $\phi$  is porosity (expressed as a fraction).

## 6.62 pH

```
pH = <value | ?>
```

→  ▼ pH ▼

Use the “pH” command to set pH in the initial system. Example:

```
pH = 5
```

is equivalent to

```
log activity H+ = -5
```

Use “?” to unset a pH value: See also the “activity”, “Eh”, “pe”, “fugacity”, “fix”, and “slide” commands.

### 6.63 phrqpitz

```
phrqpitz
```

```
File → Open → Thermo Data... → thermo_phrqpitz.tdat
```

Use the “phrqpitz” command to set the program to calculate species’ activity coefficients using the Harvie-Møller-Weare equations, as implemented in the USGS program PHRQPITZ. Executing this command automatically sets the input dataset of thermodynamic data to “thermo\_phrqpitz.tdat”. Note that dataset “thermo\_phrqpitz.tdat” is primarily intended to support calculations at or near 25°C.

### 6.64 pickup

```
pickup <system => <entire | fluid>  
pickup reactants = <entire | fluid | minerals>
```

```
Run → Pickup
```

Use the “pickup” command to take the results of a reaction path as the starting point for a new reaction path. You may pick up the entire system, or just the fluid or minerals resulting from a reaction path, and use this as your new initial system or reactant list. Default choices are “system” and “entire”. Examples:

```
pickup  
pickup fluid  
pickup reactants  
pickup reactants = minerals
```

When you do a simple “pickup” (i.e., “pickup system = entire”), the program retains within the system all kinetic reactions that were defined in the original path, at the reactions’ endpoint state. A “pickup fluid” (fully, “pickup system = fluid”) command retains only the kinetic reactions occurring in the fluid – the kinetic redox and aqueous



complexation reactions – in the new reaction path; kinetic reactions involving minerals, surfaces, a gas phase, or microbes are discarded.

Picking up the entire endpoint system, or just the endpoint minerals, as reactants (i.e., “pickup reactants” or “pickup reactants = minerals”) causes the program to retain the kinetic reactions involving mineral precipitation and dissolution. In these cases, the other types of kinetic reactions are discarded. The command “pickup reactants = fluid” causes the program to discard any kinetic reactions that may be set.

The commands

```
pickup TDS
pickup density
```

are obsolete, because releases 7.0 and later of the software calculate the TDS and density automatically.

## 6.65 pitz\_dgamma

```
pitz_dgamma = <value | ?>
```

**Config** → **Iteration...**

Use the “pitz\_dgamma” command to control the relative change in an activity coefficient’s value the program allows during each Newton-Raphson iteration, when the Harvie-Møller-Weare activity model has been invoked. By default, the program allows a 10% change, which corresponds to a value of 0.1.

## 6.66 pitz\_precon

```
pitz_precon = <value | ?>
```

**Config** → **Iteration...**

Use the “pitz\_precon” command to control the maximum number of passes the program takes through the pre-conditioning loop before beginning a Newton-Raphson iteration, when a virial activity model (Harvie-Møller-Weare or SIT) has been invoked. By default, the program makes up to 10 passes. In cases of difficult convergence, counter-intuitively, it can sometimes be beneficial to decrease this value.

## 6.67 pitz\_relax

```
pitz_relax = <value | ?>
```

**Config** → **Iteration...**

The “pitz\_relax” command controls under-relaxation when evaluating the Harvie-Møller-Weare equations. The program at each Newton-Raphson iteration assigns activity coefficients as a weighted average of the newly calculated value and the corresponding value at the previous iteration level. Setting pitz\_relax to zero eliminates under-relaxation, so the newly calculated values are used directly; a value of one, in contrast, should be avoided because it would prevent the activity coefficients from being updated. By default, the program carries an under-relaxation factor of 0.5.

### 6.68 plot

```
plot <character | binary> <on | off>
```

**Config** → **Output...**

Use the “plot” command to set the format of the “React\_plot.gtp” dataset. The dataset is written in XML, a standard format that is easy to parse for use with alternative plotting programs. Numerical data in the dataset can be represented in either standard decimal notation (keyword “character”) for user readability or a binary encoding (keyword “binary”) that maintains full precision of data. The default format, XML with binary encoded data, also zips the output file to reduce output size and improve file opening speed. The command “plot off” causes **React** to bypass writing calculation results to the “React\_plot.gtp” dataset, which is used to pass input to **Gtplot**. By default, the program writes output to the dataset. The command “plot on” (or just “plot”) re-enables the output. To see the current setting, type “show print”.

### 6.69 pluses

```
pluses <off>
```

**Config** → **Stepping...**

Use the “pluses” command to cause **React** to simply output a plus sign (“+”) each time it iterates to a solution, rather than printing a banner showing the number of iterations required and final residual value.

### 6.70 polymorphs

```
polymorphs <mineral>
```

**Config** → **Show...** → Minerals → *select mineral*

Use the “polymorphs” command to list each polymorph (mineral identical in composition to the specified mineral) in the current database. Example:

```
polymorphs Albite
```

## 6.71 porosity

```
porosity = <field variable | ?>
```

```
Medium → porosity +  ▼
```

Use the “porosity” command to set (as a volume fraction) the initial porosity of the system. Porosity, the fraction of the system occupied by fluid, is the ratio of fluid volume to the sum of fluid, mineral, and inert volume.

The examples

```
porosity = 0.30
porosity = 30%
```

are equivalent.

When you specify the porosity, the program will figure the difference between the volume of a system of the given porosity and fluid volume, and the volume taken up initially in the system by minerals and fluid. The program assigns this difference as inert, non-reactive volume (see the “inert” command). In this case, the program ignores any settings that may have been made with the “inert” command.

When you do not specify an initial porosity with the “porosity” command, on the other hand, the program calculates it from volumes in the system of fluid, minerals, and inert space. To restore this default behavior, enter the command with an argument of “?”.

## 6.72 precip

```
precip <off>
```

```
Config → Options...
```

Use the “precip off” command to prevent new minerals from precipitating over the course of a simulation. By default, they are allowed to precipitate. Use

```
precip
precip off
```

See also the “no-precip” command.

## 6.73 press\_model

```
press_model <Tsonopoulos | Peng-Robinson | Spycher-Reed | default | off>
```

**Config** → **Options...**

The “press\_model” command (also: “pressure\_model”) lets you control the method used to calculate fugacity coefficients and gas partial pressures. Three nonideal pressure models are coded in the software: Tsonopoulos, Peng-Robinson, and Spycher-Reed, as described in the **GWB Essentials Guide**.

By default, the pressure model is taken from the header lines of the thermo dataset in use, but you can use the “press\_model” command to override the default setting. Keywords “Tsonopoulos”, “Peng-Robinson”, and “Spycher-Reed” set the pressure model directly (you need only enter the first three letters), whereas “default” returns to the setting in the thermo dataset, and “off” (also: “ideal”) disables the feature, forcing all fugacity coefficients to one.

Examples:

```
press_model Peng-Robinson  
press_model default
```

## 6.74 print

```
print <option> = <long | short | none>  
print <off | on>  
print <numeric | alphabetic>
```

**Config** → **Output...**

Use the “print” command (also: “printout”) to control the amount of detail to be written into the “React\_output.txt” dataset. For example, the dataset can contain information about each aqueous species, information on only species with concentrations greater than  $10^{-8}$  molal, or no species information.

Options, which may be abbreviated to three letters, and their default settings are:

species	short
surfaces	long
saturations	short
gases	long
basis	none
orig_basis	long
elements	long
reactions	none
stagnant	none

The “print” command can also be used to arrange entries in the output dataset either numerically or alphabetically:

print numeric
print alphabetic

To see the current print settings, type “show print”. Finally, the command “print off” causes **React** to bypass writing calculation results to the “React\_output.txt” and “React\_isotope.txt” datasets. By default, the program writes to the datasets. The command “print on” (or just “print”) re-enables the output.

## 6.75 pwd

<b>pwd</b>
<b>File</b> → <b>Working Directory...</b>

The “pwd” command returns the name of the current working directory. The command has the same effect as typing “show directory”. See the “chdir” command.

## 6.76 quit

<b>quit</b>
<b>File</b> → <b>Quit</b>

Use the “quit” command (also: “end”, “q”) to exit the program. Touching **Ctrl+Q** serves the same purpose as the “quit” command.

## 6.77 ratio

**ratio** <species ratio> = <value | ?>

**Basis** →  ▼ ratio ▼

Use the “ratio” command to constrain an activity ratio in the initial system. Example:

```
swap Ca++/Na+^2 for Ca++
ratio Ca++/Na+^2 = 0.2
```

Use “?” to unset a ratio value: See also the “activity”, “pH”, “Eh”, “pe”, “fugacity”, “fix”, and “slide” commands.

## 6.78 react

**react** <amount> <unit> <as <element symbol>> \  
<species | mineral | end member | gas | oxide> <cutoff> = <value>

**Reactants** →  → 

Simple	Aqueous...
Fixed	Mineral...
Sliding	Gas...
Kinetic	Oxide...

Use the “react” command (abbrev.: “rct”) to define the reactants for the current simulation. To set a kinetic rate law for a reactant, use the “kinetic” command.

Units for the amount of reactant to add over a reaction path can be:

mol	mmol	umol	nmol	
kg	g	mg	ug	ng
eq	meq	ueq	neq	
cm3	m3	km3	l	
mol/kg	mmol/kg	umol/kg	nmol/kg	
molal	mmolal	umolal	nmolal	
mol/l	mmol/l	umol/l	nmol/l	
g/kg	mg/kg	ug/kg	ng/kg	
wt%	"wt fraction"			
g/l	mg/l	ug/l	ng/l	
eq/kg	meq/kg	ueq/kg	neq/kg	
eq/l	meq/l	ueq/l	neq/l	

Units of mass or volume can be expressed per volume of the porous medium. Examples:

mol/cm <sup>3</sup>	g/cm <sup>3</sup>
mmol/m <sup>3</sup>	ug/m <sup>3</sup>
volume%	"vol. fract"

Units of mass or volume can be set as absolute rates by appending “/s”, “/day”, “/yr”, or “/m.y.”. For example,

mmol/s	g/day	cm <sup>3</sup> /yr
mol/kg/s	mg/kg/day	cm <sup>3</sup> /kg/yr

Use the “as” keyword to specify reactant masses as elemental equivalents. For example, the command

```
react 10 umol/kg CH3COO- as C
```

specifies 5 umol/kg of acetate ion, since each acetate contains two carbons, whereas

```
react 20 mg/kg SO4-- as S
```

would cause the program to add 59.9 mg/kg of sulfate, since the ion’s mole weight is about 3 times that of sulfur itself.

You can set a cutoff to limit the amount of a reactant. For example, if you set the amount of a reactant to two moles and set a cutoff of one, then **React** will add one mole of the reactant over the first half of the path and none over the second half. Enter the cutoff value in the same units as the amount of reactant. Examples:

```
react 10 grams Quartz
react 1e-2 mol Muscovite cutoff = .5e-2
react .01 mol/day HCl
```

See also the “kinetic” and “remove reactant” commands.

## 6.79 reactants

```
reactants times <value>
```

```
Reactants → reactants times  ▼
```

Use the “reactants” command to vary the total amounts of the reactants by a given factor.

Example:

```
reactants times 1/10
```

## 6.80 read

```
read <dataset>
```

**File** → **Open** → **Read Script...**

Use the “read” command to begin reading commands from a script stored in a dataset. Example:

```
read Seawater
```

Control returns to the user after the script has been read, unless the script contains a “quit” command. You can also use the “read” command in place of the “data” or “surface\_data” command to read a thermo or surface reaction dataset.

When typing a “read” command, you can use the spelling completion feature to complete dataset names: touch “[tab]” or “[esc]” to cycle through the possible completions, or **Ctrl+D** to list possible completions.

## 6.81 remove

```
remove <basis specie(s)> <solid solution(s)> <reactant(s)>
```

```
remove basis <basis specie(s)>
```

```
remove solid_solution <solid solution(s)>
```

```
remove reactant <reactant(s)>
```

**Basis** or **Reactants** or

**Config** → **Solid Solutions...** →

*select entry(ies)* →

Use the “remove” command (also: “rm”) to eliminate one or more basis entries or reactants from consideration in the calculation. Example:

```
remove Na+
remove Quartz Calcite
remove reactant H2O
```

Components can be reentered into the basis using the “swap”, “add”, “activity”, and “fugacity” commands. You can also use the “remove” command to remove solid solutions configured in a run (but not those set in the thermo dataset):

```
remove solid_solution mySS
```



## 6.82 report

```
report <option>  
report set_digits <value>
```

Once the program has completed a calculation, you can use the “report” command to return aspects of the calculation results. You can use this command from the **Command** pane to query the program interactively. More significantly, the “report” command provides a means of communicating calculation results to control scripts and external programs running GWB programs by remote control.

For arguments available, see the **Report Command** chapter in the **GWB Reference Manual**. For details about control scripts and the plug-in feature, see the **Control Scripts** and **Plug-in Feature** chapters of the **GWB Reference Manual**. Use the “set\_digits” argument to set the number of significant digits in the values reported, and the “set\_node” argument to identify the node of interest.

## 6.83 reset

```
reset  
reset system  
reset reactants  
reset variables
```

**File** → **Reset Configuration**

Use the “reset” command to begin defining the chemical system again with a clean slate. Your current settings will be lost, and all options will be returned to their default states. The command, however, does not alter the setting for the thermo dataset. The “reset system” command resets only the initial system. Similarly, typing “reset reactants” resets the reactant system, and “reset variables” sets each settable variable to its default value.

## 6.84 resume

```
resume
```

**File** → **Resume**

Use the “resume” command to restore the configuration from when you last exited the program. The program configuration is stored in a file “react\_resume.rea” in your profile directory (e.g., “c:\Documents and Settings\jones\Application Data\GWB”). Your current configuration is discarded.

## 6.85 save

```
save <dataset> <hex>
```

```
File → Save As...
```

Use the “save” command to write the current chemical system into a dataset in **React** format commands. The dataset can be used as an **React** input script. Examples:

```
save  
save kspar.rea
```

If you don't specify a dataset name, **React** will ask for one. The optional keyword “hex” causes the program to output numbers as hexadecimal values.

## 6.86 script

```
script  
script end
```

Use the “script” command to mark the beginning, and optionally the end, of a control script. Control scripts differ from standard input files in that they can contain not only **React** commands, but control structures such as loops and if-else branches. Control scripts follow the Tcl syntax, described in [www.tcl.tk](http://www.tcl.tk) and [mini.net/tcl](http://mini.net/tcl), as well as several widely available textbooks.

Within a control script, filenames are written with double rather than single backslashes. For example, a “read” command might appear as

```
read GWB_files\\My_file.rea
```

within a control script.

## 6.87 segregate

```

segregate <mineral(s)>
segregate <mineral> <value>
segregate <mineral> <initial value> <final value>
segregate <mineral> <value> Xi = <value> <value> Xi = <value>

```

**Config** → **Isotopes...** →

**Config** → **Isotopes...** → *select entry(ies)* →

The “segregate” command causes minerals to be isolated from isotopic exchange over the course of a reaction path. By default, a mineral in the equilibrium system remains in isotopic equilibrium with the fluid and other minerals. A segregated mineral, on the other hand, changes in isotopic composition only when it precipitates from solution; it alters the system’s composition only if it dissolves. Example:

```
segregate Quartz Calcite "Maximum Microcline"
```

Optionally, a fraction of a mineral’s mass may be isotopically segregated, and that fraction may vary linearly with reaction progress. Examples:

```

segregate Quartz 100%, Muscovite 7/10
segregate Ca-Saponite 100% 0%
segregate Ca-Saponite 80% Xi = .3, 20% Xi = .7

```

In the latter example, the program segregates 80% of the mass of Ca-Saponite until the reaction progress variable Xi reaches .3, decreases the segregated fractionation until it reaches 20% when Xi equals .7, and then holds the value constant until the end of the path. To display the isotopically segregated minerals, type “show isotopes”.

Note, you use the name of the corresponding mineral to segregate a solid solution end member.

## 6.88 show

```

show <option>
show <aqueous | minerals | solid_solutions | gases | oxides | surfaces> \
  <with | w/> <basis entry | string>

```

**Config** → **Show...**

## GWB Command Reference

---

Use the “show” command to display specific information about the current system or database. The options are:

show	show all	show altered
show aqueous	show basis	show commands
show couples	show directory	show elements
show gases	show initial	show isotopes
show minerals	show oxides	show printout
show reactants	show show	show solid_solutions
show suppressed	show surfaces	show system
show variables		

The command “show show” gives a list of show command options. When you type “show aqueous” (or “minerals”, “solid\_solutions”, “gases”, “oxides”, or “surfaces”), the program lists all entries of that type in the thermo database. The “solid\_solutions” option additionally includes solutions defined locally. A long form lets you limit the query to entries composed of a particular basis species or containing a text string in the name:

```
show aqueous with Al+++  
show minerals w/ chal
```

There is also a compound form of the “show couples” command:

```
show coupling reactions
```

This command produces a complete list of the redox couples, in reaction form.

### 6.89 simax

```
simax = <value | ?>
```

```
Config → Iteration...
```

The “simax” command sets in molal units the maximum value of the stoichiometric ionic strength used in calculating water activity when Helgeson’s B-dot Debye-Hückel model is employed. By default, this variable is set to 3 molal. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.90 slide

**slide** <unit> <species | gas> to <value>

Use the “slide” command to linearly adjust the activity of the specified species, fugacity of the gas, or an activity ratio toward <value>, which is attained at the end of the path. Note that the interpolation is made linearly on the logarithm of activity or fugacity if <value> is set as a log, and that <unit> can be “activity” or “fugacity” (“a” or “f” for short), “ratio”, “pH”, “pe”, or “Eh”, or omitted. Examples:

```
slide pH to 5
slide activity Cl- to 2/3
slide f CO2(g) to 10^-3.5
slide log f O2(g) to -65
```

## 6.91 solid\_solution

**solid\_solution** <name = <label>> <type> <discrete | continuous> \  
 <<mineral> tag = <label>> <contains> <<mineral> tag = <label>> \  
 <from <value>> <to <value>> <step = <value>> <<variable> = <value>>  
**solid\_solution** <solid solution> <options>  
**solid\_solution remove** <solid solution>

**Config** → **Solid Solutions...** →

**Config** → **Solid Solutions...** → *select entry(ies)* →

Use the “solid\_solution” command (abbrev.: “solid\_soln”) to define binary solid solutions in addition to any provided in the current thermo database. Supply a name with the “name” keyword, or simply set one as the first argument.

You can select the type of solid solution, either “ideal” (the default), “site\_mixing”, “guggenheim”, “regular”, “cubic”, or “third\_order”. For a site-mixing solution, set the site number “bsite”, which defaults to 1. For a Guggenheim solution, set dimensionless

parameters “a0”, “a1”, and “a2” or a temperature expansion for their J/mol counterparts using coefficients “p1”-“p9”, each of which defaults to zero. For the other nonideal models, similarly, set coefficients “p1”, “p2”, etc., as appropriate. For more information, see **Solid solutions** under **Configuring the programs** in the **GWB Essentials Guide** and the **Thermo Datasets** chapter in the **GWB Reference Manual**. You can as well choose whether the solid solution is to be treated as continuous, the default option, or as a sequence of “discrete” minerals.

You need to specify two end members for a solid solution, each associated with an abbreviated “tag”. The tag cannot be the name of another entry in the database. You can set a composition range in terms of the mole fraction of the most recently referenced end member, using keywords “from” and “to”, as well as a compositional “step” separating tranches of a discrete solution; by default, composition ranges from zero to one and the step is one-twentieth of the range. Keyword “contains” may be inserted for clarity. Examples:

```
solid_solution name = my_ss Calcite contains Strontianite from 0% to 10% a0= 0.05
solid_solution my_ss Calcite Strontianite from 0 to .1 a0= 0.05
solid_solution Biotite site_mixing Annite Phlogopite bsite= 3
```

If you don’t specify a solid solution’s name, the app will create one automatically using the tags for the two end members, separated by a hyphen. If you don’t set a tag for an end member, the app will pick up the first six characters of the corresponding mineral’s name. For example, the command

```
solid_solution Calcite contains Strontianite
```

produces a solid solution named “Calcit-Stront”, whereas typing

```
solid_solution Calcite tag Cc contains Strontianite tag Str
```

yields the name “Cc-Str”. To redefine the properties of a solid solution, enter the name followed by the properties to be changed:

```
solid_solution my_ss a0 = 1 a1 = 2
```

(You may prefer to use the “alter” command to change solutions defined within the thermo database; the two commands are functionally equivalent.) The command

```
solid_solution remove my_ss
```

discards a previously defined solution.

## 6.92 sorbate

```
sorbate <exclude | include>
```

**Config** → **Options...**

Use the “sorbate” command to tell the program, when considering sorption onto surfaces (see the “surface\_data” command), whether to include or exclude sorbed species in figuring the composition of the initial system. By default, the program does not include sorbed species in this calculation. If you set the Ca<sup>++</sup> concentration to 15 mg/kg, for example, the initial system would contain that amount in the fluid and an additional amount sorbed onto mineral surfaces. If you type the command “sorbate include”, however, that amount would apply to the sum of the Ca<sup>++</sup> sorbed and in solution.

## 6.93 span

```
span <value | min | ?> <unit> <to> <value | max | ?> <unit>  
span <off | on | reset>
```

**Config** → **Options...**

The “span” command lets you set a temperature range across which stability must be defined for each species considered in the calculation. The lower limit may be identified with keyword “from” or “Tmin”, and the upper denoted by “to” or “Tmax”. A value of “?” unsets the bound in question.

With *T*-table datasets, simply setting “span = on” without specifying a range forces the program to load species’ log *K*s in polynomial mode. In other words, it will take log *K*s from polynomial fits against temperature to the stability constants in the thermo dataset, even if the calculation is performed entirely at one of the dataset’s principal temperatures. Otherwise, an isothermal calculation run at one of the principal temperatures (commonly 0°C, 25°C, 60°C, ...) takes log *K*s for aqueous species, minerals, and so on directly from the values given in the thermo dataset.

Temperature ranges of validity can also be set in the thermo dataset for the virial coefficients used to calculate “Pitzer” and SIT activity coefficients. The “span” command works in a similar manner in such cases, loading only virial coefficients with validity that spans the calculation’s temperature range.

As an example of using the “span” command, you may wish to compare results calculated at 25°C with those figured at 30°C. Since log *K* for a number of species in the thermo dataset is given at only 25°C, unless the “extrap” option is set, the former calculation may well take into account a somewhat larger set of species than the latter. Entering the command

```
span 25 C to 30 C
```

assures the same set of species is loaded in each case, and that both runs take log  $K$ s from polynomial fits.

Set the bounds to specific temperatures, or to “min” or “max”, which respectively represent the lowest and highest temperatures considered in the thermo dataset, as loaded at run time. If you set the temperature range directly, it will be bracketed to that of the thermo dataset. For example, the command

```
span -1000 C to 1000 C
```

is functionally the same as

```
span 0 C to 300 C
```

when the “thermo.tdat” dataset is loaded, since the dataset’s range is 0°C to 300°C.

The temperature bounds set here are the same as those set with the “heat\_source” command: The command

```
span 20 C to 100 C
```

may be equivalently expressed

```
heat_source Tmin = 20 C, Tmax = 100 C
```

as long as the heat source option is enabled.

Keyword “off” disables the feature, leaving values for the limits intact, and keyword “on” re-enables the feature. The “reset” keyword clears the limits, disabling the feature. By default, the span feature is disabled.

### 6.94 start\_date

```
start_date <value | off>
```

**Config** → **Stepping...**

Use the “start\_date” command to set an explicit starting date of the reaction. This can be used to coordinate the plotting of dated scatter data samples stored in a **GSS** spreadsheet on the reaction path in **Gtplot**. The date should be in the format “MM/DD/YYYY”. Use “off” to return to the default of not set.

```
start_date 10/30/2008  
start_date off
```



## 6.95 start\_time

```
start_time <value | off>
```

**Config** → **Stepping...**

Use the “start\_time” command to set an explicit starting time of the reaction. This can be used to coordinate the plotting of timed scatter data samples stored in a **GSS** spreadsheet on the reaction path in **Gtplot**. The time should be in the format “HH:MM:SS”. Use “off” to return to the default of not set.

```
start_time 11:30:00  
start_time off
```

## 6.96 step\_increase

```
step_increase = <value | ?>
```

**Config** → **Stepping...**

Use the “step\_increase” command to set the greatest proportional increase, from one step to the next, in the size of the time step. This variable does not apply to reaction paths with logarithmic reaction stepping (see variable “delxi”).

By default, this variable is set to 1.5. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.97 step\_max

```
step_max = <value | ?>
```

**Config** → **Stepping...**

Use the “step\_max” command to limit the number of reaction steps the program may take to trace a simulation. Use a “?” to restore the default state, which is no prescribed limit. To see the current setting, type “show variables”.

## 6.98 suffix

```
suffix <string>
```

**Config** → **Output...**

Use the “suffix” command to alter the names of the output datasets (“React\_output.txt”, “React\_isotope.txt”, and “React\_plot.gtp”) by adding a trailing string. Example:

```
suffix _run2
```

produces output datasets with names such as “React\_output\_run2.txt”.

### 6.99 sulfur-34

```
sulfur-34 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “sulfur-34” command (also, “34-S”) to set the <sup>34</sup>S isotopic composition of the initial fluid, reactant species (aqueous species, minerals, end members, gases, or oxides) or segregated minerals. The composition may be set on any scale (e.g., CDT), but you must be consistent throughout the calculation. Example:

```
sulfur-34 fluid = +45, H2S(g) = -2
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
sulfur-34 remove  
sulfur-34 off
```

clear all settings for <sup>34</sup>S isotopes from the calculation.

See also the “<isotope>” section above, and the “carbon-13”, “hydrogen-2”, and “oxygen-18” commands.

### 6.100 suppress

```
suppress <species, minerals, solid solutions, gases, surface species | ALL>
```

**Config** → **Suppress...**

Use the “suppress” command (also: “kill”) to prevent certain aqueous species, surface species, minerals, solid solutions, or gases from being considered in a calculation. Example:

```
suppress H3SiO4- Quartz "Maximum Microcline"
```

prevents the three entries listed from being loaded from the database. Typing “suppress ALL” suppresses all of the minerals and solid solutions in the thermodynamic database, as well as any solid solutions defined locally.

The “unsuppress” command reverses the process. To suppress all but a few minerals, you could type

```
suppress ALL
unsuppress Quartz Muscovite Kaolinite
```

## 6.101 surface\_capacitance

```
surface_capacitance = <value | ?>
surface_capacitance C1 = <value | ?> C2 = <value | ?>
surface_capacitance on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant capacitance  ▼ F/m<sup>2</sup>  
 or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_capacitance”) to set, in units of F/m<sup>2</sup>, the capacitance or capacitances of a sorbing surface.

For a two-layer complexation model, when you set a capacitance with the command, or if a value for capacitance is set in the header section of the surface reaction dataset, **React** will model surface complexation for the surface in question using the constant capacitance model, rather than the full two-layer model.

A triple-layer or CD-MUSIC complexation model, on the other hand, requires two capacitances,  $C_1$  and  $C_2$ . Default capacitances are set in the header of the surface dataset, but you may use the “C1” and “C2” arguments to override the default settings from the command line.

If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_capacitance on HFO = 2
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface.

## 6.102 surface\_data

```
surface_data <sorption dataset>
surface_data remove <sorption dataset | surface type>
surface_data OFF
```

**File** → **Open** → **Sorbing Surfaces...** →

**File** → **Open** → **Sorbing Surfaces...** → *select surface(s)* →   
 or **Config** → **Sorbing Surfaces...**

Use the “surface\_data” command (abbrev.: “surf\_data”) to specify an input dataset of surface sorption reactions to be considered in the calculation. The dataset name should be enclosed in quotes if it contains any unusual characters. Use the “remove” argument to eliminate a surface dataset, specified by name or surface type (e.g., “HFO”), from consideration. The argument “OFF” disables consideration of all surface complexes.

You can specify more than one sorbing surface in a model by repeating the “surface\_data” command for different datasets (a dataset of surface reactions for sorption onto hydrous ferric oxide, several triple-layer and CD-MUSIC datasets, as well as example datasets for the ion exchange,  $K_d$ , Freundlich, and Langmuir models are distributed with the software). To remove a dataset of surface reactions from consideration, you use commands such as

```
surface_data remove FeOH.sdat
surface_data remove HFO
surface_data OFF
```

The latter command removes all of the surface datasets that have been loaded.

## 6.103 surface\_potential

```
surface_potential = <value | ?>
surface_potential on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant potential  ▼ mV  
 or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_potential”) to set, in units of mV, the electrical potential for a two-layer sorbing surface. When you set this value (or if a value is set in the header section of the surface reaction dataset), **React** will model surface complexation for the surface in question using the constant potential method, rather than invoking the full two-layer model.

If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_potential on HFO = 0
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface.

## 6.104 swap

```
swap <new basis> <for> <basis species>
```

```
Basis → ↺
```

Use the “swap” command to change the set of basis entries. All reactions are written internally in terms of a set of basis species that you can alter to constrain the composition of the initial system. An aqueous species, mineral, gas, or activity ratio can be swapped into the basis in place of one of the original basis species listed in the database. Examples:

```
swap CO3--      for HCO3-
swap Quartz     for SiO2(aq)
swap CO2(g)     for H+
swap O2(g)      for O2(aq)
swap Ca++/Na+^2 for Ca++
```

Each end member of a continuous solid solution can similarly be swapped into the basis. Use the syntax <solid solution>::<end member tag>. For example, the sodium end member Albite (tag “Ab”) of the plagioclase feldspar solid solution (name “Plag”) can be swapped into the basis simultaneously with the calcium end member Anorthite (tag “An”) as follows:

```
swap Plag::Ab for Na+
swap Plag::An for Al+++
```

The new species must contain in its composition the original basis species being swapped out (you can’t swap lead for gold). For example, CO<sub>2</sub>(g) is composed of HCO<sub>3</sub><sup>-</sup>, H<sup>+</sup>, and water in “thermo.tdat”. The reactions in the thermo dataset (once reduced to the set of basis and redox species and modified to reflect enabled redox couples) show the basis entries for which a species may be swapped. For a list of original basis species, type “show basis”. To reverse a swap, type “unswap <species>”.

## 6.105 system

```
system <system command>
```

Use the “system” command to execute a command on the computer’s operating system (a “DOS command”) without leaving **React**. Example:

```
system "copy React_output.txt React_output2.txt"
```

Enclose multiword system commands in quotes. An alternative way to fork a system command is to begin the **React** command with a “\$”:

```
$copy React_output.txt React_output2.txt
```

In this case, no quotation marks are needed.

When a “system” command is executed, a “Command Prompt” window will appear briefly on your screen. Due to limitations of the Windows operating system, you cannot fork a command that requires user input, and you will not be able to see any output (including error messages) that might be generated by the command.

## 6.106 TDS

```
TDS = <value | ?>
```

```
Basis →  advanced → TDS  ▼ mg/kg
```

Use the “TDS” command to set in mg/kg the total dissolved solids for the initial fluid, if you don’t want the program to calculate this value automatically. The program uses the TDS when needed to convert input constraints into molal units.

To restore automatic calculation of the TDS, type the command with no argument or with an argument of “?”. To see the variable’s current setting, type “show variables”.

## 6.107 temperature

```

temperature = <value> <unit>
temperature initial = <value> <unit> final = <value> <unit>
temperature initial = <value> <unit> reactants = <value> <unit>
temperature constant = <on | off>
temperature reset

```

**Basis** → temperature initial  C ▼ → normal  
 slide temperature to  C ▼ sliding temperature  
 reactant temperature  C ▼ reactant mixing  
 -----  
 hold isothermal

Use the “temperature” command (also: “T”) to set the temperature of the system or reactants. The **Units Recognized** chapter in the **GWB Reference Manual** lists possible units, which default to “C”. Examples:

```

temperature 25 C
T initial = 25 C, final = 300 C
T initial = 398 K, reactants = 498 K

```

Temperature values can range over the span of the thermo dataset, from 0°C to 300°C for “thermo.tdat”; 25°C is the default.

The “constant” keyword lets you hold temperature constant over the course of the calculation at the initial value, regardless of other settings; “on” enables the feature, as does omitting an argument; “off” disables it. The “reset” keyword restores temperature settings to their default states.

## 6.108 theta

```

theta = <value | ?>

```

**Config** → **Stepping...**

Use the “theta” command to set the time weighting variable used in evaluating kinetic rate laws. The value may vary from zero (full weighting at the old time level) to one (full weighting at the new time level). By default, this variable is set to 0.6. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.109 timax

```
timax = <value | ?>
```

**Config** → **Iteration...**

The “timax” command sets in molal units the maximum value of ionic strength used in calculating species’ activity coefficients when Debye-Hückel methods are employed. The limiting value is also used when water activity is calculated according to Wolery (1992b). By default, this variable is set to 3 molal. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 6.110 time

```
time <start = <value> <unit>> <end = <value> <unit>>
```

```
time = off
```

**Basis** →  time    start  ▼ day ▼    end  ▼ day ▼

Use the “time” command (also: “t”) to set the time span of a kinetic reaction path. The starting time, by default, is zero, and the default end time is 1 day. Unit choices are listed in the **Units Recognized GWB Reference Manual** in the **GWB Reference Manual**; “day” is the default. Argument “off” switches the program out of kinetic mode. Examples:

```
time end = 100 years
time start 10 days, end 20 days
```

## 6.111 title

```
title <character string>
```

**Config** → **Output...**

Use the “title” command to set a title to be passed to the graphics program. Example:

```
title "Yucca Mountain Groundwater"
```

Be sure to put multiword titles in quotes.



## 6.112 unalter

**unalter** <species | mineral | solid solution | gas | surface species | **ALL**>

**Config** → **Alter Log Ks...** → *select entry(ies)* →

Use the “unalter” command to reverse the effect of having changed the log  $K$ ’s for a species, mineral, gas, or surface reaction, the selectivity coefficient for an exchange species, the  $K_d$  for a sorbed species, the  $K_f$  and  $n_f$  for a Freundlich species, or the properties of a solid solution in the current dataset. Example:

```
unalter Quartz
```

In this case, the log  $K$  values for quartz revert to those in the current thermodynamic dataset. The argument “ALL” resets the log  $K$ ’s, selectivity coefficients,  $K_d$ ’s, or  $K_f$ ’s and  $n_f$ ’s for all species, minerals, gases, and surface species, as well as the properties of all solid solutions in the dataset.

## 6.113 unsegregate

**unsegregate** <mineral(s) | **ALL**>

**Config** → **Isotopes...** → *select entry(ies)* →

Use the “unsegregate” command to remove minerals from the list of minerals to be segregated isotopically.

## 6.114 unsuppress

**unsuppress** <species, minerals, solid solutions, gases, surface species | **ALL**>

**Config** → **Suppress...**

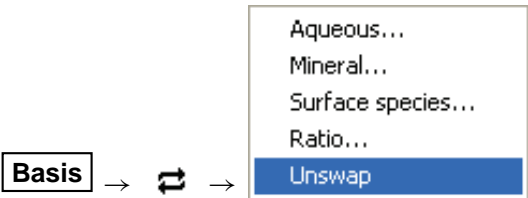
Use the “unsuppress” command (also: “include”) to include in the calculation aqueous species, surface species, minerals, solid solutions, or gases that have previously been suppressed. Examples:

```
unsuppress Quartz Albite "Albite low"
unsuppress ALL
```

The argument “ALL” clears any species, minerals, solid solutions, or gases that have been suppressed.

## 6.115 unswap

**unswap** <species | ALL>



Use the “unswap” command to reverse a basis swap. Example:

```
unswap Quartz (or unswap SiO2(aq))
```

to reverse the effect of the previous command

```
swap Quartz for SiO2(aq)
```

At this point, SiO<sub>2</sub>(aq) is back in the basis. The “ALL” argument reverses all basis swaps.

## 6.116 usgovt

**usgovt**

**Help** → **About React...**

Display the restricted rights legend that governs use, duplication, or disclosure by the U.S. Government.

## 6.117 volume

**volume** = <value | ?> <unit>

**Medium** → bulk volume  +  ▼ cm3 ▼

Use the “volume” (also “bulk\_volume”) command to cause the program to scale the initial system to a specific size. By default, the program does not scale the system. In this case, the bulk volume  $V_b$

$$V_b = V_f + V_{mn} + V_{in} + V_{stag} \quad (6.1)$$

is computed as the sum of the fluid volume  $V_f$ , mineral volume  $V_{mn}$ , inert volume  $V_{in}$ , and the volume  $V_{stag}$  of the stagnant zone, if one has been specified using the “dual\_porosity” command.

If  $X_{stag}$  is the stagnant fraction of the system, which is zero if a dual porosity model has not been invoked, the above equation can be rewritten

$$(1 - X_{stag}) V_b = V_f + V_{mn} + V_{in} \quad (6.2)$$

When you use the “volume” command to set the bulk volume  $V_b$ , the program scales the fluid volume  $V_f$  and optionally the mineral and inert volumes,  $V_{mn}$  and  $V_{in}$ , so the system fills the specified space.

The command

```
volume = 2 m3
```

for example, causes the program to scale the fluid volume  $V_f$  and optionally the mineral and inert volumes,  $V_{mn}$  and  $V_{in}$ , so the system occupies 2 m<sup>3</sup>. Default behavior, in which no scaling is performed, can be restored at any time with the command

```
volume = ?
```

The unit you use to constrain the amount of each mineral in the system controls whether or not the program adjusts that mineral’s volume. If you constrain Quartz to “10 cm<sup>3</sup>”, for example, Quartz will occupy 10 cm<sup>3</sup> after scaling. Setting Quartz in relative units such “volume%” or “mmol/m<sup>3</sup>”, in contrast, causes the program to fix the mineral’s volume relative to  $V_b$ .

Similarly if you set inert volume in an absolute unit, such as

```
inert = 50 cm3
```

the program will honor that constraint and hold  $V_{in}$  steady. The inert portion of the system in this example will occupy 50 cm<sup>3</sup> after scaling. The command

```
inert = 10 volume%
```

however, causes the program to scale the inert volume to 10% of the system’s bulk volume.



# Phase2 Commands

## 7.1 <unit>

<value> <free> <unit> <as element symbol> <basis entry>

**Basis** →  ▼ free kg ▼

To constrain the initial system, enter a command containing only the above entries. Entries may appear in any order. The qualifier “free” specifies that the constraint applies to the free rather than to the bulk basis entry. **Phase2** recognizes the following units for constraining the initial system:

*By mass or volume:*

mol	mmol	umol	nmol	
kg	g	mg	ug	ng
eq	meq	ueq	neq	
cm3	m3	km3	l	

*By concentration:*

mol/kg	mmol/kg	umol/kg	nmol/kg
molal	mmolal	umolal	nmolal
mol/l	mmol/l	umol/l	nmol/l
g/kg	mg/kg	ug/kg	ng/kg
wt%	"wt fraction"		
g/l	mg/l	ug/l	ng/l
eq/kg	meq/kg	ueq/kg	neq/kg
eq/l	meq/l	ueq/l	neq/l

*By carbonate alkalinity (applied to bicarbonate component):*

eq_acid	meq_acid	ueq_acid	neq_acid
eq_acid/kg	meq_acid/kg	ueq_acid/kg	neq_acid/kg
eq_acid/l	meq_acid/l	ueq_acid/l	neq_acid/l
g/kg_as_CaCO3	mg/kg_as_CaCO3	ug/kg_as_CaCO3	ng/kg_as_CaCO3
wt%_as_CaCO3			
g/l_as_CaCO3	mg/l_as_CaCO3	ug/l_as_CaCO3	ng/l_as_CaCO3
mol/kg_as_Ca...	mmol/kg_as_Ca...	umol/kg_as_Ca...	nmol/kg_as_CaCO3
mol/l_as_CaCO3	mmol/l_as_CaCO3	umol/l_as_CaCO3	nmol/l_as_CaCO3

*Per volume of the system:*

## GWB Command Reference

mol/cm3	mmol/cm3	umol/cm3	nmol/cm3
kg/cm3	g/cm3	mg/cm3	ug/cm3
ng/cm3			
mol/m3	mmol/m3	umol/m3	nmol/m3
kg/m3	g/m3	mg/m3	ug/m3
ng/m3			
volume%	"vol. fract."		
<i>By activity:</i>			
activity	fugacity	ratio	
pH	V	pe	
<i>By partial pressure:</i>			
Pa	MPa	atm	bar
psi			

Activity and fugacity may be abbreviated to "a" or "f". Keyword "total" reverses a setting of "free".

Use the "as" keyword to constrain mass in terms of elemental equivalents. For example, the command

```
CH3COO- = 10 umol/kg as C
```

specifies 5 umol/kg of acetate ion, since each acetate contains two carbons, whereas

```
20 mg/kg SO4-- as S
```

would specify 59.9 mg/kg of sulfate, since the ion's mole weight is about 3 times that of sulfur itself.

Examples:

```
55 mg/kg HCO3-
Na+ = 1 molal
1 ug/kg U++++
100 free cm3 Dolomite
pH = 8
Eh = .550 V
log f O2(g) = -60
HCO3- = 30 mg/kg as C
```

You may in a similar fashion constrain the concentration of a kinetic aqueous or surface complex. For example, the commands

```
kinetic AIF++
AIF++ = 1 umol/kg
```

set the concentration of the kinetic complex AIF<sup>++</sup> to 1  $\mu\text{mol kg}^{-1}$ .

## 7.2 <isotope>

```
<isotope | symbol> <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the name of any isotope system loaded in the isotope dataset (also, the isotope's symbol) to set the isotopic composition of the initial fluid, reactant species (aqueous species, minerals, end members, gases, or oxides) or segregated minerals. The composition may be set on any scale (e.g., SMOW, PDB, ...), but you must be consistent throughout the calculation.

For example, if the  $^{17}\text{O}$  isotope system has been added to the isotope dataset, you could enter:

```
oxygen-17 fluid = -10, Quartz = +15
or
17-O fluid = -10, Quartz = +15
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
oxygen-17 remove
oxygen-17 off
```

clear all settings for  $^{17}\text{O}$  isotopes from the calculation.

## 7.3 activity

```
activity <species> = <value | ?>
```

**Basis** →  ▼ activity ▼

Use the “activity” command (abbrev.: “a”) to constrain the activity of an aqueous species in the initial system. Examples:

```
activity Na+ = 0.3
log a H+ = -5
```

See also the “pH”, “Eh”, “pe”, “ratio”, “fugacity”, “fix”, and “slide” commands.

## 7.4 add

**add** <basis species>

**Basis** → add

Use the “add” command to include a basis species in the calculation. Example:

add HCO3-

See also the “swap”, “activity”, “fugacity”, “pH”, “pe”, and “Eh” commands.

## 7.5 alkalinity

**alkalinity** = <value> <unit>

**Basis** → HCO3-  ▼ mg/kg\_as\_CaCO3 ▼

Use the “alkalinity” command to constrain the total concentration of  $\text{HCO}_3^-$  to reflect the solution’s carbonate alkalinity. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “mg/kg\_as\_CaCO3” is the default. To use this option, the solution pH must be set explicitly.

## 7.6 alter

**alter** <species | mineral | gas> <log  $K$ s>

**alter** <species | mineral | gas> <poly coefs> **TminK** = <value> **TmaxK** = <value>

**alter** <surface species> <poly coefs> **TminK** = <value> **TmaxK** = <value>

**alter** <surface species> **logK** = <value> **dlogK/dT** = <value>

**alter** <exchange species> **beta** = <value>

**alter** <sorbed species> **Kd** = <value>

**alter** <sorbed species> **Kf** = <value> **nf** = <value>

**alter** <solid solution> <type> <**discrete** | **continuous**> \

<**from** <value> **to** <value> **step** <value>> <<variable> = <value>>

**Config** → **Alter Log Ks...**

**Config** → **Solid Solutions...**

Use the “alter” command to change the temperature expansion for a species’, mineral’s, or gas’ log  $K$ , to adjust the stability of a surface species, or to change the properties of a solid solution.

Temperature expansions for log  $K$ s are given in the thermo dataset either as  $T$ -tables, or by up to six coefficients of a polynomial. When the current thermo dataset uses



*T*-tables, you list replacement values at each of the principal temperatures specified in the dataset, most commonly 0°C, 25°C, 60°C, 100°C, 150°C, 200°C, 250°C, and 300°C. Example:

```
alter Anhydrite -4.3009 -4.4199 -4.7126 -5.1758 -6.2299 500 500 500
```

Values of “500” represent a lack of data at the corresponding temperature.

For a thermo dataset constructed from polynomial expansions, on the other hand, list instead up to six polynomial coefficients

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4
```

You can optionally append a temperature range of validity, in Kelvins, for the polynomial:

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4 TminK= 293 TmaxK= 383
```

Absent a range, the polynomial is taken to span the principal temperatures. You may adjust a reaction’s temperature range without specifying polynomial coefficients, but setting only coefficients resets the temperature range.

For a surface complex, use the “alter” command to set log *K*, its temperature derivative, or both

```
alter >(w)FeOCa+ logK = 6.0 dlogK/dT = 0.02
```

whereas for an ion exchange reaction, you set the selectivity coefficient  $\beta$

```
alter >X2:Ca beta = .033
```

directly, rather than as a logarithm. For a sorbing species, set  $K_d$  or, for a Freundlich species,  $K_f$ ,  $n_f$ , or both, as follows

```
alter >Pb++ Kd = .03
alter >Sr++ Kf = .015 nf = .8
```

For a solid solution, use the “alter” command to set the activity coefficient model, discrete or continuous behavior, the composition range in terms of the mole fraction of the most recently specified end member, and parameters for the chosen activity model, as described for the “solid\_solution” command. A solid solution “my\_ss” might be set as a subregular Guggenheim solution with the command

```
alter my_ss guggenheim a0 = 1 a1 = 2
```

You may prefer to use the “alter” command to adjust an entry from the thermo database, and the “solid\_solution” command to reconfigure a user-defined solution, but the two commands work interchangeably for this purpose.

Type “show alter” to list altered species or solutions and their settings; the “unalter” command reverses the process.

### 7.7 b-dot

**b-dot**

**File** → **Open** → **Thermo Data...** → thermo.tdat

The “b-dot” command (formerly: “debye-huckel”) causes the program to read the “thermo.tdat” thermodynamic dataset, which invokes the “B-dot” form of the extended Debye-Hückel equation to calculate activity coefficients for aqueous species.

### 7.8 balance

**balance** <on> <basis entry>  
**balance** <off>

**Basis** →  ▼ unit ▼ → Balance species

Use the “balance” command to specify the basis entry to be used for electrical charge balancing. The basis entry must be a charged aqueous species. By default, **Phase2** balances on  $\text{Cl}^-$ .

The command “balance off” disables **Phase2’s** charge balancing feature. In this case, the user is responsible for prescribing charge-balanced input constraints.

### 7.9 carbon-13

**carbon-13** <fluid | reactant | segregated mineral> = <value>

**Config** → **Isotopes...**

Use the “carbon-13” command (also, “13-C”) to set the  $^{13}\text{C}$  isotopic composition of the initial fluid, reactant species (aqueous species, minerals, end members, gases, or oxides) or segregated minerals. The composition may be set on any scale (e.g., PDB), but you must be consistent throughout the calculation. Example:

carbon-13 fluid = -10, Calcite = +4

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
carbon-13 remove  
carbon-13 off
```

clear all settings for  $^{13}\text{C}$  isotopes from the calculation.

See also the “<isotope>” section above, and the “hydrogen-2”, “oxygen-18”, and “sulfur-34” commands.

## 7.10 chdir

```
chdir <directory> <ALL>
```

**File** → **Working Directory...**

Use the “chdir” command (abbrev.: “work\_dir”, “cd”) to change the working directory. The program reads input scripts relative to the current working directory and writes output into it. Typing the command “chdir” without an argument causes the program to display the name of the working directory. The “ALL” argument changes the working directories of all of the GWB apps. The command

```
chdir ~
```

changes to the user’s home directory, if one is defined by the operating system.

## 7.11 clear

```
clear
```

**View** → **Clear Screen**

Use the “clear” command (abbrev.: “c”) or right-click in the text window and choose **Clear Screen** to clear it.

## 7.12 clipboard

```
clipboard
```

**Edit** → **Copy**

The “clipboard” command causes the contents of the **Results** pane to be copied to the clipboard, from where it can be pasted into various types of documents.

## 7.13 conductivity

**conductivity** <conductivity dataset>

**File** → **Open** → **Conductivity Data...**

Use the “conductivity” command to change the input file of coefficients used to calculate electrical conductivity. Example:

```
conductivity "..\my_conductivity.dat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. Beginning with GWB11, the applications compute electrical conductivity using either of two different approaches, the USGS and APHA methods; the USGS method is the default. The required coefficients are defined in the files “conductivity-USGS.dat” and “conductivity-APHA.dat”, respectively, which are installed in the same directory as the thermo datasets (commonly “\Program Files\GWB\Gtdata”).

## 7.14 couple

**couple** <redox species | element(s) | **ALL**>

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “couple” command to enable any redox coupling reactions that have been disabled with the “decouple” command. You specify one or more redox species or elements. For example, the command

```
couple Carbon
```

couple all redox reactions involving the element carbon. Argument “ALL” enables all of the coupling reactions in the thermo dataset.

## 7.15 cpr

**cpr** = <field variable | ?> <unit> <**steady** | **transient**>

**Medium** → cpr (minerals)   ▼ cal/g/C ▼

Use the “cpr” command to set the heat capacity of the rock (mineral) framework. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/g°C” is the default. This value is used during polythermal mixing simulations. The “transient” keyword causes the model to evaluate the field

variable continuously over the course of the simulation, if it is set with an equation, script, or function.

By default, this variable is set to 0.2 cal/g°C. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 7.16 cpu\_max

```
cpu_max = <value | ?>
```

**Config** → **Stepping...**

Use the “cpu\_max” command to limit the amount of computing time a simulation may take. You set the maximum computing time in seconds, or use a “?” to restore the default state, which is no prescribed limit. To see the current setting, type “show variables”.

## 7.17 cpw

```
cpw = <field variable | ?> <unit> <steady | transient>
```

**Medium** → cpw (fluid)  +  ▼ cal/g/C ▼

Use the “cpw” command to set in cal/g°C the heat capacity of the fluid. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/g°C” is the default. This value is used during polythermal mixing simulations. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function.

By default, this variable is set to 1.0 cal/g°C. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 7.18 data

```
data <thermo dataset> <verify>
```

**File** → **Open** → **Thermo Data...**

Use the “data” command to change the input file of thermodynamic data. Example:

```
data "..\my_thermo.tdat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. The “verify” option causes the program to read the named dataset only if it has not already been read.

## 7.19 decouple

```
decouple <redox species | element(s) | ALL>
```

```
Config → Redox Couples... → select couple(s) → select with...
```

Use the “decouple” command to disable the coupling reactions for one or more redox species, in order to calculate a model assuming redox disequilibrium. The redox species then become available for use as basis species and may be constrained independently of the original basis entries. You can disable as many coupling reactions as you want.

You specify either one or more redox species or elements. For example, the command

```
decouple Carbon
```

decouples all redox reactions involving the element carbon. Argument “ALL” disables all of the coupling reactions in the thermo dataset. Use the “couple” command to enable coupling reactions, once they have been disabled.

## 7.20 delQ

```
delQ = <value | ?>
```

```
Config → Stepping...
```

Use the “delQ” command to control the lengths of time steps taken in a simulation accounting for reaction kinetics. The program limits how much the ion activity product  $Q$  can change over a step, for each kinetic reaction considered. The setting for “delQ” is the projected change  $\Delta Q/Q$  allowed in the relative value of the activity product. You can set a larger value to permit longer time steps, or a smaller value to improve stability. The default setting is 0.1. Type the command with no argument or with an argument of “?” to restore the default. To see the current setting, type “show variables”.

## 7.21 delxi

```
delxi <x | y> = <value | ?> <linear | log>
```

```
Config → Stepping...
```

Use the “delxi” command to set the maximum length (in terms of reaction progress, which varies from zero to one over the course of the simulation) of the reaction step along the  $x$  or  $y$  axis, and to specify reaction stepping on a linear or logarithmic scale.

The command

```
delxi x = .1 linear
```

for example, causes the program, in the absence of other constraints on the reaction step, to take steps along  $x$  through reaction progress of .1, .2, .3, . . . , 1.0. Alternatively, the commands

```
dx_init x = .001
delxi x = .5 log
```

produce a path with steps .001, .003, .01, .03, .1, .3, and 1.0 (see the “dx\_init” command).

If you enter neither an “ $x$ ” nor a “ $y$ ” argument, the command defaults to the axis set by the “scope” command.

By default, this variable is set to 0.01 and “linear”. To restore the default settings, type the command with no argument or with an argument of “?”. To see the current settings, type “show variables”.

## 7.22 density

```
density = <value | ?>
density = <TDS | chlorinity>
density = <batzle-wang | phillips>
```

**Basis** →  advanced → density  ▼ g/cm3  
**Config** → **Options...**

You can use the “density” command to set in g/cm<sup>3</sup> the fluid density the program uses to convert compositional constraints to molality, the concentration unit it carries internally. If you set the initial Na<sup>+</sup> composition in mg/l, for example, the program needs to know the density of the initial fluid to determine Na<sup>+</sup> molality.

The program by default converts units using a density value it calculates automatically, as discussed below. This value is sufficient for most purposes, and hence it is generally not necessary to set fluid density explicitly. You might, however, want to set the density if you are working at high temperature, but your analysis is expressed per liter of solution at room temperature.

You can also use the “density” command to tell the program how to calculate the default density it uses to convert units, and the fluid density it reports in the simulation results. The program by default uses the Batzle-Wang equation to figure density, but

you can use the command "density = phillips" to select the method of Phillips et al., instead.

As well, the program normally figures density as that of an NaCl solution with the same TDS as the fluid in question, at the temperature of interest. With the command "density = chlorinity" you can tell the program to instead use the density of an NaCl solution of equivalent chlorinity.

To restore automatic calculation, type the command with no argument or with an argument of "?". To see the current setting of this variable, type "show variables".

### 7.23 dump

```
dump <x | y> <off>
```

**Config** → **Stepping...**

Use the "dump" command to eliminate minerals present at the beginning of the reaction before the program begins to trace the staging "y" path or the scanning "x" paths. Use

```
dump x
dump y off
```

If you enter neither an "x" nor a "y" argument, the command defaults to the axis set by the "scope" command.

### 7.24 dx\_init

```
dx_init <x | y> = <value | ?>
```

**Config** → **Stepping...**

Use the "dx\_init" command to set the length of the initial time step along the x or y axis. You set this value in terms of reaction progress, which varies from zero to one over the course of the simulation.

If you enter neither an "x" nor a "y" argument, the command defaults to the axis set by the "scope" command.

By default, the variable is ignored. To restore the default state, type the command with no argument or with an argument of "?". To see the current setting of this variable, type "show variables".



## 7.25 dxprint

**dxprint** <x | y> = <value | ?>

**Config** → **Output...**

Use the “dxprint” command to set along the *x* or *y* axis the interval in reaction progress (which varies from zero to one over the course of the simulation) between entries in the “Phase2\_output.txt” dataset. A value of zero causes the program to write the results after each step in reaction progress, which can produce large amounts of output. This variable setting does not apply to reactions paths with logarithmic reaction stepping (see the “delxi” command), in which case all points in reaction progress are written to the plot dataset.

If you enter neither an “*x*” nor a “*y*” argument, the command defaults to the axis set by the “scope” command.

By default, this variable is set to 0.01. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 7.26 Eh

**Eh** = <value | ?> <unit>

**Basis** →  ▼ V ▼

Use the “Eh” command to set Eh in the initial system. Example:

Eh = 200 mV

sets the system’s oxidation state to correspond to an Eh of 0.2 volt. See also the “activity”, “pH”, “pe”, “fugacity”, “fix”, and “slide” commands.

## 7.27 epsilon

**epsilon** = <value | ?>

**Config** → **Iteration...**

Use the “epsilon” command to set the convergence criterion (dimensionless) for iterating to a solution of the equations representing the distribution of chemical mass. By default, this variable is set to  $5 \times 10^{-11}$ . To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 7.28 exchange\_capacity

```
exchange_capacity = <value | ?> <units>
exchange_capacity on <type> = <value | ?> <units>
```

**Config** → **Sorbing Surfaces...** → exchange capacity  ▾ eq ▾  
 or **File** → **Open** → **Sorbing Surfaces...**

Use the “exchange\_capacity” command (abbrev.: “ex\_capacity” or “exch\_capacity”) to set the exchange capacity (i.e., the CEC) of the system when modeling ion exchange reactions or sorption according to Langmuir isotherms. For ion exchange reactions, you set units of electrical equivalents (“eq”, “meq”, and so on) or equivalents per gram of dry sediment (“eq/g”, “meq/g”, ...). For Langmuir reactions, you similarly set a value in mole units: “mol”, “mmol”, “mol/g”, “mmol/g”. If you set units per gram of sediment, the program multiplies the value entered by the mass of rock in the system (including equilibrium and kinetic minerals as well as inert volume) to get the system’s capacity.

If you read in a surface that sorbs by ion exchange or Langmuir isotherms, you must set a value for its exchange capacity. If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example

```
exchange_capacity on IonEx = .0008 eq/g
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the surface represented by the sample dataset “IonEx.sdat”, for example, is “IonEx”. You can use the “show” command to display the “type” of each active surface. See also the “surface\_data” and “inert” commands.

## 7.29 explain

```
explain <species | mineral(s) | solid solution(s) | gas(es) | surface species>
```

**Config** → **Show...** → *select species*

Use the “explain” command to get more information (such as the mole weight of a species, a mineral’s formula, mole volume, and density, or a solid solution’s end-member minerals) about species, minerals, solid solutions, and gases in the dataset. Example:

```
explain Analcime
```

## 7.30 explain\_step

```
explain_step <off>
```

**Config** → **Stepping...**

The “explain\_step” option causes the program to report on the Results pane the factor controlling the length of each time step, whenever the step size is limited by the need to maintain numerical stability.

## 7.31 extrapolate

```
extrapolate <on | off>
```

**Config** → **Options...**

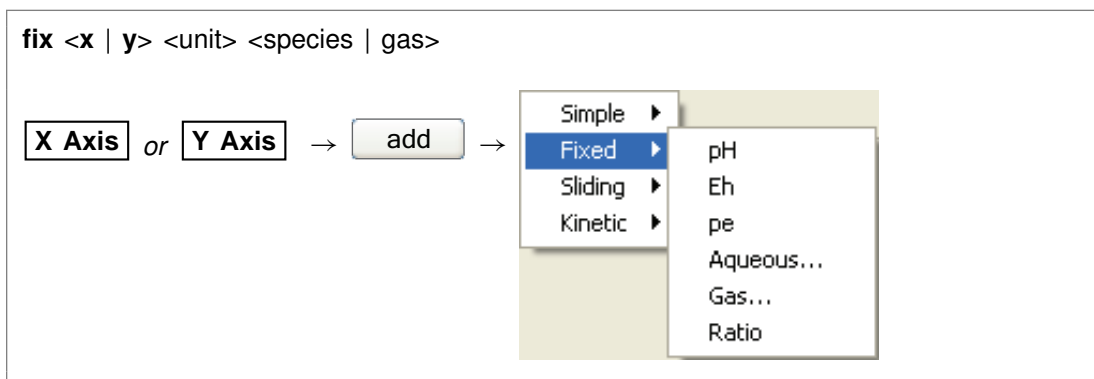
Use the “extrapolate” option to cause the program to extrapolate log  $K$ s for reactions forming aqueous species, minerals, and gases to temperatures beyond the data’s known range of validity. The option should be used with considerable care.

The temperature range of validity for a species’ log  $K$  is defined by its temperature expansion, taken from the thermo dataset. For  $T$ -table expansions, it is the span of principal temperatures at which log  $K$ s are not 500, whereas for polynomial expansions, the range is listed explicitly.

In normal operation, the program loads only species with ranges of validity encompassing the calculation’s temperature range. When the “extrapolate” option is “on”, in contrast, the program projects to the temperature of interest the log  $K$ s of species that would not otherwise be loaded.

Temperature ranges of validity can also be set in the thermo dataset for the virial coefficients used to calculate “Pitzer” and SIT activity coefficients. The “extrapolate on” option works in a similar manner in such cases, allowing virial coefficients to be used at temperatures beyond the coefficients’ known ranges of validity.

## 7.32 fix



Use the “fix” command to hold the activity of a species, fugacity of a gas, or an activity ratio constant over the course of the staging path (“y”) or the scanning paths (“x”). The <unit> can be “activity” or “fugacity” (“a” or “f” for short), “ratio”, “pH”, “pe”, or “Eh”, or it can be omitted. Examples:

```
fix x pH
fix x a H+
fix x f O2(g)
fix y ratio Ca++/Na+^2
```

If you enter neither an “x” nor a “y” argument, the command defaults to the axis set by the “scope” command.

## 7.33 flash

**flash** <on | off | fluid | system>

**Config** → **Stepping...**

Use the “flash” command to set a “flash” model in which the original fluid (or fluid and minerals, if keyword “system” is used) is removed over the course of the reaction path. Keywords “on” and “fluid” are synonymous. The command “flash” without an argument is the same as “flash fluid”. Generally, the fluid is replaced by a reactant fluid. Use

```
flash
flash system
flash off
```

## 7.34 flow-through

**flow-through** <off>

**Config** → **Stepping...**

Use the “flow-through” command to turn on or off the flow-through option by which mineral precipitates are isolated from back-reaction. Use

```
flow-through
flow-through off
```

## 7.35 flush

**flush** <off>

**Config** → **Stepping...**

Use the “flush” command to turn on or off the flush option by which fluid reactants displace existing fluid from the system over the course of the reaction path. Use

```
flush
flush off
```

## 7.36 fugacity

**fugacity** <gas> = <value | ?>

**Basis** →  ▼ fugacity ▼

Use the “fugacity” command (abbrev.: “f”) to set gas fugacities (on an atm scale) in the initial system. Examples:

```
fugacity O2(g) = .2
f CO2(g) = 0.0003
log f S2(g) = -30
```

Use “?” to unset a fugacity value: See also the “activity”, “ratio”, “pH”, “Eh”, “pe”, “fix”, and “slide” commands.

## 7.37 go

```
go <initial | single | x | y>
```

```
Run → Go  
Run → Go Single  
Run → Go Initial  
Run → Go X  
Run → Go Y
```

```
Results → Run
```

Use the “go” command to initiate calculation of a model based on the current configuration.

Used with the “initial” argument, the program calculates the initial state of the medium, but does not trace a reaction model. Typing the command with the argument “single” causes the simulation to run on a single processor, regardless of the setting for the “threads” command. The commands “go x” and “go y” causes the program to trace a single scanning (x) or staging (y) path.

## 7.38 grep

```
grep <string>
```

```
File → View → .\Phase2_output.txt → Edit → Find
```

Use the “grep” command to search for each occurrence of the given character string in the current output dataset. For example, use

```
grep Albite
```

to obtain a list of each line in “Phase2\_output.txt” (or the current output dataset, as modified by the “suffix” command) containing the string “Albite”.

## 7.39 h-m-w

```
h-m-w
```

```
File → Open → Thermo Data... → thermo_hmw.tdat
```

Use the “h-m-w” command (abbrev.: “hmw”) to set the program to calculate species’ activity coefficients by using the Harvie-Møller-Weare equations. Executing this command automatically sets the input dataset of thermodynamic data to “thermo\_hmw.tdat”. Note that dataset “thermo\_hmw.tdat” supports calculations at 25°C only.

## 7.40 heat\_source

```
heat_source = <field variable | ?> <unit> <steady | transient> \
             <Tmin = <value | min | ?> <unit>> <Tmax = <value | max | ?> <unit>>
```

```
heat_source = <off | on | reset>
```

**Medium** →  internal heat source   cal/cm<sup>3</sup>/s ▼  
 min  C ▼ max  C ▼

Use the “heat\_source” command (also: “heat\_src”) to set the rate of internal heat production within the medium. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/cm<sup>3</sup>/s” is the default. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function. By default, the program does not account for internal heat production. The “?” argument resets the default value of zero.

The “Tmin” and “Tmax” keywords (also: “temp\_max”, “temp\_min”) prescribe the allowable temperature range for the simulation. Set the keywords to specific temperatures, or to “min” or “max”, which respectively represent the lowest and highest temperatures considered in the thermo dataset, as loaded at run time. If you set the temperature range directly, it will be bracketed to that of thermo dataset. For example, the command

```
heat_source Tmin = -1000 C, Tmax = 1000 C
```

is functionally the same as

```
heat_source Tmin = 0 C, Tmax = 300 C
```

when the “thermo.tdat” dataset is loaded, since the dataset’s range is 0°C to 300°C.

The upper and lower temperature bounds serve two purposes. First, the simulation will give an error message and stop if temperature at any point in the domain falls more than 5°C less than the minimum value, or exceeds the maximum value by more than this amount. Second, unless the “extrap” option is set, the model will load for the simulation only those species for which log *K* values are available in the thermodynamic dataset over the specified temperature range.

The temperature bounds specified with the “Tmin” and “Tmax” keywords are the same as those set with the “span” command: The command

```
heat_source Tmin = 20 C, Tmax = 100 C
```

may be equivalently expressed

```
span 20 C to 100 C
```

as long as the heat source option is enabled.

Values for the keywords default to the temperature span of the thermodynamic database, as set in the database header. Keyword “off” disables the heat source, leaving the source rate and temperature range intact, and keyword “on” re-enables the heat source; “reset” disables the source and discards any settings for the source rate and temperature range.

### 7.41 help

**help** <help | subject>

**Help**

Use the “help” command (also: “man”) to display online documentation for the GWB package. The command “help help” shows a list of the types of documentation available.

### 7.42 history

**history**

**Run** → **History...**

Use the “history” command to view a list of previously executed commands. You can re-execute commands in the **History...** dialog box by double-clicking on a command, selecting one or more commands, and then choosing **Execute**, or with a history substitution command, as explained in the **Command Line Interface** chapter in the **GWB Reference Manual**.

### 7.43 hydrogen-2

**hydrogen-2** <fluid | reactant | segregated mineral> = <value>

**Config** → **Isotopes...**

Use the “hydrogen-2” command (also, “2-H”) to set the  $^2\text{H}$  isotopic composition of the initial fluid, reactant species (aqueous species, minerals, end members, gases, or oxides) or segregated minerals. The composition may be set on any scale (e.g., SMOW), but you must be consistent throughout the calculation. Example:

hydrogen-2 fluid = -120, Muscovite = -40

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.



The commands

```
hydrogen-2 remove
hydrogen-2 off
```

clear all settings for  $^2\text{H}$  isotopes from the calculation.

See also the “<isotope>” section above, and the “carbon-13”, “oxygen-18”, and “sulfur-34” commands.

## 7.44 inert

```
inert = <field variable | ?> <units>
```

**Medium** → inert volume  +  ▼ cm3 ▼

Use the “inert” command to set the volume of non-reacting space in the system. You may set a value in units of volume, including  $\text{cm}^3$ ,  $\text{m}^3$ , and  $l$ , as well as volume% and “vol. fract.”. The default setting is zero and the default unit is  $\text{cm}^3$ .

Assuming you have not set a value for the initial fluid fraction in the system using the “porosity” command, the program figures the porosity over the course of the calculation as a derived variable. Specifically, it divides the fluid volume by the sum of the fluid volume, mineral volume, and inert volume, and reports this value as a result.

When you have set a value for initial porosity with the “porosity” command, on the other hand, the program works in the contrary sense. In this case, it calculates the inert volume as that required to form a system of the specified initial porosity; the program now ignores any entry you may have set using the “inert” command.

To restore the default state, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show”.

## 7.45 isotope\_data

```
isotope_data <dataset>
```

**File** → **Open** → **Isotope Data...**

Use the “isotope\_data” command (also: “idata”) to set the name of the database containing isotope fractionation factors for species, minerals, and gases. Example:

```
isotope_data Isotope.mydata
```

## 7.46 itmax

**itmax** = <value | ?>

**Config** → **Iteration...**

Use the “itmax” command to set the maximum number of iterations that may be taken in an attempt to converge to a solution for the equations representing the distribution of chemical mass. By default, this variable is set to 999. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 7.47 Kd

**Kd** <off>

**Config** → **Options...**

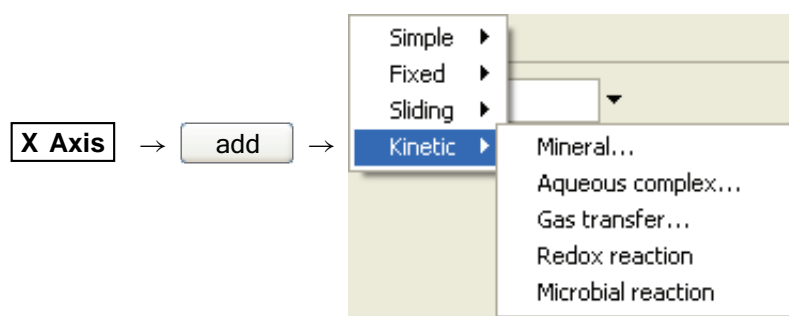
The “Kd” command controls whether the program calculates  $K_d$  distribution coefficients for sorbing components, in units of liters per kg sediment mass. This calculation requires that the mineral mass in the system (as specified for individual minerals and/or in terms of inert volume) be set correctly.

## 7.48 kinetic

```

kinetic <species | mineral | end member | gas> <variable> = <value>
kinetic <species | mineral | end member | gas> <variable> = <field variable> \
    <steady | transient>
kinetic <species | mineral | end member | gas> <apower | mpower(species)> = <value>
kinetic <redox(label)> <variable> = <value>
kinetic <redox(label)> <variable> = <field variable> <steady | transient>
kinetic <redox(label)> <apower | mpower(species)> = <value>
kinetic <microbe(label)> <variable> = <value>
kinetic <microbe(label)> <variable> = <field variable> <steady | transient>
kinetic <microbe(label)> <apower | mpower(species)> = <value>
kinetic <microbe(label)> <apower | mpower(species)> = <value> \
    <apowerA | mpowerA(species)> = <value> \
    <apowerD | mpowerD(species)> = <value>

```



Use the “kinetic” command to set variables defining a kinetic rate law for (1) dissolution or precipitation of any mineral or end member in the initial system or reactant list, (2) the association or dissociation of any aqueous or surface complex in the system modeled, (3) the transfer of gases into or out of an external reservoir, (4) a redox reaction, including those promoted by catalysis or enzymes, or (5) a microbial metabolism.

In the first three cases, you identify the kinetic reaction by the name of the mineral, end member, species, or gas involved. In the case of a redox reaction, you set a label that begins with the characters “redox”, such as “redox-1” or “redox-Fe”. For a microbial reaction, set a label that starts with “microbe”, such as “microbe-Ecoli”.

The rate law you specify in a “kinetic” command, by default, applies to the dissolution of a mineral or end member, dissociation of a complex, dissolution of a gas, or forward progress of a redox or microbial reaction. The synonymous keywords “forward”, “dissolution”, and “dissociation” set this behavior. Including in a “kinetic” command the keyword “reverse” or its synonyms “precipitation”, “complexation”, “association”, or “exsolution” invokes the opposite behavior. In this case, the rate law applies to the reverse reaction: mineral or end member precipitation, complex association, or gas exsolution.

You can append the “transient” keyword when setting the following field variables: rate constant, specific surface area, activation energy, pre-exponential factor, nucleus

area, and critical saturation index. If the variable is defined by an equation, script, or external function, it will then be re-evaluated continuously over the course of the run.

These kinetic rate law apply only to the scanning “x” paths.

See also the “react” and “remove reactant” commands.

The following paragraphs apply to all types of kinetic reactions. You set the rate constant either directly using the “rate\_con” keyword, or by setting an activation energy and pre-exponential factor with keywords “act\_en” and “pre-exp”. In the absence of promoting and inhibiting species (see next paragraph), you set the rate constant and preexponential factor in (1) mol/cm<sup>2</sup> s for mineral, end member, and gas transfer reactions, (2) molal/s or molal/cm<sup>2</sup> s (the latter when accounting for heterogeneous catalysis) for complexation and redox reactions, and (3) mol/mg s for microbial reactions. The activation energy is specified in J/mol. Example:

```
kinetic "Albite low" rate_con = 1e-15
```

You can set “rate\_con”, “act\_en”, and “pre-exp” as field variables (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

You use the “apower” or “mpower” (also “apow” or “mpow”) keyword to specify any promoting or inhibiting species in the kinetic rate law. Keyword “apower” sets the exponent of a species activity, and “mpower” the exponent of a species molality. Promoting species have positive powers, and the powers of inhibiting species are negative. For example, the command

```
kinetic "Albite low" apower(H+) = 1
```

sets H<sup>+</sup> as a promoting species, the activity of which is raised to a power of one.

You can use aqueous species, minerals (represented by activity, which is one, or molality), end members (activity or mole fraction), gas species (fugacity or partial pressure), surface complexes (molal concentration), and solvent water (activity) as promoting and inhibiting species. When setting an end member, use keyword “xpower” to set the rate law in terms of mole fraction. When setting a gas, use keyword “fpower” to set the rate law in terms of fugacity, and “ppower” to use partial pressure, instead. The generic keyword “power” sets the activity of the solvent or an aqueous species, the activity of an end member, the fugacity of a gas, and the molality of a surface species.

The “order1” and “order2” keywords set nonlinear rate laws. Keyword “order1” represents the power of the  $Q/K$  term, and “order2” represents the power of the  $(1 - Q/K)$  term.

Use the “rate\_law” keyword to set the form of the kinetic rate law for a specific mineral, end member, species, gas, redox reaction, or microbial metabolism. You may set the keyword equal to (1) a character string containing the rate law, (2) the name of a file containing a basic-like script, or (3) the name of a function in a library. The name of a file containing a rate law script must end in “.bas”. To specify a function from a library, set the name of a dynamic link library (DLL) separated from the function name

by a colon (":"), such as "rate\_laws.dll:my\_ratelaw"; the library file must end in ".dll". To return to the program's built-in rate law, enter "rate\_law = off" or "rate\_law = ?".

The following paragraphs apply to dissolution and precipitation reactions. You set the specific surface area of a kinetic mineral or end member (in  $\text{cm}^2/\text{g}$ ) with the "surface" keyword. For example,

```
kinetic "Albite low" surface = 1000
```

The "cross-affinity" option lets you use the saturation state of one mineral to model the reaction rate of another (or of an end member), as is sometimes useful for example in studying glass dissolution. To do so, you use the "xaffin" option. For (a hypothetical) example, the command

```
kinetic Quartz xaffin = Cristobalite
```

causes the program to calculate the reaction rate of quartz according to the fluid's saturation state with respect to cristobalite. The command

```
kinetic Quartz xaffin = OFF
```

turns off the option.

Finally, you use the "nucleus" and "critSI" keywords to set the area available for nucleation (in  $\text{cm}^2/\text{cm}^3$  fluid volume) and the critical saturation index above which the mineral or end member can nucleate. Each of these values, by default, is zero.

Keywords "surface", "nucleus", and "critSI" can be set as field variables (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

The following paragraphs apply to reactions for aqueous and surface complexes.

When you specify a kinetic reaction for the association of an aqueous complex or surface complex, or its dissociation, you can set the complex's initial concentration directly. The concentration can be set heterogeneously, as a field variable. If you do not specify an initial concentration, or set an entry of "?", the program takes the complex at the start of the simulation to be in equilibrium with the initial fluid.

You specify the initial concentration within a "kinetic" command or as a separate command line. For example, the commands

```
kinetic AIF++ rate_con = 3.3e-6, mpow(AIF++) = 1
AIF++ = 1 umol/kg
```

are equivalent to

```
kinetic AIF++ 1 umol/kg rate_con = 3.3e-6, mpow(AIF++) = 1
```

Either case defines a kinetic reaction for decomposition of the  $\text{AlF}^{++}$  ion pair, setting it initially to a free concentration of  $1 \mu\text{mol kg}^{-1}$ .

The following paragraphs apply to gas transfer reactions. Use the “f\_ext” keyword to specify the fugacity of the gas in question in the external reservoir, or keyword “P\_ext” to set its partial pressure. In the latter case, you may append a pressure unit; the default is bar. Keyword “contact” sets the contact area between fluid and external reservoir, in  $\text{cm}^2/\text{kg}$  of water. Example:

```
kinetic CO2(g) f_ext = 10^-3.5, contact = 10
```

Both values can be set as field variables, as described in the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**.

If you do not set a value for the gas’ external fugacity, or set “f\_ext = ?”, the program uses the fugacity in the initial fluid, at the start of the simulation, and the external fugacity.

The following paragraphs apply to redox reactions. You set the form of the redox reaction to be considered as a character string, using the “rxn” (or “reaction”) keyword. For example,

```
kinetic redox-1 rxn = "Fe++ + 1/4 O2(aq) + H+ -> Fe+++ + 1/2 H2O"
```

To specify that the reaction be promoted by a heterogeneous catalyst, set keyword “catalyst” to the name of the catalyzing mineral, or simply to “on”. In the former case, you use keyword “surface” to set the specific surface area of the catalytic mineral (in  $\text{cm}^2/\text{g}$ ). If you have set “catalyst = on”, however, you use the “surface” keyword to set total catalytic area, in  $\text{cm}^2$ . Setting “catalyst = off” disables the catalysis feature.

To set an enzymatically promoted reaction, set keyword “me” to the name of the aqueous species serving as the enzyme, or simply to the value to be used as the enzyme’s molality. In the former case, the program tracks the enzyme molality  $m_E$  over the course of the simulation from the calculated distribution of species. If you have set a numeric value for  $m_E$  using the “mE” keyword, the program uses this value directly. You may alternatively specify the enzyme species or its activity  $a_E$  using keyword “aE”, in which case variables  $m_E$ ,  $m_A$ , and  $m_P$  in the rate law are replaced by the activities  $a_E$ ,  $a_A$ , and  $a_P$ .

For an enzymatic reaction, you further set the half-saturation constants  $K_A$  and  $K_P$  for the forward and reverse reactions in molal with the “KA” and “KP” keywords. You must set a value for  $K_A$ , but may omit  $K_P$ , in which case the  $m_P/K_P$  term in the rate law will be ignored. Setting “enzyme = off” disables the enzyme feature.

The following paragraphs apply to microbial reactions. You set the form of the metabolic reaction using the “rxn” (or “reaction”) keyword, in the same manner as with redox reactions. For example,

```
kinetic microbe-1 rxn = "CH4(aq) + 2 O2(aq) -> HCO3- + H+ + H2O"
```

Set the half-saturation constants  $K_D$  and  $K_A$  for the electron donating and accepting reactions with the “KD” and “KA” keywords. These values default to zero.

You set the powers of species in the numerator of the rate law with the “mpower” keyword, as with other types of kinetic reactions. Use keywords “mpowerD” and “mpowerA” (or “mpowD” and “mpowA”) to set the powers  $p_D$ , etc., of species from the electron donating and accepting reactions, respectively, within the product functions in the rate law’s denominator. For example,

```
kinetic microbe-1 mpower(CH4(aq)) = 1, mpowerD(CH4(aq)) = 1
```

sets the power of the electron-donating species  $\text{CH}_4(\text{aq})$  to one in both the rate law numerator and denominator. Keywords “PKD” and “PKA” set the overall powers  $p_{KD}$  and  $p_{KA}$  of the electron donating and accepting terms in the denominator of the rate law; by default, these are one.

You set the free energy  $\Delta G_{\text{ATP}}$  of ATP hydrolysis (in kJ/mol) with the “ATP\_energy” keyword, and the value of  $n_{\text{ATP}}$  with keyword “ATP\_number”. These values default to zero.

Use the “biomass” keyword to set the initial biomass concentration, in mg/kg. You can set this value as a field variable (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

The “growth\_yield” keyword sets the microbe’s growth yield in mg biomass/mol of reaction progress, and “decay\_con” sets its decay constant in  $\text{s}^{-1}$ ; both values default to zero.

## 7.49 log

```
log <variable> = <value>
```

Use the “log” command to set variables on a logarithmic scale. Examples:

```
log fugacity O2(g) = -65
log activity U++++ = -10
```

## 7.50 mobility

**mobility** = <surface type> <field variable> <**steady** | **transient**>

**Config** → **Sorbing Surfaces...** → mobility    ▼

or **File** → **Open** → **Sorbing Surfaces...**

Use the “mobility” command to set up a complexing surface in your model as a mobile colloid. A mobile colloid is composed of the mineral (or minerals) associated with a complexing surface, as well as the ion complexes present on that surface. Only datasets with model type “two-layer”, “three-layer”, or “cd-music” as set in the dataset header are surface complexation models, and hence only those datasets can be used to form a mobile colloid.

Mobility refers to the fraction of the surface in question that can move in the model by advection and dispersion. A surface with a mobility of one moves freely, whereas a mobility of zero sets the surface to be stationary. Intermediate values arise, for example, when some of the surface is attached to the medium, or when colloid motion is impeded by electrostatic interactions. By default in the software the mobility of any surface is zero.

To set a mobile colloid, begin by reading in a surface complexation dataset using the “surface\_data” command. Then, use the “mobility” command, referencing the surface’s label, to set the colloid’s mobility. The label is given at the head of the surface dataset, on a line beginning “Surface type”. The label in dataset “FeOH.sdat”, for example, is “HFO”. If you omit the label, the program will assume you are referring to the surface complexation dataset most recently read.

You can define the mobility as a field variable, which means you can have the program calculate mobility using an equation, script, or compiled function you provide. When you set the “transient” keyword, the program upon undertaking each time step in the simulation evaluates mobility at each nodal block. In the “steady” case, which is the default, the program evaluates mobility at each block just once, at the start of the run.

Example:

```
surface_data FeOH.sdat
mobility HFO = 100%
```

where “HFO” is the label for the surface defined by dataset “FeOH.sdat”.

Restore the default behavior of immobility by entering a command such as

```
mobility HFO ?
```



## 7.51 no-precip

```
no-precip <off>
```

**Config** → **Options...**

Use the “no-precip” command (also: “noprecip”) to prevent new minerals from precipitating over the course of a simulation. By default, they are allowed to precipitate. Use:

```
no-precip
no-precip off
```

See also the “precip” command.

## 7.52 nswap

```
nswap = <value | ?>
```

**Config** → **Iteration...**

Use the “nswap” command to set the maximum number of times that the program may swap entries in the basis in an attempt to converge to a stable mineral assemblage. By default, this variable is set to 30. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 7.53 Nx

```
Nx = <value | ?>
```

**Config** → **Output...**

Use the “Nx” command to set the number of entries along the x axis in the “Phase2\_plot.p2p” dataset. The minimum is 3 entries.

By default, this variable is set to 301. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 7.54 Ny

```
Ny = <value | ?>
```

**Config** → **Output...**

Use the “Ny” command to set the number of entries along the *y* axis in the “Phase2\_plot.p2p” dataset. The minimum is 3 entries.

By default, this variable is set to 301. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

### 7.55 oxygen-18

```
oxygen-18 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “oxygen-18” command (also, “18-O”) to set the <sup>18</sup>O isotopic composition of the initial fluid, reactant species (aqueous species, minerals, end members, gases, or oxides) or segregated minerals. The composition may be set on any scale (e.g., SMOW), but you must be consistent throughout the calculation. Example:

```
oxygen-18 fluid = -10, Quartz = +15
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
oxygen-18 remove  
oxygen-18 off
```

clear all settings for <sup>18</sup>O isotopes from the calculation.

See also the “<isotope>” section above, and the “carbon-13”, “hydrogen-2”, and “sulfur-34” commands.

### 7.56 p2plot

```
p2plot <p2plot options>
```

**Run** → **P2plot**

The “p2plot” command launches program **P2plot** from the **Phase2** command line, invoking input from the current graphics interface dataset (by default, “Phase2\_plot.p2p”; see the “suffix” command). For example,

```
p2plot -graph 2d
```

starts **P2plot** in 2D diagram mode.

## 7.57 pause

pause

Use the “pause” command to cause **Phase2** to pause temporarily during input. This command is useful when you are debugging scripts.

## 7.58 pe

pe = <value | ?>

**Basis** →  ▼ pe ▼

Use the “pe” command to set oxidation state in the initial system in terms of pe. Example:

pe = 10

is equivalent to

log activity e- = -10

where “e-” is the electron. Use “?” to unset a pe value: See also the “activity”, “Eh”, “pH”, “fugacity”, “fix”, and “slide” commands.

## 7.59 permeability

permeability <intercept = field variable | ?> <unit> <steady | transient> \  
 <porosity = field variable | ?> <steady | transient> \  
 <mineral = field variable | ?> <steady | transient>

**Medium** → A (porosity)  +  ▼  
 B (intercept)  +  ▼ darcy ▼

You use the “permeability” command to set the correlation by which the program calculates sediment permeability. **Phase2** calculates this value as a reported variable consistent with **X1t** and **X2t**, but does not use it in its calculations.

The correlation gives log permeability in any of the units listed in the **Units Recognized** chapter of the **GWB Reference Manual** (darcys by default) as a linear function of the porosity (expressed as a volume fraction) of a nodal block and, optionally, the

## GWB Command Reference

---

volume fractions of one or more minerals. The “transient” keyword causes the model to evaluate the coefficient in question continuously over the course of the simulation, if it is set with an equation, script, or function.

Examples:

```
permeability intercept = -11 cm2 porosity = 15
permeability Kaolinite = -8
```

The latter command adds a term for the mineral Kaolinite to the existing correlation. To remove a term from the correlation, set a value of “?”. The entry

```
permeability Kaolinite = ?
```

for example, removes the correlation entry for that mineral.

The default correlation is

$$\log k = -5 + 15\phi$$

where  $k$  is permeability in darcys and  $\phi$  is porosity (expressed as a fraction).

### 7.60 pH

```
pH = <value | ?>
```

```
Basis →  ▼ pH ▼
```

Use the “pH” command to set pH in the initial system. Example:

```
pH = 5
```

is equivalent to

```
log activity H+ = -5
```

Use “?” to unset a pH value: See also the “activity”, “Eh”, “pe”, “fugacity”, “fix”, and “slide” commands.

### 7.61 phrqpitz

```
phrqpitz
```

```
File → Open → Thermo Data... → thermo_phrqpitz.tdat
```

Use the “phrqpitz” command to set the program to calculate species’ activity coefficients using the Harvie-Møller-Weare equations, as implemented in the USGS program PHRQPITZ. Executing this command automatically sets the input dataset of thermodynamic data to “thermo\_phrqpitz.tdat”. Note that dataset “thermo\_phrqpitz.tdat” is primarily intended to support calculations at or near 25°C.

## 7.62 pitz\_dgamma

```
pitz_dgamma = <value | ?>
```

**Config** → **Iteration...**

Use the “pitz\_dgamma” command to control the relative change in an activity coefficient’s value the program allows during each Newton-Raphson iteration, when the Harvie-Møller-Weare activity model has been invoked. By default, the program allows a 10% change, which corresponds to a value of 0.1.

## 7.63 pitz\_precon

```
pitz_precon = <value | ?>
```

**Config** → **Iteration...**

Use the “pitz\_precon” command to control the maximum number of passes the program takes through the pre-conditioning loop before beginning a Newton-Raphson iteration, when a virial activity model (Harvie-Møller-Weare or SIT) has been invoked. By default, the program makes up to 10 passes. In cases of difficult convergence, counter-intuitively, it can sometimes be beneficial to decrease this value.

## 7.64 pitz\_relax

```
pitz_relax = <value | ?>
```

**Config** → **Iteration...**

The “pitz\_relax” command controls under-relaxation when evaluating the Harvie-Møller-Weare equations. The program at each Newton-Raphson iteration assigns activity coefficients as a weighted average of the newly calculated value and the corresponding value at the previous iteration level. Setting pitz\_relax to zero eliminates under-relaxation, so the newly calculated values are used directly; a value of one, in contrast, should be avoided because it would prevent the activity coefficients from being updated. By default, the program carries an under-relaxation factor of 0.5.

## 7.65 plot

```
plot <character | binary> <on | off>
```

**Config** → **Output...**

Use the “plot” command to set the format of the “Phase2\_plot.p2p” dataset. The dataset is written in XML, a standard format that is easy to parse for use with alternative plotting programs. Numerical data in the dataset can be represented in either standard decimal notation (keyword “character”) for user readability or a binary encoding (keyword “binary”) that maintains full precision of data. The default format, XML with binary encoded data, also zips the output file to reduce output size and improve file opening speed. The command “plot off” causes **Phase2** to bypass writing calculation results to the “Phase2\_plot.p2p” dataset, which is used to pass input to **P2plot**. By default, the program writes output to the dataset. The command “plot on” (or just “plot”) re-enables the output. To see the current setting, type “show print”.

## 7.66 pluses

```
pluses <off | on | multicore | banner>
```

**Config** → **Stepping...**

The “pluses” command controls whether the program writes a plus sign (“+”) to the screen each time it converges to a solution of the chemical equations in a reaction path (keyword “on”); by default, it does not (keyword “off”). You can also set the program to write a distinct symbol for each thread active in tracing the simulation (keyword “multicore”), allowing you to observe the model as it runs in parallel. You can also specify the program write a banner giving details such as the number of iterations required for convergence and the final residuals (keyword “banner”). The latter option produces unwieldy amounts of output, except for simple simulations.

## 7.67 polymorphs

```
polymorphs <mineral>
```

**Config** → **Show...** → Minerals → *select mineral*

Use the “polymorphs” command to list each polymorph (mineral identical in composition to the specified mineral) in the current database. Example:

```
polymorphs Albite
```

## 7.68 porosity

**porosity** = <field variable | ?>

**Medium** → porosity

Use the “porosity” command to set (as a volume fraction) the initial porosity of the system. Porosity, the fraction of the system occupied by fluid, is the ratio of fluid volume to the sum of fluid, mineral, and inert volume.

The examples

```
porosity = 0.30
porosity = 30%
```

are equivalent.

When you specify the porosity, the program will figure the difference between the volume of a system of the given porosity and fluid volume, and the volume taken up initially in the system by minerals and fluid. The program assigns this difference as inert, non-reactive volume (see the “inert” command). In this case, the program ignores any settings that may have been made with the “inert” command.

When you do not specify an initial porosity with the “porosity” command, on the other hand, the program calculates it from volumes in the system of fluid, minerals, and inert space. To restore this default behavior, enter the command with an argument of “?”.

## 7.69 precip

**precip** <off>

**Config** → **Options...**

Use the “precip off” command to prevent new minerals from precipitating over the course of a simulation. By default, they are allowed to precipitate. Use

```
precip
precip off
```

See also the “no-precip” command.

## 7.70 press\_model

```
press_model <Tsonopoulos | Peng-Robinson | Spycher-Reed | default | off>
```

**Config** → **Options...**

The “press\_model” command (also: “pressure\_model”) lets you control the method used to calculate fugacity coefficients and gas partial pressures. Three nonideal pressure models are coded in the software: Tsonopoulos, Peng-Robinson, and Spycher-Reed, as described in the **GWB Essentials Guide**.

By default, the pressure model is taken from the header lines of the thermo dataset in use, but you can use the “press\_model” command to override the default setting. Keywords “Tsonopoulos”, “Peng-Robinson”, and “Spycher-Reed” set the pressure model directly (you need only enter the first three letters), whereas “default” returns to the setting in the thermo dataset, and “off” (also: “ideal”) disables the feature, forcing all fugacity coefficients to one.

Examples:

```
press_model Peng-Robinson
press_model default
```

## 7.71 print

```
print <option> = <long | short | none>
print <off | on>
print <numeric | alphabetic>
```

**Config** → **Output...**

Use the “print” command (also: “printout”) to control the amount of detail to be written into the “Phase2\_output.txt” dataset. For example, the dataset can contain information about each aqueous species, information on only species with concentrations greater than  $10^{-8}$  molal, or no species information.

By default, the program does not produce these datasets, which can become quite lengthy. The command “print on” (or just “print”) causes them to be written, and “print off” disables them.



Options, which may be abbreviated to three letters, and their default settings are:

species	short
surfaces	long
saturations	short
gases	long
basis	none
orig_basis	long
elements	long
reactions	none
stagnant	none

The “print” command can also be used to arrange entries in the output dataset either numerically or alphabetically:

print numeric
print alphabetic

To see the current print settings, type “show print”.

## 7.72 pwd

<b>pwd</b>
<b>File</b> → <b>Working Directory...</b>

The “pwd” command returns the name of the current working directory. The command has the same effect as typing “show directory”. See the “chdir” command.

## 7.73 quit

<b>quit</b>
<b>File</b> → <b>Quit</b>

Use the “quit” command (also: “end”, “q”) to exit the program. Touching **Ctrl+Q** serves the same purpose as the “quit” command.

## 7.74 ratio

<b>ratio</b> <species ratio> = <value   ?>
<b>Basis</b> → <input type="text"/> ▼ ratio ▼

## GWB Command Reference

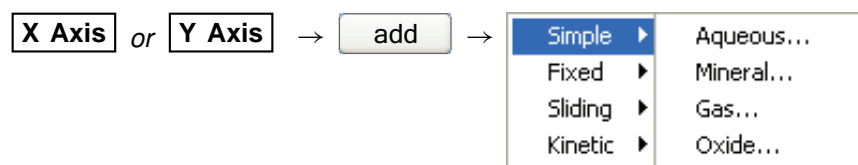
Use the “ratio” command to constrain an activity ratio in the initial system. Example:

```
swap Ca++/Na+^2 for Ca++  
ratio Ca++/Na+^2 = 0.2
```

Use “?” to unset a ratio value: See also the “activity”, “pH”, “Eh”, “pe”, “fugacity”, “fix”, and “slide” commands.

### 7.75 react

```
react <x | y> <amount> <unit> <as <element symbol>> \  
  <species | mineral | end member | gas | oxide> <cutoff> = <value>
```



Use the “react” command (abbrev.: “rct”) to define the reactants for the current simulation. To set a kinetic rate law for a reactant, use the “kinetic” command.

Use keyword “x” to set the reactant along the scanning paths, and “y.” for the staging path). If you enter neither an “x” nor a “y” argument, the command defaults to the axis set by the “scope” command.

Units for the amount of reactant to add over a reaction path can be:

mol	mmol	umol	nmol	
kg	g	mg	ug	ng
eq	meq	ueq	neq	
cm3	m3	km3	l	
mol/kg	mmol/kg	umol/kg	nmol/kg	
molal	mmolal	umolal	nmolal	
mol/l	mmol/l	umol/l	nmol/l	
g/kg	mg/kg	ug/kg	ng/kg	
wt%	"wt fraction"			
g/l	mg/l	ug/l	ng/l	
eq/kg	meq/kg	ueq/kg	neq/kg	
eq/l	meq/l	ueq/l	neq/l	

Units of mass or volume can be expressed per volume of the porous medium. Examples:

mol/cm3	g/cm3
mmol/m3	ug/m3
volume%	"vol. fract"

Units of mass or volume on the scanning paths can be set as absolute rates by appending “/s”, “/day”, “/yr”, or “/m.y.”. For example,

mmol/s	g/day	cm <sup>3</sup> /yr
mol/kg/s	mg/kg/day	cm <sup>3</sup> /kg/yr

Use the “as” keyword to specify reactant masses as elemental equivalents. For example, the command

```
react x 10 umol/kg CH3COO- as C
```

specifies 5 umol/kg of acetate ion on the scanning paths, since each acetate contains two carbons, whereas

```
react x 20 mg/kg SO4-- as S
```

would cause the program to add 59.9 mg/kg of sulfate, since the ion’s mole weight is about 3 times that of sulfur itself.

You can set a cutoff to limit the amount of a reactant. For example, if you set the amount of a reactant to two moles and set a cutoff of one, then **Phase2** will add one mole of the reactant over the first half of the path and none over the second half. Enter the cutoff value in the same units as the amount of reactant. Examples:

```
react y 10 grams Quartz
react x 1e-2 mol Muscovite cutoff = .5e-2
react x .01 mol/day HCl
```

See also the “kinetic” and “remove reactant” commands.

## 7.76 reactants

```
reactants <x | y> times <value>
```

X Axis or  Y Axis → reactants times  ▼

Use the “reactants” command to vary the total amounts of the reactants on the scanning paths (keyword “x”) or the staging path (“y”) by a given factor. If you enter neither an “x” nor a “y” argument, the command defaults to the axis set by the “scope” command.

Example:

```
reactants x times 1/10
```

## 7.77 read

```
read <dataset>
```

**File** → **Open** → **Read Script...**

Use the “read” command to begin reading commands from a script stored in a dataset. Example:

```
read Seawater
```

Control returns to the user after the script has been read, unless the script contains a “quit” command. You can also use the “read” command in place of the “data” or “surface\_data” command to read a thermo or surface reaction dataset.

When typing a “read” command, you can use the spelling completion feature to complete dataset names: touch “[tab]” or “[esc]” to cycle through the possible completions, or **Ctrl+D** to list possible completions.

## 7.78 remove

```
remove <basis specie(s)> <solid solution(s)> <reactant(s)>
```

```
remove basis <basis specie(s)>
```

```
remove solid_solution <solid solution(s)>
```

```
remove reactant <x | y> <reactant(s)>
```

**Basis** or **X Axis** or **Y Axis** or

**Config** → **Solid Solutions...** →

*select entry(ies)* →

Use the “remove” command (also: “rm”) to eliminate one or more basis entries or reactants from consideration in the calculation. Removing reactants without the reactant keyword removes reactants from both the scanning paths (keyword “x”) and staging path (“y”). Example:

```
remove Na+
remove Quartz Calcite
remove reactant x H2O
```

Components can be reentered into the basis using the “swap”, “add”, “activity”, and “fugacity” commands. You can also use the “remove” command to remove solid solutions configured in a run (but not those set in the thermo dataset):

```
remove solid_solution mySS
```

## 7.79 reset

```
reset
reset system
reset reactants
reset variables
```

**File** → **Reset Configuration**

Use the “reset” command to begin defining the chemical system again with a clean slate. Your current settings will be lost, and all options will be returned to their default states. The command, however, does not alter the setting for the thermo dataset. The “reset system” command resets only the initial system. Similarly, typing “reset reactants” resets the reactant system, and “reset variables” sets each settable variable to its default value.

## 7.80 resume

```
resume
```

**File** → **Resume**

Use the “resume” command to restore the configuration from when you last exited the program. The program configuration is stored in a file “phase2\_resume.rea” in your profile directory (e.g., “c:\Documents and Settings\jones\Application Data\GWB”). Your current configuration is discarded.

## 7.81 save

```
save <dataset> <hex>
```

**File** → **Save As...**

Use the “save” command to write the current chemical system into a dataset in **Phase2** format commands. The dataset can be used as an **Phase2** input script. Examples:

```
save
save kspar.rea
```

If you don’t specify a dataset name, **Phase2** will ask for one. The optional keyword “hex” causes the program to output numbers as hexadecimal values.

## 7.82 scope

```
scope = <x | y>
```

Use the “scope” command to set the default axis direction for commands that can configure the scanning ( $x$ ) or staging ( $y$ ) axis. Those commands include “delxi”, “dx\_init”, “dxprint”, “fix”, “kinetic”, “react”, “reactant times”, “remove”, “slide”, “step\_increase”, “step\_max”, and “temperature”. By default, the scope points to the  $x$  axis.

The scope command allows you to enter a series of commands without repeatedly specifying the target axis. For example, typing

```
scope y
react Quartz
react K-feldspar
```

is equivalent to entering the commands

```
react y Quartz
react y K-feldspar
```

To see the current scope from the command line, enter “show scope”.

## 7.83 script

```
script
script end
```

Use the “script” command to mark the beginning, and optionally the end, of a control script. Control scripts differ from standard input files in that they can contain not only **Phase2** commands, but control structures such as loops and if-else branches. Control scripts follow the Tcl syntax, described in [www.tcl.tk](http://www.tcl.tk) and [mini.net/tcl](http://mini.net/tcl), as well as several widely available textbooks.

Within a control script, filenames are written with double rather than single backslashes. For example, a “read” command might appear as

```
read GWB_files\\My_file.rea
```

within a control script.

## 7.84 segregate

```

segregate <mineral(s)>
segregate <mineral> <value>
segregate <mineral> <initial value> <final value>
segregate <mineral> <value> Xi = <value> <value> Xi = <value>

```

**Config** → **Isotopes...** →

**Config** → **Isotopes...** → *select entry(ies)* →

The “segregate” command causes minerals to be isolated from isotopic exchange over the course of a reaction path. By default, a mineral in the equilibrium system remains in isotopic equilibrium with the fluid and other minerals. A segregated mineral, on the other hand, changes in isotopic composition only when it precipitates from solution; it alters the system’s composition only if it dissolves. Example:

```
segregate Quartz Calcite "Maximum Microcline"
```

Optionally, a fraction of a mineral’s mass may be isotopically segregated, and that fraction may vary linearly with reaction progress. Examples:

```

segregate Quartz 100%, Muscovite 7/10
segregate Ca-Saponite 100% 0%
segregate Ca-Saponite 80% Xi = .3, 20% Xi = .7

```

In the latter example, the program segregates 80% of the mass of Ca-Saponite until the reaction progress variable Xi reaches .3, decreases the segregated fractionation until it reaches 20% when Xi equals .7, and then holds the value constant until the end of the path. To display the isotopically segregated minerals, type “show isotopes”.

Note, you use the name of the corresponding mineral to segregate a solid solution end member.

## 7.85 show

```

show <option>
show <aqueous | minerals | solid_solutions | gases | oxides | surfaces> \
  <with | w/> <basis entry | string>

```

**Config** → **Show...**

## GWB Command Reference

---

Use the “show” command to display specific information about the current system or database. The options are:

show	show all	show altered
show aqueous	show basis	show commands
show couples	show directory	show elements
show gases	show initial	show isotopes
show minerals	show oxides	show printout
show reactants	show scope	show show
show solid_solutions	show suppressed	show surfaces
show system	show variables	

The command “show show” gives a list of show command options. When you type “show aqueous” (or “minerals”, “solid\_solutions”, “gases”, “oxides”, or “surfaces”), the program lists all entries of that type in the thermo database. The “solid\_solutions” option additionally includes solutions defined locally. A long form lets you limit the query to entries composed of a particular basis species or containing a text string in the name:

```
show aqueous with Al+++  
show minerals w/ chal
```

There is also a compound form of the “show couples” command:

```
show coupling reactions
```

This command produces a complete list of the redox couples, in reaction form.

### 7.86 simax

```
simax = <value | ?>
```

```
Config → Iteration...
```

The “simax” command sets in molal units the maximum value of the stoichiometric ionic strength used in calculating water activity when Helgeson’s B-dot Debye-Hückel model is employed. By default, this variable is set to 3 molal. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.



## 7.87 slide

**slide** <x | y> <unit> <species | gas> **to** <value>

Use the “slide” command to linearly adjust the activity of the specified species, fugacity of the gas, or an activity ratio toward <value>. If keyword “x” is set, <value> is attained at the end of the scanning paths, whereas with keyword “y”, the end of the staging path marks the endpoint. Note that the interpolation is made linearly on the logarithm of activity or fugacity if <value> is set as a log, and that <unit> can be “activity” or “fugacity” (“a” or “f” for short), “ratio”, “pH”, “pe”, or “Eh”, or omitted. Examples:

```
slide x pH to 5
slide x activity Cl- to 2/3
slide x f CO2(g) to 10^-3.5
slide y log f O2(g) to -65
```

## 7.88 solid\_solution

**solid\_solution** <name = <label>> <type> <discrete | continuous> \  
 <<mineral> tag = <label>> <contains> <<mineral> tag = <label>> \  
 <from <value>> <to <value>> <step = <value>> <<variable> = <value>>  
**solid\_solution** <solid solution> <options>  
**solid\_solution remove** <solid solution>

**Config** → **Solid Solutions...** →

**Config** → **Solid Solutions...** → *select entry(ies)* →

Use the “solid\_solution” command (abbrev.: “solid\_soln”) to define binary solid solutions in addition to any provided in the current thermo database. Supply a name with the “name” keyword, or simply set one as the first argument.

You can select the type of solid solution, either “ideal” (the default), “site\_mixing”, “guggenheim”, “regular”, “cubic”, or “third\_order”. For a site-mixing solution, set the

site number “bsite”, which defaults to 1. For a Guggenheim solution, set dimensionless parameters “a0”, “a1”, and “a2” or a temperature expansion for their J/mol counterparts using coefficients “p1”-“p9”, each of which defaults to zero. For the other nonideal models, similarly, set coefficients “p1”, “p2”, etc., as appropriate. For more information, see **Solid solutions** under **Configuring the programs** in the **GWB Essentials Guide** and the **Thermo Datasets** chapter in the **GWB Reference Manual**. You can as well choose whether the solid solution is to be treated as continuous, the default option, or as a sequence of “discrete” minerals.

You need to specify two end members for a solid solution, each associated with an abbreviated “tag”. The tag cannot be the name of another entry in the database. You can set a composition range in terms of the mole fraction of the most recently referenced end member, using keywords “from” and “to”, as well as a compositional “step” separating tranches of a discrete solution; by default, composition ranges from zero to one and the step is one-twentieth of the range. Keyword “contains” may be inserted for clarity. Examples:

```
solid_solution name = my_ss Calcite contains Strontianite from 0% to 10% a0= 0.05
solid_solution my_ss Calcite Strontianite from 0 to .1 a0= 0.05
solid_solution Biotite site_mixing Annite Phlogopite bsite= 3
```

If you don’t specify a solid solution’s name, the app will create one automatically using the tags for the two end members, separated by a hyphen. If you don’t set a tag for an end member, the app will pick up the first six characters of the corresponding mineral’s name. For example, the command

```
solid_solution Calcite contains Strontianite
```

produces a solid solution named “Calcit-Stront”, whereas typing

```
solid_solution Calcite tag Cc contains Strontianite tag Str
```

yields the name “Cc-Str”. To redefine the properties of a solid solution, enter the name followed by the properties to be changed:

```
solid_solution my_ss a0 = 1 a1 = 2
```

(You may prefer to use the “alter” command to change solutions defined within the thermo database; the two commands are functionally equivalent.) The command

```
solid_solution remove my_ss
```

discards a previously defined solution.

## 7.89 sorbate

```
sorbate <exclude | include>
```

```
Config → Options...
```

Use the “sorbate” command to tell the program, when considering sorption onto surfaces (see the “surface\_data” command), whether to include or exclude sorbed species in figuring the composition of the initial system. By default, the program does not include sorbed species in this calculation. If you set the  $\text{Ca}^{++}$  concentration to 15 mg/kg, for example, the initial system would contain that amount in the fluid and an additional amount sorbed onto mineral surfaces. If you type the command “sorbate include”, however, that amount would apply to the sum of the  $\text{Ca}^{++}$  sorbed and in solution.

## 7.90 span

```
span <value | min | ?> <unit> <to> <value | max | ?> <unit>  
span <off | on | reset>
```

```
Config → Options...
```

The “span” command lets you set a temperature range across which stability must be defined for each species considered in the calculation. The lower limit may be identified with keyword “from” or “Tmin”, and the upper denoted by “to” or “Tmax”. A value of “?” unsets the bound in question.

With  $T$ -table datasets, simply setting “span = on” without specifying a range forces the program to load species’ log  $K$ s in polynomial mode. In other words, it will take log  $K$ s from polynomial fits against temperature to the stability constants in the thermo dataset, even if the calculation is performed entirely at one of the dataset’s principal temperatures. Otherwise, an isothermal calculation run at one of the principal temperatures (commonly 0°C, 25°C, 60°C, ...) takes log  $K$ s for aqueous species, minerals, and so on directly from the values given in the thermo dataset.

Temperature ranges of validity can also be set in the thermo dataset for the virial coefficients used to calculate “Pitzer” and SIT activity coefficients. The “span” command works in a similar manner in such cases, loading only virial coefficients with validity that spans the calculation’s temperature range.

As an example of using the “span” command, you may wish to compare results calculated at 25°C with those figured at 30°C. Since log  $K$  for a number of species in the thermo dataset is given at only 25°C, unless the “extrap” option is set, the former calculation may well take into account a somewhat larger set of species than the latter. Entering the command

```
span 25 C to 30 C
```

assures the same set of species is loaded in each case, and that both runs take log  $K$ s from polynomial fits.

Set the bounds to specific temperatures, or to “min” or “max”, which respectively represent the lowest and highest temperatures considered in the thermo dataset, as loaded at run time. If you set the temperature range directly, it will be bracketed to that of the thermo dataset. For example, the command

```
span -1000 C to 1000 C
```

is functionally the same as

```
span 0 C to 300 C
```

when the “thermo.tdat” dataset is loaded, since the dataset’s range is 0°C to 300°C.

The temperature bounds set here are the same as those set with the “heat\_source” command: The command

```
span 20 C to 100 C
```

may be equivalently expressed

```
heat_source Tmin = 20 C, Tmax = 100 C
```

as long as the heat source option is enabled.

Keyword “off” disables the feature, leaving values for the limits intact, and keyword “on” re-enables the feature. The “reset” keyword clears the limits, disabling the feature. By default, the span feature is disabled.

### 7.91 start\_date

```
start_date <value | off>
```

**Config** → **Stepping...**

Use the “start\_date” command to set an explicit starting date of the reaction. This can be used to coordinate the plotting of dated scatter data samples stored in a **GSS** spreadsheet on the reaction path in **P2plot**. The date should be in the format “MM/DD/YYYY”. Use “off” to return to the default of not set.

```
start_date 10/30/2008  
start_date off
```

## 7.92 start\_time

```
start_time <value | off>
```

**Config** → **Stepping...**

Use the “start\_time” command to set an explicit starting time of the reaction. This can be used to coordinate the plotting of timed scatter data samples stored in a **GSS** spreadsheet on the reaction path in **P2plot**. The time should be in the format “HH:MM:SS”. Use “off” to return to the default of not set.

```
start_time 11:30:00
start_time off
```

## 7.93 step\_increase

```
step_increase <x | y> = <value | ?>
```

**Config** → **Stepping...**

Use the “step\_increase” command to set along the *x* or *y* axis the greatest proportional increase, from one step to the next, in the size of the time step. This variable does not apply to reaction paths with logarithmic reaction stepping (see variable “delxi”).

If you enter neither an “*x*” nor a “*y*” argument, the command defaults to the axis set by the “scope” command.

By default, this variable is set to 1.5. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 7.94 step\_max

```
step_max <x | y> = <value | ?>
```

**Config** → **Stepping...**

Use the “step\_max” command to limit along the *x* or *y* axis the number of reaction steps the program may take to trace a simulation.

If you enter neither an “*x*” nor a “*y*” argument, the command defaults to the axis set by the “scope” command.

Use a “?” to restore the default state, which is no prescribed limit. To see the current setting, type “show variables”.

## 7.95 suffix

```
suffix <string>
```

**Config** → **Output...**

Use the “suffix” command to alter the names of the output datasets (“Phase2\_output.txt”, “Phase2\_isotope.txt”, “Phase2\_plot.p2p”) by adding a trailing string. Example:

```
suffix _run2
```

produces output datasets with names such as “Phase2\_output\_run2.txt”.

## 7.96 sulfur-34

```
sulfur-34 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “sulfur-34” command (also, “34-S”) to set the <sup>34</sup>S isotopic composition of the initial fluid, reactant species (aqueous species, minerals, end members, gases, or oxides) or segregated minerals. The composition may be set on any scale (e.g., CDT), but you must be consistent throughout the calculation. Example:

```
sulfur-34 fluid = +45, H2S(g) = -2
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
sulfur-34 remove
sulfur-34 off
```

clear all settings for <sup>34</sup>S isotopes from the calculation.

See also the “<isotope>” section above, and the “carbon-13”, “hydrogen-2”, and “oxygen-18” commands.

## 7.97 suppress

```
suppress <species, minerals, solid solutions, gases, surface species | ALL>
```

**Config** → **Suppress...**

Use the “suppress” command (also: “kill”) to prevent certain aqueous species, surface species, minerals, solid solutions, or gases from being considered in a calculation. Example:

```
suppress H3SiO4- Quartz "Maximum Microcline"
```

prevents the three entries listed from being loaded from the database. Typing “suppress ALL” suppresses all of the minerals and solid solutions in the thermodynamic database, as well as any solid solutions defined locally.

The “unsuppress” command reverses the process. To suppress all but a few minerals, you could type

```
suppress ALL
unsuppress Quartz Muscovite Kaolinite
```

## 7.98 surface\_capacitance

```
surface_capacitance = <value | ?>
surface_capacitance C1 = <value | ?> C2 = <value | ?>
surface_capacitance on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant capacitance  ▼ F/m<sup>2</sup>  
or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_capacitance”) to set, in units of F/m<sup>2</sup>, the capacitance or capacitances of a sorbing surface.

For a two-layer complexation model, when you set a capacitance with the command, or if a value for capacitance is set in the header section of the surface reaction dataset, **Phase2** will model surface complexation for the surface in question using the constant capacitance model, rather than the full two-layer model.

A triple-layer or CD-MUSIC complexation model, on the other hand, requires two capacitances,  $C_1$  and  $C_2$ . Default capacitances are set in the header of the surface dataset, but you may use the “C1” and “C2” arguments to override the default settings from the command line.

If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_capacitance on HFO = 2
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface.

## 7.99 surface\_data

```
surface_data <sorption dataset>
surface_data remove <sorption dataset | surface type>
surface_data OFF
```

**File** → **Open** → **Sorbing Surfaces...** →

**File** → **Open** → **Sorbing Surfaces...** → *select surface(s)* →   
 or **Config** → **Sorbing Surfaces...**

Use the “surface\_data” command (abbrev.: “surf\_data”) to specify an input dataset of surface sorption reactions to be considered in the calculation. The dataset name should be enclosed in quotes if it contains any unusual characters. Use the “remove” argument to eliminate a surface dataset, specified by name or surface type (e.g., “HFO”), from consideration. The argument “OFF” disables consideration of all surface complexes.

You can specify more than one sorbing surface in a model by repeating the “surface\_data” command for different datasets (a dataset of surface reactions for sorption onto hydrous ferric oxide, several triple-layer and CD-MUSIC datasets, as well as example datasets for the ion exchange,  $K_d$ , Freundlich, and Langmuir models are distributed with the software). To remove a dataset of surface reactions from consideration, you use commands such as

```
surface_data remove FeOH.sdat
surface_data remove HFO
surface_data OFF
```

The latter command removes all of the surface datasets that have been loaded.

## 7.100 surface\_potential

```
surface_potential = <value | ?>
surface_potential on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant potential  ▼ mV  
 or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_potential”) to set, in units of mV, the electrical potential for a two-layer sorbing surface. When you set this value (or if a value is set in the header section of the surface reaction dataset), **Phase2** will model surface complexation for the surface in question using the constant potential method, rather than invoking the full two-layer model.



If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_potential on HFO = 0
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface.

## 7.101 swap

```
swap <new basis> <for> <basis species>
```

**Basis** → ↻

Use the “swap” command to change the set of basis entries. All reactions are written internally in terms of a set of basis species that you can alter to constrain the composition of the initial system. An aqueous species, mineral, gas, or activity ratio can be swapped into the basis in place of one of the original basis species listed in the database. Examples:

```
swap CO3--      for HCO3-
swap Quartz     for SiO2(aq)
swap CO2(g)     for H+
swap O2(g)      for O2(aq)
swap Ca++/Na+^2 for Ca++
```

Each end member of a continuous solid solution can similarly be swapped into the basis. Use the syntax <solid solution>::<end member tag>. For example, the sodium end member Albite (tag “Ab”) of the plagioclase feldspar solid solution (name “Plag”) can be swapped into the basis simultaneously with the calcium end member Anorthite (tag “An”) as follows:

```
swap Plag::Ab for Na+
swap Plag::An for Al+++
```

The new species must contain in its composition the original basis species being swapped out (you can’t swap lead for gold). For example, CO<sub>2</sub>(g) is composed of HCO<sub>3</sub><sup>-</sup>, H<sup>+</sup>, and water in “thermo.tdat”. The reactions in the thermo dataset (once reduced to the set of basis and redox species and modified to reflect enabled redox couples) show the basis entries for which a species may be swapped. For a list of original basis species, type “show basis”. To reverse a swap, type “unswap <species>”.

## 7.102 system

```
system <system command>
```

Use the “system” command to execute a command on the computer’s operating system (a “DOS command”) without leaving **Phase2**. Example:

```
system "copy Phase2_output.txt Phase2_output2.txt"
```

Enclose multiword system commands in quotes. An alternative way to fork a system command is to begin the **Phase2** command with a “\$”:

```
$copy Phase2_output.txt Phase2_output2.txt
```

In this case, no quotation marks are needed.

When a “system” command is executed, a “Command Prompt” window will appear briefly on your screen. Due to limitations of the Windows operating system, you cannot fork a command that requires user input, and you will not be able to see any output (including error messages) that might be generated by the command.

## 7.103 TDS

```
TDS = <value | ?>
```

```
Basis →  advanced → TDS  ▼ mg/kg
```

Use the “TDS” command to set in mg/kg the total dissolved solids for the initial fluid, if you don’t want the program to calculate this value automatically. The program uses the TDS when needed to convert input constraints into molal units.

To restore automatic calculation of the TDS, type the command with no argument or with an argument of “?”. To see the variable’s current setting, type “show variables”.

## 7.104 temperature

```

temperature = <value> <unit>
temperature <x | y> initial = <value> <unit> final = <value> <unit>
temperature <x | y> initial = <value> <unit> reactants = <value> <unit>
temperature constant = <on | off>
temperature reset

```

**Basis** → temperature  C ▾

**X Axis** or **Y Axis** →

initial temperature  C ▾ → normal

slide temperature to  C ▾ sliding temperature

reactant temperature  C ▾ reactant mixing

-----  
hold constant

Use the “temperature” command (also: “T”) to set the temperature of the system or reactants. The **Units Recognized** chapter in the **GWB Reference Manual** lists possible units, which default to “C”. Examples:

```

temperature 25 C
T x initial = 25 C, final = 300 C
T x initial = 398 K, reactants = 498 K

```

Temperature values can range over the span of the thermo dataset, from 0°C to 300°C for “thermo.tdat”; 25°C is the default.

When setting a polythermal path, you specify whether temperature is to vary over the “x” or “y” axis, the scanning and staging axes, respectively. If you enter neither an “x” nor a “y” argument, the command defaults to the axis set by the “scope” command.

The “constant” keyword lets you hold temperature constant over the course of the calculation at the initial value, regardless of other settings; “on” enables the feature, as does omitting an argument; “off” disables it. The “reset” keyword restores temperature settings to their default states.

## 7.105 theta

```

theta = <value | ?>

```

**Config** → **Stepping...**

Use the “theta” command to set the time weighting variable used in evaluating kinetic rate laws. The value may vary from zero (full weighting at the old time level) to one

(full weighting at the new time level). By default, this variable is set to 0.6. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 7.106 threads

**threads** = <value | ?>

**Config** → **Stepping...**

Use the “threads” command to control the number of threads the program spawns when running a simulation. Using more than one thread on multicore or multiprocessor computers, the program runs in parallel, dividing the computing work among the computing cores available. By default (use the “?” argument), the program spawns one thread for each computing core. Typing the command without an argument returns the current setting and the number of computing cores available.

## 7.107 timax

**timax** = <value | ?>

**Config** → **Iteration...**

The “timax” command sets in molal units the maximum value of ionic strength used in calculating species’ activity coefficients when Debye-Hückel methods are employed. The limiting value is also used when water activity is calculated according to Wolery (1992b). By default, this variable is set to 3 molal. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 7.108 time

**time** <start = <value> <unit>> <end = <value> <unit>>  
**time** = off

**Basis** →  time    start  ▼ day ▼    end  ▼ day ▼

Use the “time” command (also: “t”) to set the time span of a kinetic reaction path. The starting time, by default, is zero, and the default end time is 1 day. Unit choices are listed in the **Units Recognized GWB Reference Manual** in the **GWB Reference Manual**; “day” is the default. Argument “off” switches the program out of kinetic mode. Examples:

```
time end = 100 years
time start 10 days, end 20 days
```

## 7.109 title

```
title <character string>
```

**Config** → **Output...**

Use the “title” command to set a title to be passed to the graphics program. Example:

```
title "Yucca Mountain Groundwater"
```

Be sure to put multiword titles in quotes.

## 7.110 unalter

```
unalter <species | mineral | solid solution | gas | surface species | ALL>
```

**Config** → **Alter Log Ks...** → *select entry(ies)* →

Use the “unalter” command to reverse the effect of having changed the log  $K$ ’s for a species, mineral, gas, or surface reaction, the selectivity coefficient for an exchange species, the  $K_d$  for a sorbed species, the  $K_f$  and  $n_f$  for a Freundlich species, or the properties of a solid solution in the current dataset. Example:

```
unalter Quartz
```

In this case, the log  $K$  values for quartz revert to those in the current thermodynamic dataset. The argument “ALL” resets the log  $K$ ’s, selectivity coefficients,  $K_d$ ’s, or  $K_f$ ’s and  $n_f$ ’s for all species, minerals, gases, and surface species, as well as the properties of all solid solutions in the dataset.

## 7.111 unsegregate

```
unsegregate <mineral(s) | ALL>
```

**Config** → **Isotopes...** → *select entry(ies)* →

Use the “unsegregate” command to remove minerals from the list of minerals to be segregated isotopically.

## 7.112 unsuppress

**unsuppress** <species, minerals, solid solutions, gases, surface species | **ALL**>

**Config** → **Suppress...**

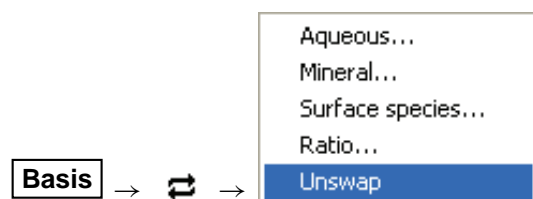
Use the “unsuppress” command (also: “include”) to include in the calculation aqueous species, surface species, minerals, solid solutions, or gases that have previously been suppressed. Examples:

```
unsuppress Quartz Albite "Albite low"
unsuppress ALL
```

The argument “ALL” clears any species, minerals, solid solutions, or gases that have been suppressed.

## 7.113 unswap

**unswap** <species | **ALL**>



Use the “unswap” command to reverse a basis swap. Example:

```
unswap Quartz (or unswap SiO2(aq))
```

to reverse the effect of the previous command

```
swap Quartz for SiO2(aq)
```

At this point, SiO<sub>2</sub>(aq) is back in the basis. The “ALL” argument reverses all basis swaps.

## 7.114 usgovt

**usgovt**

**Help** → **About Phase2...**

Display the restricted rights legend that governs use, duplication, or disclosure by the U.S. Government.

## 7.115 volume

**volume** = <value | ?> <unit>

**Medium** → bulk volume     cm3

Use the “volume” (also “bulk\_volume”) command to cause the program to scale the initial system to a specific size. By default, the program does not scale the system. In this case, the bulk volume  $V_b$

$$V_b = V_f + V_{mn} + V_{in} + V_{stag} \quad (7.1)$$

is computed as the sum of the fluid volume  $V_f$ , mineral volume  $V_{mn}$ , inert volume  $V_{in}$ , and the volume  $V_{stag}$  of the stagnant zone, if one has been specified using the “dual\_porosity” command.

If  $X_{stag}$  is the stagnant fraction of the system, which is zero if a dual porosity model has not been invoked, the above equation can be rewritten

$$(1 - X_{stag}) V_b = V_f + V_{mn} + V_{in} \quad (7.2)$$

When you use the “volume” command to set the bulk volume  $V_b$ , the program scales the fluid volume  $V_f$  and optionally the mineral and inert volumes,  $V_{mn}$  and  $V_{in}$ , so the system fills the specified space.

The command

volume = 2 m3

for example, causes the program to scale the fluid volume  $V_f$  and optionally the mineral and inert volumes,  $V_{mn}$  and  $V_{in}$ , so the system occupies 2 m<sup>3</sup>. Default behavior, in which no scaling is performed, can be restored at any time with the command

volume = ?

The unit you use to constrain the amount of each mineral in the system controls whether or not the program adjusts that mineral’s volume. If you constrain Quartz to “10 cm<sup>3</sup>”, for example, Quartz will occupy 10 cm<sup>3</sup> after scaling. Setting Quartz in relative units such “volume%” or “mmol/m<sup>3</sup>”, in contrast, causes the program to fix the mineral’s volume relative to  $V_b$ .

Similarly if you set inert volume in an absolute unit, such as

inert = 50 cm3

## GWB Command Reference

---

the program will honor that constraint and hold  $V_{in}$  steady. The inert portion of the system in this example will occupy 50 cm<sup>3</sup> after scaling. The command

```
inert = 10 volume%
```

however, causes the program to scale the inert volume to 10% of the system's bulk volume.



# X1t Commands

## 8.1 <unit>

<field variable> <free> <unit> <as element symbol> <basis entry>

**Initial** or **Fluids** →  ▼ free kg ▼

To constrain chemical composition, enter a command containing only the above entries. Entries may appear in any order. The qualifier “free” specifies that the constraint applies to the free rather than to the bulk basis entry. Settings constrain the composition of the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). **X1t** recognizes the following units:

*By mass or volume:*

mol	mmol	umol	nmol	
kg	g	mg	ug	ng
eq	meq	ueq	neq	
cm3	m3	km3	l	

*By concentration:*

mol/kg	mmol/kg	umol/kg	nmol/kg	
molal	mmolal	umolal	nmolal	
mol/l	mmol/l	umol/l	nmol/l	
g/kg	mg/kg	ug/kg	ng/kg	
wt%	"wt fraction"			
g/l	mg/l	ug/l	ng/l	
eq/kg	meq/kg	ueq/kg	neq/kg	
eq/l	meq/l	ueq/l	neq/l	

*By carbonate alkalinity (applied to bicarbonate component):*

eq_acid	meq_acid	ueq_acid	neq_acid	
eq_acid/kg	meq_acid/kg	ueq_acid/kg	neq_acid/kg	
eq_acid/l	meq_acid/l	ueq_acid/l	neq_acid/l	
g/kg_as_CaCO3	mg/kg_as_CaCO3	ug/kg_as_CaCO3	ng/kg_as_CaCO3	
wt%_as_CaCO3				
g/l_as_CaCO3	mg/l_as_CaCO3	ug/l_as_CaCO3	ng/l_as_CaCO3	
mol/kg_as_Ca...	mmol/kg_as_Ca...	umol/kg_as_Ca...	nmol/kg_as_CaCO3	
mol/l_as_CaCO3	mmol/l_as_CaCO3	umol/l_as_CaCO3	nmol/l_as_CaCO3	

## GWB Command Reference

### *Per volume of the system:*

mol/cm <sup>3</sup>	mmol/cm <sup>3</sup>	umol/cm <sup>3</sup>	nmol/cm <sup>3</sup>
kg/cm <sup>3</sup>	g/cm <sup>3</sup>	mg/cm <sup>3</sup>	ug/cm <sup>3</sup>
ng/cm <sup>3</sup>			
mol/m <sup>3</sup>	mmol/m <sup>3</sup>	umol/m <sup>3</sup>	nmol/m <sup>3</sup>
kg/m <sup>3</sup>	g/m <sup>3</sup>	mg/m <sup>3</sup>	ug/m <sup>3</sup>
ng/m <sup>3</sup>			
volume%	"vol. fract."		

### *By activity:*

activity	fugacity	ratio
pH	V	pe

### *By partial pressure:*

Pa	MPa	atm	bar
psi			

Activity and fugacity may be abbreviated to "a" or "f". Keyword "total" reverses a setting of "free".

Use the "as" keyword to constrain mass in terms of elemental equivalents. For example, the command

```
CH3COO- = 10 umol/kg as C
```

specifies 5 umol/kg of acetate ion, since each acetate contains two carbons, whereas

```
20 mg/kg SO4-- as S
```

would specify 59.9 mg/kg of sulfate, since the ion's mole weight is about 3 times that of sulfur itself.

Examples:

```
55 mmol/kg HCO3-  
Na+ = 1 molal  
1 ug/kg U++++  
10 volume% Dolomite  
pH = 8  
Eh = .550 V  
log f O2(g) = -60  
HCO3- = 30 mg/kg as C
```

You may in a similar fashion constrain the concentration of a kinetic aqueous or surface complex in the initial system, a boundary fluid, or globally. For example, the commands

```
kinetic AIF++
AIF++ = 1 umol/kg
scope ambient_water
  AIF++ = 10 umol/kg
```

set the concentration of the kinetic complex AIF<sup>++</sup> to 1  $\mu\text{mol kg}^{-1}$  in all fluids, except fluid “ambient\_water”, where its concentration is ten times greater.

## 8.2 <isotope>

```
<isotope | symbol> <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the name of any isotope system loaded in the isotope dataset (also, the isotope’s symbol) to set the isotopic composition of the initial fluid, boundary fluids, reactant species (aqueous species, minerals, end members, gases, or oxides), or segregated minerals. The composition may be set on any scale (e.g., SMOW, PDB, ...), but you must be consistent throughout the calculation.

For example, if the <sup>17</sup>O isotope system has been added to the isotope dataset, you could enter:

```
scope initial
  oxygen-17 fluid = -10
scope inlet-1
  oxygen-17 fluid = +2
scope
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
oxygen-17 remove
oxygen-17 off
```

clear all settings for <sup>17</sup>O isotopes from the calculation.

## 8.3 activity

**activity** <species> = <field variable>

or  →  ▼ activity ▼

Use the “activity” command (abbrev.: “a”) to constrain the activity of an aqueous species. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). Examples:

```
activity Na+ = 0.3
log a H+ = -5
```

See also the “pH”, “Eh”, “pe”, “ratio”, “fugacity”, “fix”, and “slide” commands.

## 8.4 add

**add** <basis species>

or  →

Use the “add” command to include a basis species in the calculation. Example:

```
add HCO3-
```

See also the “swap”, “activity”, “fugacity”, “pH”, “pe”, and “Eh” commands.

## 8.5 alkalinity

**alkalinity** = <field variable> <unit>

or  → HCO3-  ▼ mg/kg\_as\_CaCO3 ▼

Use the “alkalinity” command to constrain the total concentration of  $\text{HCO}_3^-$  to reflect the solution’s carbonate alkalinity. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “mg/kg\_as\_CaCO3” is the default. To use this option, the solution pH must be set explicitly.

## 8.6 alter

```

alter <species | mineral | gas> <log Ks>
alter <species | mineral | gas> <poly coefs> TminK = <value> TmaxK = <value>
alter <surface species> <poly coefs> TminK = <value> TmaxK = <value>
alter <surface species> logK = <value> dlogK/dT = <value>
alter <exchange species> beta = <value>
alter <sorbed species> Kd = <value>
alter <sorbed species> Kf = <value> nf = <value>
alter <solid solution> <type> <discrete | continuous> \
  <from <value> to <value> step <value>> <<variable> = <value>>

Config → Alter Log Ks...
Config → Solid Solutions...

```

Use the “alter” command to change the temperature expansion for a species’, mineral’s, or gas’ log  $K$ , to adjust the stability of a surface species, or to change the properties of a solid solution.

Temperature expansions for log  $K$ s are given in the thermo dataset either as  $T$ -tables, or by up to six coefficients of a polynomial. When the current thermo dataset uses  $T$ -tables, you list replacement values at each of the principal temperatures specified in the dataset, most commonly 0°C, 25°C, 60°C, 100°C, 150°C, 200°C, 250°C, and 300°C. Example:

```
alter Anhydrite -4.3009 -4.4199 -4.7126 -5.1758 -6.2299 500 500 500
```

Values of “500” represent a lack of data at the corresponding temperature.

For a thermo dataset constructed from polynomial expansions, on the other hand, list instead up to six polynomial coefficients

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4
```

You can optionally append a temperature range of validity, in Kelvins, for the polynomial:

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4 TminK= 293 TmaxK= 383
```

Absent a range, the polynomial is taken to span the principal temperatures. You may adjust a reaction’s temperature range without specifying polynomial coefficients, but setting only coefficients resets the temperature range.

For a surface complex, use the “alter” command to set log  $K$ , its temperature derivative, or both

```
alter >(w)FeOCa+ logK = 6.0 dlogK/dT = 0.02
```

whereas for an ion exchange reaction, you set the selectivity coefficient  $\beta$

```
alter >X2:Ca beta = .033
```

directly, rather than as a logarithm. For a sorbing species, set  $K_d$  or, for a Freundlich species,  $K_f$ ,  $n_f$ , or both, as follows

```
alter >Pb++ Kd = .03  
alter >Sr++ Kf = .015 nf = .8
```

For a solid solution, use the “alter” command to set the activity coefficient model, discrete or continuous behavior, the composition range in terms of the mole fraction of the most recently specified end member, and parameters for the chosen activity model, as described for the “solid\_solution” command. A solid solution “my\_ss” might be set as a subregular Guggenheim solution with the command

```
alter my_ss guggenheim a0 = 1 a1 = 2
```

You may prefer to use the “alter” command to adjust an entry from the thermo database, and the “solid\_solution” command to reconfigure a user-defined solution, but the two commands work interchangeably for this purpose.

Type “show alter” to list altered species or solutions and their settings; the “unalter” command reverses the process.

## 8.7 audit

```
audit <off>
```

**Config** → **Stepping...**

The program performs a global mass balance audit over the course of each simulation. By default, the program is silent unless it detects a mass imbalance, in which case it writes a warning message to the **Results** pane, as well as the bottom of the print-format output, if the print output stream is open.

You can use the “audit” option to cause the program to report at the end of the simulation the complete results of the mass balance audit for each original basis entry.

To turn audit reporting on and off, use:

```
audit  
audit off
```

## 8.8 b-dot

**b-dot**

**File** → **Open** → **Thermo Data...** → thermo.tdat

The “b-dot” command (formerly: “debye-huckel”) causes the program to read the “thermo.tdat” thermodynamic dataset, which invokes the “B-dot” form of the extended Debye-Hückel equation to calculate activity coefficients for aqueous species.

## 8.9 balance

**balance** <on> <basis entry>

**Initial** or **Fluids** →  ▼ unit ▼ → Balance species

Use the “balance” command to specify the basis entry to be used for electrical charge balancing. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). The basis entry must be a charged aqueous species. By default, **X1t** balances on  $\text{Cl}^-$ . Example:

```
balance on Na+
```

## 8.10 carbon-13

**carbon-13** <fluid | reactant | segregated mineral> = <value>

**Config** → **Isotopes...**

Use the “carbon-13” command (also, “13-C”) to set the  $^{13}\text{C}$  isotopic composition of the initial fluid, boundary fluids, reactant species (aqueous species, minerals, end members, gases, or oxides), or segregated minerals. The composition may be set on any scale (e.g., PDB), but you must be consistent throughout the calculation. Example:

```
scope initial
  carbon-13 fluid = -10
scope inlet-1
  carbon-13 fluid = +4
scope
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
carbon-13 remove  
carbon-13 off
```

clear all settings for  $^{13}\text{C}$  isotopes in the current scope.

See also the “<isotope>” section above, and the “hydrogen-2”, “oxygen-18”, and “sulfur-34” commands.

### 8.11 chdir

```
chdir <directory> <ALL>
```

**File** → **Working Directory...**

Use the “chdir” command (abbrev.: “work\_dir”, “cd”) to change the working directory. The program reads input scripts relative to the current working directory and writes output into it. Typing the command “chdir” without an argument causes the program to display the name of the working directory. The “ALL” argument changes the working directories of all of the GWB apps. The command

```
chdir ~
```

changes to the user’s home directory, if one is defined by the operating system.

### 8.12 clear

```
clear
```

**View** → **Clear Screen**

Use the “clear” command (abbrev.: “c”) or right-click in the text window and choose **Clear Screen** to clear it.

### 8.13 clipboard

```
clipboard
```

**Edit** → **Copy**

The “clipboard” command causes the contents of the **Results** pane to be copied to the clipboard, from where it can be pasted into various types of documents.



## 8.14 conductivity

**conductivity** <conductivity dataset>

**File** → **Open** → **Conductivity Data...**

Use the “conductivity” command to change the input file of coefficients used to calculate electrical conductivity. Example:

```
conductivity "..\my_conductivity.dat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. Beginning with GWB11, the applications compute electrical conductivity using either of two different approaches, the USGS and APHA methods; the USGS method is the default. The required coefficients are defined in the files “conductivity-USGS.dat” and “conductivity-APHA.dat”, respectively, which are installed in the same directory as the thermo datasets (commonly “\Program Files\GWB\Gtdata”).

## 8.15 couple

**couple** <redox species | element(s) | **ALL**>

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “couple” command to enable any redox coupling reactions that have been disabled with the “decouple” command. You specify one or more redox species or elements. For example, the command

```
couple Carbon
```

couple all redox reactions involving the element carbon. Argument “ALL” enables all of the coupling reactions in the thermo dataset.

## 8.16 Courant

**Courant** = <value | ?>

**Config** → **Stepping...**

Use the “Courant” command to constrain the time step according to the Courant condition. You enter a value for the Courant number, which is the ratio of the distance fluid travels over a time step to the length of the nodal blocks. If you set a Courant number of one, then **X1t** will select a time step over which the fluid will exactly traverse the nodal blocks. For a value of 0.5, the fluid will move halfway across the nodal

blocks, and so on. Values greater than one for the Courant number typically give unstable solutions and are therefore not recommended. By default, **X1t** assumes a Courant number of 1.0. The “?” argument resets the default value.

## 8.17 cpr

**cpr** = <field variable | ?> <unit> <steady | transient>

**Medium** → cpr (minerals)   ▼

Use the “cpr” command to set the heat capacity of the rock (mineral) framework. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/g°C” is the default. This value is used during polythermal simulations in calculating the effects of advective heat transport. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function.

By default, this variable is set to 0.2 cal/g°C. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 8.18 cpu\_max

**cpu\_max** = <value | ?>

**Config** → **Stepping...**

Use the “cpu\_max” command to limit the amount of computing time a simulation may take. You set the maximum computing time in seconds, or use a “?” to restore the default state, which is no prescribed limit. To see the current setting, type “show variables”.

## 8.19 cpw

**cpw** = <field variable | ?> <unit> <steady | transient>

**Medium** → cpw (fluid)   ▼

Use the “cpw” command to set in cal/g°C the heat capacity of the fluid. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/g°C” is the default. This value is used during polythermal simulations in calculating the effects of advective heat transport. The “transient” keyword causes the

model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function.

By default, this variable is set to 1.0 cal/g°C. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 8.20 data

**data** <thermo dataset> <verify>

**File** → **Open** → **Thermo Data...**

Use the “data” command to change the input file of thermodynamic data. Example:

```
data "..\my_thermo.tdat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. The “verify” option causes the program to read the named dataset only if it has not already been read.

## 8.21 decouple

**decouple** <redox species | element(s) | **ALL**>

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “decouple” command to disable the coupling reactions for one or more redox species, in order to calculate a model assuming redox disequilibrium. The redox species then become available for use as basis species and may be constrained independently of the original basis entries. You can disable as many coupling reactions as you want.

You specify either one or more redox species or elements. For example, the command

```
decouple Carbon
```

decouples all redox reactions involving the element carbon. Argument “ALL” disables all of the coupling reactions in the thermo dataset. Use the “couple” command to enable coupling reactions, once they have been disabled.

## 8.22 delQ

**delQ** = <value | ?>

**Config** → **Stepping...**

Use the “delQ” command to control the lengths of time steps taken in a simulation accounting for reaction kinetics. The program limits how much the ion activity product  $Q$  can change over a step, for each kinetic reaction considered. The setting for “delQ” is the projected change  $\Delta Q/Q$  allowed in the relative value of the activity product. You can set a larger value to permit longer time steps, or a smaller value to improve stability. The default setting is 0.1. Type the command with no argument or with an argument of “?” to restore the default. To see the current setting, type “show variables”.

## 8.23 deltar

**deltar** = <field variable> <unit>

**Domain** →  delta r  +  ▼ cm ▼

The “deltar” (also: “delta\_r”) command is a synonym for the “deltax” command, described below. It is provided for clarity when working with radial domains.

## 8.24 deltax

**deltax** = <field variable> <unit>

**Domain** →  delta x  +  ▼ cm ▼

The “deltax” (also: “delta\_x”) command sets the length  $\Delta x$  of nodal blocks in the finite difference grid. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cm” is the default.

When you set a value for  $\Delta x$ , the program ignores any setting of the “length” command and figures the domain length from the values of  $\Delta x$  and  $N_x$ , the number of nodal blocks specified along  $x$ . For example, the commands

Nx = 100  
deltax = 1 m

are equivalent to

Nx = 100  
length = 100 m

You can set  $\Delta x$  as a field variable, in order to create grids of varying node density. For example, the commands

```
Nx = 10
deltax = { 1 1 .5 .5 .25 .25 .5 .5 1 1 } m
```

set a grid in which nodal blocks near the center of the domain are smaller than those near the left and right sides. The domain in this case will be 7 m long. See also the “deltar”, “length”, and “Nx” commands.

## 8.25 delxi

```
delxi = <value | ?>
```

**Config** → **Stepping...**

Use the “delxi” command to set the maximum length (in terms of reaction progress, which varies from zero to one over the course of the simulation) of the time step. By default, this variable is set to 0.01. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 8.26 density

```
density = <value | ?>
density = <TDS | chlorinity>
density = <batzle-wang | phillips>
```

**Initial** →  advanced → density  ▼ g/cm<sup>3</sup>

**Fluids** →  advanced → density  ▼ g/cm<sup>3</sup>

**Config** → **Options...**

You can use the “density” command to set in g/cm<sup>3</sup> the fluid density the program uses to convert compositional constraints to molality, the concentration unit it carries internally. If you set the initial Na<sup>+</sup> composition in mg/l, for example, the program needs to know the density of the initial fluid to determine Na<sup>+</sup> molality. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command).

The program by default converts units using a density value it calculates automatically, as discussed below. This value is sufficient for most purposes, and hence it is generally not necessary to set fluid density explicitly. You might, however, want to set the density if you are working at high temperature, but your analysis is expressed per liter of solution at room temperature.

You can also use the “density” command to tell the program how to calculate the default density it uses to convert units, and the fluid density it reports in the simulation results. The program by default uses the Batzle-Wang equation to figure density, but you can use the command “density = phillips” to select the method of Phillips et al., instead.

As well, the program normally figures density as that of an NaCl solution with the same TDS as the fluid in question, at the temperature of interest. With the command “density = chlorinity” you can tell the program to instead use the density of an NaCl solution of equivalent chlorinity.

To restore automatic calculation, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 8.27 diffusion\_coef

**diffusion\_coef** = <field variable | ?> <unit> <steady | transient>

**Medium** → diffusion coefficient  +  ▼ cm2/s ▼

Use the “diffusion\_coef” command (abbrev.: “diff\_coef”) to set a value for the diffusion coefficient in one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function. By default, **X1t** assumes a diffusion coefficient of  $10^{-6}$  cm<sup>2</sup>/s. The “?” argument resets the default value.

## 8.28 discharge

**discharge** = <value | ?> <unit>  
**discharge** <interval> = <value | ?> <unit>

**discharge fluid** = <boundary fluid>  
**discharge** <interval> **fluid** = <boundary fluid>  
**discharge** <left | right> **fluid** = <boundary fluid>  
**discharge** <left | right> <interval> **fluid** = <boundary fluid>

**Flow** →  interval  discharge  ▼ cm/s ▼

Use the “discharge” command to set a value for the specific discharge (fluid volume crossing a unit area per unit time) of the fluid into the domain, and to set the fluid to appear at the left or right boundary, or both. A positive value represents flow into the left side of the domain, and negative gives flow rate into the right side.

If the “discharge” command contains the name of a simulation interval, subsequent settings on the command apply to that interval, rather than the simulation as a whole.

This behavior continues to the end of the command, or until the name of another interval is encountered.

By default, discharge is not set and the program calculates flow rate from the head drop, the permeability of the medium, and the viscosity of the pore fluid. Enter an argument of “?” to set the default condition. If, furthermore, you specify a driving force for flow using the “head\_drop” command (see discussion later in this section), the program will unset any value specified for specific discharge and resume calculating discharge from the driving force, permeability, and fluid viscosity.

The value may set be in any of the units of velocity or specific discharge listed in the **Units Recognized** chapter in the **GWB Reference Manual**. The default is cm/s, which is equivalent to  $\text{cm}^3/\text{cm}^2/\text{s}$ . In addition, you may set as the unit “pore\_volumes” (or “pv”) to specify discharge in terms of the number of times the original pore volume is to be replaced over the course of the reaction model.

Examples:

```
discharge = 15 m/day
discharge = 100 pore_volumes
discharge start = 20 m/day interval-2 = -20 m/day
```

You can also specify the boundary fluid to appear on either the left and right boundaries, or both, for the entire simulation or an individual interval. To do so, you use the “fluid” keyword. In the examples

```
discharge = .002 m/s fluid = ambient_water
discharge phase2 left fluid = pit_water
```

fluid “ambient\_water” flows across the left boundary at a discharge of  $.002 \text{ m s}^{-1}$  over the course of the simulation, except during interval “phase2”, during which time fluid “pit\_water” flows into the left of the domain, but “ambient\_water” is found to the right.

## 8.29 dispersivity

```
dispersivity = <field variable | ?> <unit> <steady | transient>
```

**Medium** → longitudinal dispersivity   cm

Use the “dispersivity” command to set a value for the hydrodynamic dispersivity. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cm” is the default. By default, **X1t** assumes a value equal to 1% of the length of the domain (see the “length” command). The “?” argument resets the default state.

## 8.30 dual\_porosity

```

dual_porosity = <on | off> <spheres | blocks | fractures> \
  <geometry = <spheres | blocks | fractures | ?>> \
  <volfrac = <field variable | ?>> \
  <Nsubnode = <value | ?>> \
  <<radius | half-width> = <field variable | ?> <unit>> \
  <diff_length = <field variable | ?> <unit>> \
  <porosity = <field variable | ?>> \
  <retardation = <field variable | ?>> <steady | transient> \
  <diff_coef = <field variable | ?> <unit>> <steady | transient> \
  <thermal_con = <field variable | ?> <unit>> <steady | transient> \
  <theta = <value | ?>> <reset | ?>

```

**Config** → **Dual Porosity...**

Use the “dual\_porosity” command (abbrev: “dual”) to configure stagnant zones in the simulation, using the dual porosity feature. Enable and disable the feature with the “on” and “off” keywords. Disabling the feature does not affect other settings, so re-enabling the feature returns the model to its most recent configuration.

With the “geometry” keyword, you configure the stagnant zone into spheres, blocks, or a fractured domain, the latter being slabs separated by fractures arrayed along the  $x$  direction. Alternatively, you can set the three configurations directly with keywords “spheres”, “blocks”, and “fractures”. Keyword “volfrac” sets the fraction of the nodal block’s bulk volume occupied by the stagnant zone.

The “Nsubnode” (or “nx”) keyword sets the number of nodes into which the stagnant zone within each node will be divided when solving for solute and temperature distributions. The “radius” (or “half-width”, for blocks and fractures) keyword sets the zone’s characteristic dimension, in units of distance (see the **Units Recognized** chapter in the **GWB Reference Manual**; default is cm), and keyword “diff\_length” sets the distance (same units) from the contact with the free-flowing zone over which the model will account for solute diffusion and heat conduction. Use keywords “porosity”, “retardation”, “diff\_coef”, and “thermal\_con” to set values for the porosity, retardation factor, diffusion coefficient, and thermal conductivity of the stagnant zone. The default unit for the diffusion coefficient is  $\text{cm}^2/\text{s}$ , and thermal conductivity defaults to  $\text{cal}/\text{cm}/\text{s}/^\circ\text{C}$  units; see the **Units Recognized** chapter for a list of options.

By default, the stagnant zone is configured in spheres divided into 5 subnodes. You must specify a value for the volume fraction of the stagnant zone, as well as one for the radius (or half-width); the diffusion length defaults to the latter value. The program uses whatever values are set for the free-flowing zone in the node in question as default values for the porosity, diffusion coefficient, and thermal conductivity of the stagnant zone; the retardation factor defaults to a value of one.

The “theta” keyword sets time weighting ( $0 \leq \theta \leq 1$ ) for the numerical solution of diffusive transport within the stagnant zone. A weight  $\theta = 0$  assigns the explicit method, and larger values invoke an implicit solution. By default, the program chooses  $\theta$



automatically, using the explicit method unless it would force too many more time steps than would otherwise be necessary. In that case, the program uses the implicit method ( $\theta = 0.6$ ), which requires more computing effort per time step, but can take long steps without becoming numerically unstable.

You can append the “transient” keyword when setting several of the parameters: the diffusion coefficient, thermal conductivity, and retardation factor. If the variable is defined by an equation, script, or external function, it will then be re-evaluated continuously over the course of the run.

As an example, the command

```
dual_porosity geometry = spheres, radius = 50 cm, volfrac = 75%
```

configures the stagnant zone into spheres of half-meter radius that occupy three-quarters of the domain. The command

```
dual_porosity reset
```

enables the feature after restoring default settings for each keyword.

### 8.31 dx\_init

```
dx_init = <value | ?>
```

**Config** → **Stepping...**

Use the “dx\_init” command to set the length of the initial time step. You set this value in terms of reaction progress, which varies from zero to one over the course of the simulation.

By default, the variable is ignored. To restore the default state, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

### 8.32 dxplot

```
dxplot = <value | ?> <linear | log> <exact | approx>
```

**Config** → **Output...**

Use the “dxplot” command to set the interval in reaction progress (which varies from zero to one over the course of the simulation) between entries in the “X1t\_plot.xtp” dataset, and to specify the entries be spaced on a linear or logarithmic scale. A value of zero causes the program to write the results after each time step, which can produce unwieldy amounts of output.

The command

```
dxplot = .1 linear
```

for example, causes the program to write entries for when reaction progress is .1, .2, .3, ..., and 1.0. Alternatively, the commands

```
dx_init = .001  
dxplot = .5 log
```

produce entries at .001, .003, .01, .03, .1, .3, and 1.0 (see the “dx\_init” command).

The arguments “exact” and “approx” define whether the program will constrain the sizes of time steps to give entries that match the specified increment exactly (the default case), or will output entries whenever the time stepping meets or exceeds the increment.

By default, this variable is set to 0.02, “linear”, and “exact”. To restore the default settings, type the command with no argument or with an argument of “?”. To see the current settings, type “show variables”.

### 8.33 dxprint

```
dxprint = <value | ?> <linear | log> <exact | approx>
```

**Config** → **Output...**

Use the “dxprint” command to set the interval in reaction progress (which varies from zero to one over the course of the simulation) between entries in the “X1t\_output.txt” dataset, and to specify the entries be spaced on a linear or logarithmic scale. A value of zero causes the program to write the results after each time step, which can produce unwieldy amounts of output.

The command

```
dxprint = .1 linear
```

for example, causes the program to write entries for when reaction progress is .1, .2, .3, ..., and 1.0. Alternatively, the commands

```
dx_init = .001  
dxprint = .5 log
```

produce entries at .001, .003, .01, .03, .1, .3, and 1.0 (see the “dx\_init” command).

The arguments “exact” and “approx” define whether the program will constrain the sizes of time steps to give entries that match the specified increment exactly (the default case), or will output entries whenever the time stepping meets or exceeds the increment.

By default, this variable is set to 0.1, “linear”, and “exact”. To restore the default settings, type the command with no argument or with an argument of “?”. To see the current settings, type “show variables”.

### 8.34 Eh

**Eh** = <field variable> <unit>

**Initial** or **Fluids** →  ▼ V ▼

Use the “Eh” command to set Eh in the initial system. Example:

Eh = 200 mV

sets the system’s oxidation state to correspond to an Eh of 0.2 volt. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). See also the “activity”, “pH”, “pe”, “fugacity”, “fix”, and “slide” commands.

### 8.35 epsilon

**epsilon** = <value | ?>

**Config** → **Iteration...**

Use the “epsilon” command to set the convergence criterion (dimensionless) for iterating to a solution of the equations representing the distribution of chemical mass. By default, this variable is set to  $5 \times 10^{-11}$ . To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

### 8.36 exchange\_capacity

**exchange\_capacity** = <field variable | ?> <units>

**exchange\_capacity on** <type> = <field variable | ?> <units>

**Config** → **Sorbing Surfaces...** → exchange capacity  ▼ eq ▼  
or **File** → **Open** → **Sorbing Surfaces...**

Use the “exchange\_capacity” command (abbrev.: “ex\_capacity” or “exh\_capacity”) to set the exchange capacity (i.e., the CEC) of the system when modeling ion exchange reactions or sorption according to Langmuir isotherms. For ion exchange reactions, you set units of electrical equivalents (“eq”, “meq”, and so on) or equivalents per gram

of dry sediment (“eq/g”, “meq/g”, ...). For Langmuir reactions, you similarly set a value in mole units: “mol”, “mmol”, “mol/g”, “mmol/g”. If you set units per gram of sediment, the program multiplies the value entered by the mass of rock in the system (including equilibrium and kinetic minerals as well as inert volume) to get the system’s capacity.

If you read in a surface that sorbs by ion exchange or Langmuir isotherms, you must set a value for its exchange capacity. If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example

```
exchange_capacity on IonEx = .0008 eq/g
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the surface represented by the sample dataset “IonEx.sdat”, for example, is “IonEx”. You can use the “show” command to display the “type” of each active surface. See also the “surface\_data” and “inert” commands.

### 8.37 explain

```
explain <species | mineral(s) | solid solution(s) | gas(es) | surface species>
```

**Config** → **Show...** → *select species*

Use the “explain” command to get more information (such as the mole weight of a species, a mineral’s formula, mole volume, and density, or a solid solution’s end-member minerals) about species, minerals, solid solutions, and gases in the dataset. Example:

```
explain Analcime
```

### 8.38 explain\_step

```
explain_step <off>
```

**Config** → **Stepping...**

The “explain\_step” option causes the program to report on the Results pane the factor controlling the length of each time step, whenever the step size is limited by the need to maintain numerical stability.

## 8.39 extrapolate

**extrapolate** <on | off>

**Config** → **Options...**

Use the “extrapolate” option to cause the program to extrapolate log  $K$ s for reactions forming aqueous species, minerals, and gases to temperatures beyond the data’s known range of validity. The option should be used with considerable care.

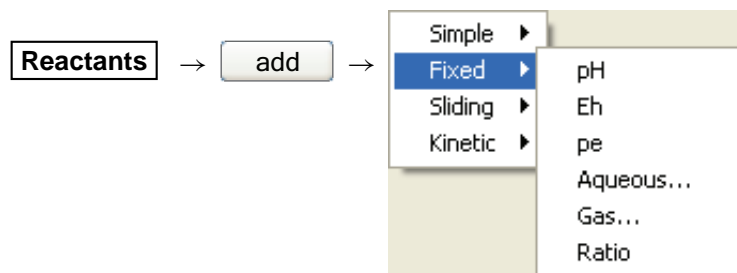
The temperature range of validity for a species’ log  $K$  is defined by its temperature expansion, taken from the thermo dataset. For  $T$ -table expansions, it is the span of principal temperatures at which log  $K$ s are not 500, whereas for polynomial expansions, the range is listed explicitly.

In normal operation, the program loads only species with ranges of validity encompassing the calculation’s temperature range. When the “extrapolate” option is “on”, in contrast, the program projects to the temperature of interest the log  $K$ s of species that would not otherwise be loaded.

Temperature ranges of validity can also be set in the thermo dataset for the virial coefficients used to calculate “Pitzer” and SIT activity coefficients. The “extrapolate on” option works in a similar manner in such cases, allowing virial coefficients to be used at temperatures beyond the coefficients’ known ranges of validity.

## 8.40 fix

**fix** <unit> <species | gas>



Use the “fix” command to hold the activity of a species, fugacity of a gas, or an activity ratio constant over the course of a run. The <unit> can be “activity” or “fugacity” (“a” or “f” for short), “ratio”, “pH”, “pe”, or “Eh”, or it can be omitted. Examples:

```
fix pH
fix a H+
fix f O2(g)
fix ratio Ca++/Na+2
```

## 8.41 fugacity

**fugacity** <gas> = <field variable>

or  →  ▼ fugacity ▼

Use the “fugacity” command (abbrev.: “f”) to set gas fugacities (on an atm scale) in the initial system. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). Examples:

```
fugacity O2(g) = .2
f CO2(g) = 0.0003
log f S2(g) = -30
```

See also the “activity”, “ratio”, “pH”, “Eh”, “pe”, “fix”, and “slide” commands.

## 8.42 go

**go** <boundary | initial | single | ask>

**Run** → **Go**  
**Run** → **Go Initial**  
**Run** → **Go Single**

→

Use the “go” command to initiate calculation of a model based on the current configuration. Used with the “boundary” argument, the command causes the program to calculate only the composition of the boundary fluids. With the “initial” argument, the program calculates the initial state of the medium as well as the composition of the boundary fluids, but does not trace a reaction model. Typing the command with the argument “single” causes the simulation to run on a single processor, regardless of the setting for the “threads” command.

Upon completing a simulation, the program normally presents a dialog box asking if you would like to continue the simulation over an extended time span. This option appears if you have triggered the calculation by entering “go” from the keyboard, selecting “Go” from the menubar, or moving to the **Results** pane and clicking the **Run** button, but not if you are reading commands from a file or running a control script. Typing

```
go ask
```

causes the program to offer the option to continue the simulation in all cases.

## 8.43 grep

**grep** <string>

**File** → **View** → **.\X1t\_output.txt** → **Edit** → **Find**

Use the “grep” command to search for each occurrence of the given character string in the current output dataset. For example, use

grep Albite

to obtain a list of each line in “X1t\_output.txt” (or the current output dataset, as modified by the “suffix” command) containing the string “Albite”.

## 8.44 h-m-w

**h-m-w**

**File** → **Open** → **Thermo Data...** → **thermo\_hmw.tdat**

Use the “h-m-w” command (abbrev.: “hmw”) to set the program to calculate species’ activity coefficients by using the Harvie-Møller-Weare equations. Executing this command automatically sets the input dataset of thermodynamic data to “thermo\_hmw.tdat”. Note that dataset “thermo\_hmw.tdat” supports calculations at 25°C only.

## 8.45 head\_drop

**head\_drop** = <value | ?> <unit>  
**head\_drop** <interval> = <value | ?> <unit>

**head\_drop fluid** = <boundary fluid>  
**head\_drop** <interval> **fluid** = <boundary fluid>  
**head\_drop** <left | right> **fluid** = <boundary fluid>  
**head\_drop** <left | right> <interval> **fluid** = <boundary fluid>

**Flow** →  interval  head drop  ▼ cm ▼

Use the “head\_drop” (also: “pot\_drop”) command to set a value for the decrease in hydraulic head (in length units) or hydraulic potential (in units of pressure) across the medium. This decrease provides the driving force for fluid flow. The program sets hydraulic head or potential along the left boundary to the value specified in the command, and sets head or potential along the right boundary to zero.

At each step in the simulation, the program uses the head or potential values at the boundaries, along with the permeability and fluid viscosity, to calculate the flow field.

Setting a positive head or potential drop drives flow from left to right, and a negative value drives flow in the opposite direction.

You set the head drop for a specific interval by including the interval name, or for all intervals by omitting it.

The “head\_drop” command also lets you specify the boundary fluids to set to the left and right sides of the domain, in the same fashion as you would use the “discharge” command.

The program recognizes hydraulic head in units of length, as listed in the **Units Recognized** chapter in the **GWB Reference Manual**, and hydraulic potential in units of pressure. By default, values are taken in atm.

Examples:

```
head_drop = 150 cm
pot_drop = 0.1 atm
head_drop start = 4 m phase2 = 2 m
```

By default, the head drop is set to zero. To return to the default value, enter the command

```
head_drop = ?
```

## 8.46 heat\_source

```
heat_source = <field variable | ?> <unit> <steady | transient> \
    <Tmin = <value | min | ?> <unit>> <Tmax = <value | max | ?> <unit>>
heat_source = <off | on | reset>
```

**Medium** →  internal heat source    +     ▼ cal/cm<sup>3</sup>/s ▼  
 min  ▼ C ▼    max  ▼ C ▼

Use the “heat\_source” command (also: “heat\_src”) to set the rate of internal heat production within the medium. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/cm<sup>3</sup>/s” is the default. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function. By default, the program does not account for internal heat production. The “?” argument resets the default value of zero.

The “Tmin” and “Tmax” keywords (also: “temp\_max”, “temp\_min”) prescribe the allowable temperature range for the simulation. Set the keywords to specific temperatures, or to “min” or “max”, which respectively represent the lowest and highest temperatures considered in the thermo dataset, as loaded at run time. If you set the temperature range directly, it will be bracketed to that of thermo dataset. For example, the command



```
heat_source Tmin = -1000 C, Tmax = 1000 C
```

is functionally the same as

```
heat_source Tmin = 0 C, Tmax = 300 C
```

when the “thermo.tdat” dataset is loaded, since the dataset’s range is 0°C to 300°C.

The upper and lower temperature bounds serve two purposes. First, the simulation will give an error message and stop if temperature at any point in the domain falls more than 5°C less than the minimum value, or exceeds the maximum value by more than this amount. Second, unless the “extrap” option is set, the model will load for the simulation only those species for which log  $K$  values are available in the thermodynamic dataset over the specified temperature range.

The temperature bounds specified with the “Tmin” and “Tmax” keywords are the same as those set with the “span” command: The command

```
heat_source Tmin = 20 C, Tmax = 100 C
```

may be equivalently expressed

```
span 20 C to 100 C
```

as long as the heat source option is enabled.

Values for the keywords default to the temperature span of the thermodynamic database, as set in the database header. Keyword “off” disables the heat source, leaving the source rate and temperature range intact, and keyword “on” re-enables the heat source; “reset” disables the source and discards any settings for the source rate and temperature range.

## 8.47 height

```
height = <value> <unit>
```

**Domain** → height (z)  ▼ cm ▼

Use the “height” command to set the height along  $z$  of the medium. You can use any unit of distance listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cm” is the default.

By default, the domain is 1 cm in height. See also the “length” and “width” commands.

## 8.48 help

```
help <help | subject>
```

**Help**

Use the “help” command (also: “man”) to display online documentation for the GWB package. The command “help help” shows a list of the types of documentation available.

## 8.49 history

```
history
```

**Run** → **History...**

Use the “history” command to view a list of previously executed commands. You can re-execute commands in the **History...** dialog box by double-clicking on a command, selecting one or more commands, and then choosing **Execute**, or with a history substitution command, as explained in the **Command Line Interface** chapter in the **GWB Reference Manual**.

## 8.50 hydrogen-2

```
hydrogen-2 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “hydrogen-2” command (also, “2-H”) to set the  $^2\text{H}$  isotopic composition of the initial fluid, boundary fluids, reactant species (aqueous species, minerals, end members, gases, or oxides), or segregated minerals. The composition may be set on any scale (e.g., SMOW), but you must be consistent throughout the calculation. Example:

```
scope initial
  hydrogen-2 fluid = -10
scope inlet-1
  hydrogen-2 fluid = +4
scope
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
hydrogen-2 remove
hydrogen-2 off
```

clear all settings for  $^2\text{H}$  isotopes in the current scope.

See also the “<isotope>” section above, and the “carbon-13”, “oxygen-18”, and “sulfur-34” commands.

## 8.51 inert

```
inert = <field variable | ?> <units>
```

**Medium** → inert volume     ▼

Use the “inert” command to set the volume of non-reacting space in the system. You may set a value in units of volume, including  $\text{cm}^3$ ,  $\text{m}^3$ , and  $l$ , as well as volume% and “vol. fract.”. The default setting is zero and the default unit is  $\text{cm}^3$ .

Assuming you have not set a value for the initial fluid fraction in the system using the “porosity” command, the program figures the porosity over the course of the calculation as a derived variable. Specifically, it divides the fluid volume by the sum of the fluid volume, mineral volume, and inert volume, and reports this value as a result.

When you have set a value for initial porosity with the “porosity” command, on the other hand, the program works in the contrary sense. In this case, it calculates the inert volume as that required to form a system of the specified initial porosity; the program now ignores any entry you may have set using the “inert” command.

To restore the default state, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show”.

## 8.52 interval

```
interval <interval> start = <value> <unit>
interval <interval> fluid = <boundary fluid>
interval <interval> <left | right> = <boundary fluid>
interval <interval> rename <new name>
```

**Intervals** →  ▼ at  ▼ day ▼ left  ▼ right  ▼

The “interval” command sets the properties of a named simulation interval, creating it if necessary. The first interval is named “start”, and the terminating point for the run is labeled “end”, but you can create any number of intervening intervals.

For each interval, you set the starting time with keyword “start”, and the boundary fluid to use during the simulation interval with keyword “fluid”. The default time is zero

days, and you create fluids with the “scope” command. In the absence of a keyword, the command expects the starting time.

You can use keyword combinations “left fluid” and “right fluid” to set separate fluids on the left and right sides of the domain.

You can also rename the interval, using keyword “rename”. You cannot, however, rename “start” or “end”.

For example, the commands

```
interval start fluid = fluid1
interval phase2 start = 1 year, fluid = fluid2
interval phase3 start = 4 years, left fluid = fluid3A, right fluid = fluid3B
interval end 10 years
```

configure a simulation spanning 10 years and consisting of three reaction intervals, each with a different boundary fluid, or fluids.

### 8.53 isotope\_data

```
isotope_data <dataset>
```

**File** → **Open** → **Isotope Data...**

Use the “isotope\_data” command (also: “idata”) to set the name of the database containing isotope fractionation factors for species, minerals, and gases. Example:

```
isotope_data Isotope.mydata
```

### 8.54 itmax

```
itmax = <value | ?>
```

**Config** → **Iteration...**

Use the “itmax” command to set the maximum number of iterations that may be taken in an attempt to converge to a solution for the equations representing the distribution of chemical mass. By default, this variable is set to 999. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.



label that begins with the characters “redox”, such as “redox-1” or “redox-Fe”. For a microbial reaction, set a label that starts with “microbe”, such as “microbe-Ecoli”.

The rate law you specify in a “kinetic” command, by default, applies to the dissolution of a mineral or end member, dissociation of a complex, dissolution of a gas, or forward progress of a redox or microbial reaction. The synonymous keywords “forward”, “dissolution”, and “dissociation” set this behavior. Including in a “kinetic” command the keyword “reverse” or its synonyms “precipitation”, “complexation”, “association”, or “exsolution” invokes the opposite behavior. In this case, the rate law applies to the reverse reaction: mineral or end member precipitation, complex association, or gas exsolution.

You can append the “transient” keyword when setting the following field variables: rate constant, specific surface area, activation energy, pre-exponential factor, nucleus area, and critical saturation index. If the variable is defined by an equation, script, or external function, it will then be re-evaluated continuously over the course of the run.

See also the “react” and “remove reactant” commands.

The following paragraphs apply to all types of kinetic reactions. You set the rate constant either directly using the “rate\_con” keyword, or by setting an activation energy and pre-exponential factor with keywords “act\_en” and “pre-exp”. In the absence of promoting and inhibiting species (see next paragraph), you set the rate constant and preexponential factor in (1) mol/cm<sup>2</sup> s for mineral, end member, and gas transfer reactions, (2) molal/s or molal/cm<sup>2</sup> s (the latter when accounting for heterogeneous catalysis) for complexation and redox reactions, and (3) mol/mg s for microbial reactions. The activation energy is specified in J/mol. Example:

```
kinetic "Albite low" rate_con = 1e-15
```

You can set “rate\_con”, “act\_en”, and “pre-exp” as field variables (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

You use the “apower” or “mpower” (also “apow” or “mpow”) keyword to specify any promoting or inhibiting species in the kinetic rate law. Keyword “apower” sets the exponent of a species activity, and “mpower” the exponent of a species molality. Promoting species have positive powers, and the powers of inhibiting species are negative. For example, the command

```
kinetic "Albite low" apower(H+) = 1
```

sets H<sup>+</sup> as a promoting species, the activity of which is raised to a power of one.

You can use aqueous species, minerals (represented by activity, which is one, or molality), end members (activity or mole fraction), gas species (fugacity or partial pressure), surface complexes (molal concentration), and solvent water (activity) as promoting and inhibiting species. When setting an end member, use keyword “xpower” to set the rate law in terms of mole fraction. When setting a gas, use keyword “fpower” to set the rate law in terms of fugacity, and “ppower” to use partial pressure, instead. The generic keyword “power” sets the activity of the solvent or an aqueous species,

the activity of an end member, the fugacity of a gas, and the molality of a surface species.

The “order1” and “order2” keywords set nonlinear rate laws. Keyword “order1” represents the power of the  $Q/K$  term, and “order2” represents the power of the  $(1 - Q/K)$  term.

Use the “rate\_law” keyword to set the form of the kinetic rate law for a specific mineral, end member, species, gas, redox reaction, or microbial metabolism. You may set the keyword equal to (1) a character string containing the rate law, (2) the name of a file containing a basic-like script, or (3) the name of a function in a library. The name of a file containing a rate law script must end in “.bas”. To specify a function from a library, set the name of a dynamic link library (DLL) separated from the function name by a colon (“:”), such as “rate\_laws.dll:my\_ratelaw”; the library file must end in “.dll”. To return to the program’s built-in rate law, enter “rate\_law = off” or “rate\_law = ?”.

The following paragraphs apply to dissolution and precipitation reactions. You set the specific surface area of a kinetic mineral or end member (in  $\text{cm}^2/\text{g}$ ) with the “surface” keyword. For example,

```
kinetic "Albite low" surface = 1000
```

The “cross-affinity” option lets you use the saturation state of one mineral to model the reaction rate of another (or of an end member), as is sometimes useful for example in studying glass dissolution. To do so, you use the “xaffin” option. For (a hypothetical) example, the command

```
kinetic Quartz xaffin = Cristobalite
```

causes the program to calculate the reaction rate of quartz according to the fluid’s saturation state with respect to cristobalite. The command

```
kinetic Quartz xaffin = OFF
```

turns off the option.

Finally, you use the “nucleus” and “critSI” keywords to set the area available for nucleation (in  $\text{cm}^2/\text{cm}^3$  fluid volume) and the critical saturation index above which the mineral or end member can nucleate. Each of these values, by default, is zero.

Keywords “surface”, “nucleus”, and “critSI” can be set as field variables (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

The following paragraphs apply to reactions for aqueous and surface complexes. When you specify a kinetic reaction for the association of an aqueous complex or surface complex, or its dissociation, you can set the complex’s initial concentration directly. The concentration can be set heterogeneously, as a field variable. If you do not specify an initial concentration, or set an entry of “?”, the program takes the complex at the start of the simulation to be in equilibrium with the initial fluid.

## GWB Command Reference

---

You specify the initial concentration within a “kinetic” command or as a separate command line. For example, the commands

```
kinetic AIF++ rate_con = 3.3e-6, mpow(AIF++) = 1
AIF++ = 1 umol/kg
```

are equivalent to

```
kinetic AIF++ 1 umol/kg rate_con = 3.3e-6, mpow(AIF++) = 1
```

Either case defines a kinetic reaction for decomposition of the  $\text{AIF}^{++}$  ion pair, setting it initially to a free concentration of  $1 \mu\text{mol kg}^{-1}$ .

In **X1t** and **X2t**, you can similarly set the species' concentration within the boundary fluids directly

```
kinetic AIF++
scope fluid-1
  AIF++ = 4 umol/kg
scope fluid-2
  AIF++ = 8 umol/kg
```

or on a “kinetic” command

```
kinetic AIF++ fluid-1 = 4 umol/kg, fluid-2 = 8 umol/kg
```

If you do not set a value for a boundary fluid, or set it to “?”, the program takes the species to be in equilibrium with the fluid.

The following paragraphs apply to gas transfer reactions. Use the “f\_ext” keyword to specify the fugacity of the gas in question in the external reservoir, or keyword “P\_ext” to set its partial pressure. In the latter case, you may append a pressure unit; the default is bar. Keyword “contact” sets the contact area between fluid and external reservoir, in  $\text{cm}^2/\text{kg}$  of water. Example:

```
kinetic CO2(g) f_ext = 10^-3.5, contact = 10
```

Both values can be set as field variables, as described in the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**.

If you do not set a value for the gas' external fugacity, or set “f\_ext = ?”, the program uses the fugacity in the initial fluid, at the start of the simulation, and the external fugacity.

The following paragraphs apply to redox reactions. You set the form of the redox reaction to be considered as a character string, using the “rxn” (or “reaction”) keyword. For example,



```
kinetic redox-1 rxn = "Fe++ + 1/4 O2(aq) + H+ -> Fe+++ + 1/2 H2O"
```

To specify that the reaction be promoted by a heterogeneous catalyst, set keyword “catalyst” to the name of the catalyzing mineral, or simply to “on”. In the former case, you use keyword “surface” to set the specific surface area of the catalytic mineral (in  $\text{cm}^2/\text{g}$ ). If you have set “catalyst = on”, however, you use the “surface” keyword to set total catalytic area, in  $\text{cm}^2$ . Setting “catalyst = off” disables the catalysis feature.

To set an enzymatically promoted reaction, set keyword “me” to the name of the aqueous species serving as the enzyme, or simply to the value to be used as the enzyme’s molality. In the former case, the program tracks the enzyme molality  $m_E$  over the course of the simulation from the calculated distribution of species. If you have set a numeric value for  $m_E$  using the “mE” keyword, the program uses this value directly. You may alternatively specify the enzyme species or its activity  $a_E$  using keyword “aE”, in which case variables  $m_E$ ,  $m_A$ , and  $m_P$  in the rate law are replaced by the activities  $a_E$ ,  $a_A$ , and  $a_P$ .

For an enzymatic reaction, you further set the half-saturation constants  $K_A$  and  $K_P$  for the forward and reverse reactions in molal with the “KA” and “KP” keywords. You must set a value for  $K_A$ , but may omit  $K_P$ , in which case the  $m_P/K_P$  term in the rate law will be ignored. Setting “enzyme = off” disables the enzyme feature.

The following paragraphs apply to microbial reactions. You set the form of the metabolic reaction using the “rxn” (or “reaction”) keyword, in the same manner as with redox reactions. For example,

```
kinetic microbe-1 rxn = "CH4(aq) + 2 O2(aq) -> HCO3- + H+ + H2O"
```

Set the half-saturation constants  $K_D$  and  $K_A$  for the electron donating and accepting reactions with the “KD” and “KA” keywords. These values default to zero.

You set the powers of species in the numerator of the rate law with the “mpower” keyword, as with other types of kinetic reactions. Use keywords “mpowerD” and “mpowerA” (or “mpowD” and “mpowA”) to set the powers  $p_D$ , etc., of species from the electron donating and accepting reactions, respectively, within the product functions in the rate law’s denominator. For example,

```
kinetic microbe-1 mpower(CH4(aq)) = 1, mpowerD(CH4(aq)) = 1
```

sets the power of the electron-donating species  $\text{CH}_4(\text{aq})$  to one in both the rate law numerator and denominator. Keywords “PKD” and “PKA” set the overall powers  $p_{KD}$  and  $p_{KA}$  of the electron donating and accepting terms in the denominator of the rate law; by default, these are one.

You set the free energy  $\Delta G_{\text{ATP}}$  of ATP hydrolysis (in  $\text{kJ/mol}$ ) with the “ATP\_energy” keyword, and the value of  $n_{\text{ATP}}$  with keyword “ATP\_number”. These values default to zero.

Use the “biomass” keyword to set the initial biomass concentration, in mg/kg. You can set this value as a field variable (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

The “growth\_yield” keyword sets the microbe’s growth yield in mg biomass/mol of reaction progress, and “decay\_con” sets its decay constant in s<sup>-1</sup>; both values default to zero.

## 8.57 left

```
left fluid = <boundary fluid>
left <interval> fluid = <boundary fluid>
left = <normal | inlet | outlet>
```

**Flow** → boundaries: left  ▼

Use the “left” command to control the boundary condition at the left side of the domain: the fluid to appear at the boundary, and whether solutes can diffuse and disperse, and heat conduct, across the boundary.

If the command contains the name of a simulation interval, subsequent settings on the command apply to that interval, rather than the simulation as a whole. This behavior continues to the end of the command, or until the name of another interval is encountered.

You use the “fluid” keyword to specify the boundary fluid to appear on the left boundary, for the entire simulation or an individual interval. For example, the commands

```
left fluid = ambient_water
left phase2 fluid = pit_water
```

set fluid “ambient\_water” on the left boundary over the course of the simulation, except during interval “phase2”, in which case fluid “pit\_water” is found on the boundary.

The program by default uses “inlet/free outlet” boundary conditions: where fluid flows into the domain, solute mass can advect, diffuse, and disperse across the boundary, but where it exits the domain, transport occurs by advection alone. Similarly, heat can conduct across an inlet, but not an outlet.

You can force the program to treat the left side of the domain as an inlet or free outlet boundary, regardless of the direction of fluid migration. Keyword “inlet” causes the program to allow diffusion, hydrodynamic dispersion and heat conduction across the left bound, whereas keyword “outlet” creates an insulated boundary sealed against transport by these processes:


```
left outlet
```

In either case, the advective transport of solute mass and heat depends on the direction and rate of fluid migration, and is not affected by the command setting

See also the “right”, “discharge”, “head drop”, and “interval” commands.

## 8.58 length

**length** = <value> <unit>

**Domain** →  length (x)  ▼ cm ▼

Use the “length” command to set the length along  $x$  of the medium. You can use any unit of distance listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cm” is the default. By default, the domain is 100 cm long. See also the “width” and “height” commands.

## 8.59 log

**log** <variable> = <value>

Use the “log” command to set variables on a logarithmic scale. Examples:

log fugacity O2(g) = -65  
log activity U++++ = -10

## 8.60 mobility

**mobility** = <surface type> = <field variable> <steady | transient>

**mobility** = <surface type> = <value>

**mobility** = <surface type> <initial = <field variable> <steady | transient>> \  
<boundary fluid> = <value>

**Config** → **Sorbing Surfaces...** → mobility  +  ▼

+ fluids → fluid  ▼

or **File** → **Open** → **Sorbing Surfaces...**

Use the “mobility” command to set up a complexing surface in your model as a mobile colloid. A mobile colloid is composed of the mineral (or minerals) associated with a complexing surface, as well as the ion complexes present on that surface. Only datasets with model type “two-layer”, “three-layer”, or “cd-music” as set in the dataset header are surface complexation models, and hence only those datasets can be used to form a mobile colloid.

Mobility refers to the fraction of the surface in question that can move in the model by advection and dispersion. A surface with a mobility of one moves freely, whereas a mobility of zero sets the surface to be stationary. Intermediate values arise, for

example, when some of the surface is attached to the medium, or when colloid motion is impeded by electrostatic interactions. By default in the software the mobility of any surface is zero.

To set a mobile colloid, begin by reading in a surface complexation dataset using the “surface\_data” command. Then, use the “mobility” command, referencing the surface’s label, to set the colloid’s mobility. The label is given at the head of the surface dataset, on a line beginning “Surface type”. The label in dataset “FeOH.sdat”, for example, is “HFO”. If you omit the label, the program will assume you are referring to the surface complexation dataset most recently read.

You can define the mobility of the domain (i.e., “initial” scope) as a field variable, which means you can have the program calculate mobility using an equation, script, or compiled function you provide. When you set the “transient” keyword, the program upon undertaking each time step in the simulation evaluates mobility at each nodal block. In the “steady” case, which is the default, the program evaluates mobility at each block just once, at the start of the run.

You can further define colloid mobility in a boundary fluid. If you do not set a value, mobility in an inlet stream is taken as the initial value assigned the first interior node in the simulation.

In its simplest form, the command sets mobility for the current scope, either the initial system or a boundary fluid. You can, however, set mobility for one or more scopes with a single command, by citing each scope’s name.

For example, since “HFO” is the label for the surface defined by dataset “FeOH.sdat”, the commands

```
surface_data FeOH.sdat
scope initial
  mobility HFO = 100%
scope inlet_fluid1
  mobility HFO = 0%
```

are equivalent to

```
surface_data FeOH.sdat
mobility HFO initial = 100%, inlet_fluid1 = 0%
```

Restore the default behavior of immobility by entering a command such as

```
mobility HFO ?
```

## 8.61 no-precip

**no-precip** <off>

**Config** → **Options...**

Use the “no-precip” command (also: “noprecip”) to prevent new minerals from precipitating over the course of a simulation. By default, they are allowed to precipitate. Use:

```
no-precip
no-precip off
```

See also the “precip” command.

## 8.62 nswap

**nswap** = <value | ?>

**Config** → **Iteration...**

Use the “nswap” command to set the maximum number of times that the program may swap entries in the basis in an attempt to converge to a stable mineral assemblage. By default, this variable is set to 30. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 8.63 Nx

**Nx** = <value | ?>

**Domain** → nodes (Nx)  ▼

Use the “Nx” command to set in the simulation the number of nodal blocks along the  $x$  direction. By default, the program divides the medium into ten blocks along  $x$ .

## 8.64 oxygen-18

**oxygen-18** <fluid | reactant | segregated mineral> = <value>

**Config** → **Isotopes...**

Use the “oxygen-18” command (also, “18-O”) to set the  $^{18}\text{O}$  isotopic composition of the initial fluid, boundary fluids, reactant species (aqueous species, minerals, end

## GWB Command Reference

---

members, gases, or oxides), and segregated minerals. The composition may be set on any scale (e.g., SMOW), but you must be consistent throughout the calculation. Example:

```
scope initial
  oxygen-18 fluid = -10
scope inlet-1
  oxygen-18 fluid = +4
scope
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
oxygen-18 remove
oxygen-18 off
```

clear all settings for  $^{18}\text{O}$  isotopes in the current scope.

See also the “<isotope>” section above, and the “carbon-13”, “hydrogen-2”, and “sulfur-34” commands.

### 8.65 pause

```
pause
```

Use the “pause” command to cause **X1t** to pause temporarily during input. This command is useful when you are debugging scripts.

### 8.66 pe

```
pe = <field variable>
```

```
Initial or Fluids →  ▼ pe ▼
```

Use the “pe” command to set oxidation state in the initial system in terms of pe. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). Example:

```
pe = 10
```

is equivalent to

```
log activity e- = -10
```

where “e-” is the electron. See also the “activity”, “Eh”, “pH”, “fugacity”, “fix”, and “slide” commands.

## 8.67 permeability

```
permeability <intercept = field variable | ?> <unit> <steady | transient> \
  <porosity = field variable | ?> <steady | transient> \
  <mineral = field variable | ?> <steady | transient>
```

**Medium** → A (porosity)  +  ▼

B (intercept)  +  ▼ darcy ▼

You use the “permeability” command to set the correlation by which the program calculates sediment permeability. The correlation gives log permeability in any of the units listed in the **Units Recognized** chapter of the **GWB Reference Manual** (darcys by default) as a linear function of the porosity (expressed as a volume fraction) of a nodal block and, optionally, the volume fractions of one or more minerals. The “transient” keyword causes the model to evaluate the coefficient in question continuously over the course of the simulation, if it is set with an equation, script, or function.

Examples:

```
permeability intercept = -11 cm2 porosity = 15
permeability Kaolinite = -8
```

The latter command adds a term for the mineral Kaolinite to the existing correlation. To remove a term from the correlation, set a value of “?”. The entry

```
permeability Kaolinite = ?
```

for example, removes the correlation entry for that mineral.

The default correlation is

$$\log k = -5 + 15\phi$$

where  $k$  is permeability in darcys and  $\phi$  is porosity (expressed as a fraction).

## 8.68 pH

**pH** = <field variable>

**Initial** or **Fluids** →  ▼ pH ▼

Use the “pH” command to set pH in the initial system. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). Example:

pH = 5

is equivalent to

log activity H+ = -5

See also the “activity”, “Eh”, “pe”, “fugacity”, “fix”, and “slide” commands.

## 8.69 phrqpitz

**phrqpitz**

**File** → **Open** → **Thermo Data...** → thermo\_phrqpitz.tdat

Use the “phrqpitz” command to set the program to calculate species’ activity coefficients using the Harvie-Møller-Weare equations, as implemented in the USGS program PHRQPITZ. Executing this command automatically sets the input dataset of thermodynamic data to “thermo\_phrqpitz.tdat”. Note that dataset “thermo\_phrqpitz.tdat” is primarily intended to support calculations at or near 25°C.

## 8.70 pitz\_dgamma

**pitz\_dgamma** = <value | ?>

**Config** → **Iteration...**

Use the “pitz\_dgamma” command to control the relative change in an activity coefficient’s value the program allows during each Newton-Raphson iteration, when the Harvie-Møller-Weare activity model has been invoked. By default, the program allows a 10% change, which corresponds to a value of 0.1.



## 8.71 pitz\_precon

```
pitz_precon = <value | ?>
```

**Config** → **Iteration...**

Use the “pitz\_precon” command to control the maximum number of passes the program takes through the pre-conditioning loop before beginning a Newton-Raphson iteration, when a virial activity model (Harvie-Møller-Weare or SIT) has been invoked. By default, the program makes up to 10 passes. In cases of difficult convergence, counter-intuitively, it can sometimes be beneficial to decrease this value.

## 8.72 pitz\_relax

```
pitz_relax = <value | ?>
```

**Config** → **Iteration...**

The “pitz\_relax” command controls under-relaxation when evaluating the Harvie-Møller-Weare equations. The program at each Newton-Raphson iteration assigns activity coefficients as a weighted average of the newly calculated value and the corresponding value at the previous iteration level. Setting pitz\_relax to zero eliminates under-relaxation, so the newly calculated values are used directly; a value of one, in contrast, should be avoided because it would prevent the activity coefficients from being updated. By default, the program carries an under-relaxation factor of 0.5.

## 8.73 plot

```
plot <character | binary> <on | off>
```

**Config** → **Output...**

Use the “plot” command to set the format of the “X1t\_plot.xtp” dataset. The dataset is written in XML, a standard format that is easy to parse for use with alternative plotting programs. Numerical data in the dataset can be represented in either standard decimal notation (keyword “character”) for user readability or a binary encoding (keyword “binary”) that maintains full precision of data. The default format, XML with binary encoded data, also zips the output file to reduce output size and improve file opening speed. The command “plot off” causes **X1t** to bypass writing calculation results to the “X1t\_plot.xtp” dataset, which is used to pass input to **Xtplot**. By default, the program writes output to the dataset. The command “plot on” (or just “plot”) re-enables the output. To see the current setting, type “show print”.

## 8.74 pluses

**pluses** <off | on | multicore | banner>

**Config** → **Stepping...**

The “pluses” command controls whether the program writes a plus sign (“+”) to the screen each time it converges to a solution of the chemical equations at a nodal block (keyword “on”); by default, it does not (keyword “off”). You can also set the program to write a distinct symbol for each thread active in tracing the simulation (keyword “multicore”), allowing you to observe the model as it runs in parallel. You can also specify the program write a banner giving details such as the number of iterations required for convergence and the final residuals (keyword “banner”). The latter option produces unwieldy amounts of output, except for simple simulations.

## 8.75 polymorphs

**polymorphs** <mineral>

**Config** → **Show...** → Minerals → *select mineral*

Use the “polymorphs” command to list each polymorph (mineral identical in composition to the specified mineral) in the current database. Example:

polymorphs Albite

## 8.76 porosity

**porosity =** <field variable | ?>

**Medium** → porosity

Use the “porosity” command to set (as a volume fraction) the initial porosity of the medium. Porosity, the fraction of a nodal block occupied by fluid, is the ratio of fluid volume to the sum of fluid, mineral, and inert volume.

The examples

porosity = 0.30  
porosity = 30%

are equivalent.

When you specify the porosity, the program will at each nodal block figure the difference between the volume taken up initially by minerals and fluid, and the block's bulk volume. The program assigns this difference as inert, non-reactive volume (see the “inert” command). In this case, the program ignores any settings that may have been made with the “inert” command.

When you do not specify an initial porosity with the “porosity” command, on the other hand, the program calculates it from volumes in the nodal block of fluid, minerals, and inert space. To restore this default behavior, enter the command with an argument of “?”.

## 8.77 precip

```
precip <off>
```

**Config** → **Options...**

Use the “precip off” command to prevent new minerals from precipitating over the course of a simulation. By default, they are allowed to precipitate. Use

```
precip
precip off
```

See also the “no-precip” command.

## 8.78 press\_model

```
press_model <Tsonopoulos | Peng-Robinson | Spycher-Reed | default | off>
```

**Config** → **Options...**

The “press\_model” command (also: “pressure\_model”) lets you control the method used to calculate fugacity coefficients and gas partial pressures. Three nonideal pressure models are coded in the software: Tsonopoulos, Peng-Robinson, and Spycher-Reed, as described in the **GWB Essentials Guide**.

By default, the pressure model is taken from the header lines of the thermo dataset in use, but you can use the “press\_model” command to override the default setting. Keywords “Tsonopoulos”, “Peng-Robinson”, and “Spycher-Reed” set the pressure model directly (you need only enter the first three letters), whereas “default” returns to the setting in the thermo dataset, and “off” (also: “ideal”) disables the feature, forcing all fugacity coefficients to one.

Examples:

```
press_model Peng-Robinson
press_model default
```

## 8.79 print

```
print <option> = <long | short | none>
print <off | on>
print <numeric | alphabetic>
```

**Config** → **Output...**

Use the “print” command (also: “printout”) to control the amount of detail to be written into the “X1t\_output.txt” dataset. For example, the dataset can contain information about each aqueous species, information on only species with concentrations greater than  $10^{-8}$  molal, or no species information.

By default, the program does not produce these datasets, which can become quite lengthy. The command “print on” (or just “print”) causes them to be written, and “print off” disables them.

Options, which may be abbreviated to three letters, and their default settings are:

species	short
surfaces	long
saturation	short
gases	long
basis	none
orig_basis	long
elements	long
reactions	none
stagnant	none

The “stagnant” option includes in the printout the diffusive profile within the stagnant zone, when the dual porosity feature is in use.

The “print” command can also be used to arrange entries in the output dataset either numerically or alphabetically:

```
print numeric
print alphabetic
```

To see the current print settings, type “show print”.

## 8.80 pwd

**pwd**

**File** → Working Directory...

The “pwd” command returns the name of the current working directory. The command has the same effect as typing “show directory”. See the “chdir” command.

## 8.81 quit

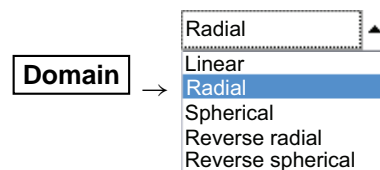
**quit**

**File** → Quit

Use the “quit” command (also: “end”, “q”) to exit the program. Touching **Ctrl+Q** serves the same purpose as the “quit” command.

## 8.82 radial

**radial** <r1 = value> <unit> <r2 = value> <unit>  
 <angle = value> <unit> <normal | reverse>  
**radial** <off | on>



Use the “radial” command (also: “wedge”) to set a radial flow system extending from radius “r1” to radius “r2”, where “r1” is less than “r2”. Radii may be entered in units of distance (see the **Units Recognized** chapter in the **GWB Reference Manual**); “cm” is the default. Keyword “angle” gives the angle of divergence of the medium, in “radians” (the default) or “degrees”. By default, this value is 0.1 radians. To return to a linear geometry, type “radial off”. The “normal” keyword sets the domain to narrow from right to left; the “reverse” keyword sets it narrowing from left to right. See also the “spherical” command.

## 8.83 ratio

**ratio** <species ratio> = <value>

**Initial** or **Fluids** →  ▼ ratio ▼

Use the “ratio” command to constrain an activity ratio in the initial system. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). Example:

```
swap Ca++/Na+^2 for Ca++
ratio Ca++/Na+^2 = 0.2
```

See also the “activity”, “pH”, “Eh”, “pe”, “fugacity”, “fix”, and “slide” commands.

## 8.84 react

**react** <amount (field variable)> <unit> <as <element symbol>> \  
<species | mineral | end member | gas | oxide> <cutoff> = <field variable>

**Reactants** → add → 

Simple	Aqueous...
Fixed	Mineral...
Sliding	Gas...
Kinetic	Oxide...

Use the “react” command (abbrev.: “rct”) to define the reactants for the current simulation. To set a kinetic rate law for a reactant, use the “kinetic” command.

Units for the amount of reactant to add over a reaction path can be:

mol	mmol	umol	nmol	
kg	g	mg	ug	ng
eq	meq	ueq	neq	
cm3	m3	km3	l	
mol/kg	mmol/kg	umol/kg	nmol/kg	
molal	mmolal	umolal	nmolal	
mol/l	mmol/l	umol/l	nmol/l	
g/kg	mg/kg	ug/kg	ng/kg	
wt%	"wt fraction"			
g/l	mg/l	ug/l	ng/l	
eq/kg	meq/kg	ueq/kg	neq/kg	
eq/l	meq/l	ueq/l	neq/l	

Units of mass or volume can be expressed per volume of the porous medium. Examples:

mol/cm <sup>3</sup>	g/cm <sup>3</sup>
mmol/m <sup>3</sup>	ug/m <sup>3</sup>
volume%	"vol. fract"

Units of mass or volume can be set as absolute rates by appending “/s”, “/day”, “/yr”, or “/m.y.”. For example,

mmol/s	g/day	cm <sup>3</sup> /yr	
mol/kg/s	mg/kg/day	cm <sup>3</sup> /kg/yr	
mol/cm <sup>3</sup> /s	kg/m <sup>3</sup> /yr	m <sup>3</sup> /m <sup>3</sup> /yr	volume%/m.y.

The latter units refer to rates per cm<sup>3</sup> or m<sup>3</sup> of porous medium, per unit time.

Use the “as” keyword to specify reactant masses as elemental equivalents. For example, the command

```
react 10 umol/kg CH3COO- as C
```

specifies 5 umol/kg of acetate ion, since each acetate contains two carbons, whereas

```
react 20 mg/kg SO4-- as S
```

would cause the program to add 59.9 mg/kg of sulfate, since the ion’s mole weight is about 3 times that of sulfur itself.

You can set a cutoff to limit the amount of a reactant. For example, if you set the amount of a reactant to two moles and set a cutoff of one, then **X1t** will add one mole of the reactant over the first half of the path and none over the second half. Enter the cutoff value in the same units as the amount of reactant. Examples:

```
react 10 grams Quartz
react 1e-2 mol Muscovite cutoff = .5e-2
react .01 mol/day HCl
```

See also the “kinetic” and “remove reactant” commands.

## 8.85 read

```
read <dataset>
```

**File** → **Open** → **Read Script...**

Use the “read” command to begin reading commands from a script stored in a dataset. Example:

```
read Seawater
```

Control returns to the user after the script has been read, unless the script contains a “quit” command. You can also use the “read” command in place of the “data” or “surface\_data” command to read a thermo or surface reaction dataset.

When typing a “read” command, you can use the spelling completion feature to complete dataset names: touch “[tab]” or “[esc]” to cycle through the possible completions, or **Ctrl+D** to list possible completions.

## 8.86 remove

```
remove <basis specie(s)> <solid solution(s)> <reactant(s)> <interval(s)> \
    <boundary fluid(s)>
remove basis <basis specie(s)>
remove solid_solution <solid solution(s)>
remove reactant <reactant(s)>
remove interval <interval(s)>
remove scope <boundary fluid(s)>
```

**Initial** or **Fluids** or **Reactants** or **Intervals** or

**Config** → **Solid Solutions...** →

*select entry(ies)* →

Use the “remove” command (also: “rm”) to eliminate one or more basis entries or reactants from consideration in the calculation. Example:

```
remove Na+
remove Quartz Calcite
remove reactant H2O
```

Components can be reentered into the basis using the “swap”, “add”, “activity”, and “fugacity” commands. You can also use the “remove” command to remove solid solutions configured in a run (but not those set in the thermo dataset):

```
remove solid_solution mySS
```

You can also use the “remove” command to remove boundary fluids and simulation intervals

```
remove scope inlet_fluid1
remove interval phase2
```

from the run configuration.



## 8.87 report

```
report <option>  
report set_digits <value>  
report set_node <index>
```

Once the program has completed a calculation, you can use the “report” command to return aspects of the calculation results. You can use this command from the **Command** pane to query the program interactively. More significantly, the “report” command provides a means of communicating calculation results to control scripts and external programs running GWB programs by remote control.

For arguments available, see the **Report Command** chapter in the **GWB Reference Manual**. For details about control scripts and the plug-in feature, see the **Control Scripts** and **Plug-in Feature** chapters of the **GWB Reference Manual**. Use the “set\_digits” argument to set the number of significant digits in the values reported, and the “set\_node” argument to identify the node of interest.

## 8.88 reset

```
reset  
reset system  
reset reactants  
reset variables  
  
File → Reset Configuration
```

Use the “reset” command to begin defining the chemical system again with a clean slate. Your current settings will be lost, and all options will be returned to their default states. The command, however, does not alter the setting for the thermo dataset. The “reset system” command resets only the initial system. Similarly, typing “reset reactants” resets the reactant system, and “reset variables” sets each settable variable to its default value.

## 8.89 resume

```
resume  
  
File → Resume
```

Use the “resume” command to restore the configuration from when you last exited the program. The program configuration is stored in a file “x1t\_resume.x1t” in your profile directory (e.g., “c:\Documents and Settings\jones\Application Data\GWB”). Your current configuration is discarded.

## 8.90 right

```
right fluid = <boundary fluid>
right <interval> fluid = <boundary fluid>
right = <normal | inlet | outlet>
```

**Flow** → boundaries: right  ▼

Use the “right” command to control the boundary condition at the right side of the domain: the fluid to appear at the boundary, and whether solutes can diffuse and disperse, and heat conduct, across the boundary.

If the command contains the name of a simulation interval, subsequent settings on the command apply to that interval, rather than the simulation as a whole. This behavior continues to the end of the command, or until the name of another interval is encountered.

You use the “fluid” keyword to specify the boundary fluid to appear on the right boundary, for the entire simulation or an individual interval. For example, the commands

```
right fluid = ambient_water
right phase2 fluid = pit_water
```

set fluid “ambient\_water” on the right boundary over the course of the simulation, except during interval “phase2”, in which case fluid “pit\_water” is found on the boundary.

The program by default uses “inlet/free outlet” boundary conditions: where fluid flows into the domain, solute mass can advect, diffuse, and disperse across the boundary, but where it exits the domain, transport occurs by advection alone. Similarly, heat can conduct across an inlet, but not an outlet.

You can force the program to treat the right side of the domain as an inlet or free outlet boundary, regardless of the direction of fluid migration. Keyword “inlet” causes the program to allow diffusion, hydrodynamic dispersion and heat conduction across the right bound, whereas keyword “outlet” creates an insulated boundary sealed against transport by these processes:

```
right outlet
```

In either case, the advective transport of solute mass and heat depends on the direction and rate of fluid migration, and is not affected by the command setting

See also the “left”, “discharge”, “head drop”, and “interval” commands.

## 8.91 save

```
save <dataset> <hex>
```

**File** → **Save As...**

Use the “save” command to write the current chemical system into a dataset in **X1t** format commands. The dataset can be used as an **X1t** input script. Examples:

```
save
save kspar.x1t
```

If you don't specify a dataset name, **X1t** will ask for one. The optional keyword “hex” causes the program to output numbers as hexadecimal values.

## 8.92 scope

```
scope <initial | boundary fluid>
scope <initial | boundary fluid> = <initial | boundary fluid>
scope <boundary fluid> rename <new name>
```

**Initial** or **Fluids**

**Initial** or **Fluids** → **copy**

The “scope” command defines whether subsequent user commands apply to the initial condition of the medium, to a boundary fluid, or globally, to all fluids that have been defined. The scope is initially set global. When you enter a scope command with “initial” as an argument

```
scope initial
```

the scope is set to the initial system. For any other argument

```
scope my_fluid
```

the scope is set to a boundary fluid of that name, creating it if necessary.

Entering the command without an argument

```
scope
```

returns the scope global, as does entering any command of global meaning, such as “react” or “kinetic”. To see the current scope, type “show scope”.

## GWB Command Reference

---

The “scope” command can also be used to copy constraints on fluid composition and temperature from one aspect of the calculation to another. For example, the command

```
scope my_fluid2 = my_fluid1
```

copies the basis constraints, temperature, TDS, and density from a boundary fluid “my\_fluid1” to another “my\_fluid2”.

```
scope inlet = initial
```

sets the inlet fluid to be the same as the fluid in the initial system.

You can rename a boundary fluid, but not the initial system, using the “rename” keyword.

### 8.93 script

```
script  
script end
```

Use the “script” command to mark the beginning, and optionally the end, of a control script. Control scripts differ from standard input files in that they can contain not only **X1t** commands, but control structures such as loops and if-else branches. Control scripts follow the Tcl syntax, described in [www.tcl.tk](http://www.tcl.tk) and [mini.net/tcl](http://mini.net/tcl), as well as several widely available textbooks.

Within a control script, filenames are written with double rather than single backslashes. For example, a “read” command might appear as

```
read GWB_files\\My_file.x1t
```

within a control script.

## 8.94 segregate

```

segregate <mineral(s)>
segregate <mineral> <value>
segregate <mineral> <initial value> <final value>
segregate <mineral> <value> Xi = <value> <value> Xi = <value>

```

**Config** → **Isotopes...** →

**Config** → **Isotopes...** → *select entry(ies)* →

The “segregate” command causes minerals to be isolated from isotopic exchange over the course of a reaction path. By default, a mineral in the equilibrium system remains in isotopic equilibrium with the fluid and other minerals. A segregated mineral, on the other hand, changes in isotopic composition only when it precipitates from solution; it alters the system’s composition only if it dissolves. Example:

```
segregate Quartz Calcite "Maximum Microcline"
```

Optionally, a fraction of a mineral’s mass may be isotopically segregated, and that fraction may vary linearly with reaction progress. Examples:

```

segregate Quartz 100%, Muscovite 7/10
segregate Ca-Saponite 100% 0%
segregate Ca-Saponite 80% Xi = .3, 20% Xi = .7

```

In the latter example, the program segregates 80% of the mass of Ca-Saponite until the reaction progress variable  $X_i$  reaches .3, decreases the segregated fractionation until it reaches 20% when  $X_i$  equals .7, and then holds the value constant until the end of the path. To display the isotopically segregated minerals, type “show isotopes”.

Note, you use the name of the corresponding mineral to segregate a solid solution end member.

## 8.95 show

```

show <option>
show <aqueous | minerals | solid_solutions | gases | oxides | surfaces> \
  <with | w/> <basis entry | string>

```

**Config** → **Show...**

Use the “show” command to display specific information about the current system or database. The options are:

show	show all	show altered
show aqueous	show basis	show boundary
show commands	show couples	show directory
show elements	show gases	show initial
show isotopes	show minerals	show oxides
show printout	show reactants	show scope
show show	show solid_solutions	show suppressed
show surfaces	show system	show variables

The command “show show” gives a list of show command options. When you type “show aqueous” (or “minerals”, “solid\_solutions”, “gases”, “oxides”, or “surfaces”), the program lists all entries of that type in the thermo database. The “solid\_solutions” option additionally includes solutions defined locally. A long form lets you limit the query to entries composed of a particular basis species or containing a text string in the name:

```
show aqueous with Al+++  
show minerals w/ chal
```

There is also a compound form of the “show couples” command:

```
show coupling reactions
```

This command produces a complete list of the redox couples, in reaction form.

### 8.96 simax

```
simax = <value | ?>
```

```
Config → Iteration...
```

The “simax” command sets in molal units the maximum value of the stoichiometric ionic strength used in calculating water activity when Helgeson’s B-dot Debye-Hückel model is employed. By default, this variable is set to 3 molal. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 8.97 slide

**slide** <unit> <species | gas> **to** <field variable>

Use the “slide” command to linearly adjust the activity of the specified species, fugacity of the gas, or an activity ratio toward <value>, which is attained at the end of the path. Note that the interpolation is made linearly on the logarithm of activity or fugacity if <value> is set as a log, and that <unit> can be “activity” or “fugacity” (“a” or “f” for short), “ratio”, “pH”, “pe”, or “Eh”, or omitted. Examples:

```
slide pH to 5
slide activity Cl- to 2/3
slide f CO2(g) to 10^-3.5
slide log f O2(g) to -65
```

## 8.98 solid\_solution

```
solid_solution <name = <label>> <type> <discrete | continuous> \
  <<mineral> tag = <label>> <contains> <<mineral> tag = <label>> \
  <from <value>> <to <value>> <step = <value>> <<variable> = <value>>
solid_solution <solid solution> <options>
solid_solution remove <solid solution>
```

**Config** → **Solid Solutions...** →

**Config** → **Solid Solutions...** → *select entry(ies)* →

Use the “solid\_solution” command (abbrev.: “solid\_soln”) to define binary solid solutions in addition to any provided in the current thermo database. Supply a name with the “name” keyword, or simply set one as the first argument.

You can select the type of solid solution, either “ideal” (the default), “site\_mixing”, “guggenheim”, “regular”, “cubic”, or “third\_order”. For a site-mixing solution, set the site number “bsite”, which defaults to 1. For a Guggenheim solution, set dimensionless

parameters “a0”, “a1”, and “a2” or a temperature expansion for their J/mol counterparts using coefficients “p1”-“p9”, each of which defaults to zero. For the other nonideal models, similarly, set coefficients “p1”, “p2”, etc., as appropriate. For more information, see **Solid solutions** under **Configuring the programs** in the **GWB Essentials Guide** and the **Thermo Datasets** chapter in the **GWB Reference Manual**. You can as well choose whether the solid solution is to be treated as continuous, the default option, or as a sequence of “discrete” minerals.

You need to specify two end members for a solid solution, each associated with an abbreviated “tag”. The tag cannot be the name of another entry in the database. You can set a composition range in terms of the mole fraction of the most recently referenced end member, using keywords “from” and “to”, as well as a compositional “step” separating tranches of a discrete solution; by default, composition ranges from zero to one and the step is one-twentieth of the range. Keyword “contains” may be inserted for clarity. Examples:

```
solid_solution name = my_ss Calcite contains Strontianite from 0% to 10% a0= 0.05
solid_solution my_ss Calcite Strontianite from 0 to .1 a0= 0.05
solid_solution Biotite site_mixing Annite Phlogopite bsite= 3
```

If you don’t specify a solid solution’s name, the app will create one automatically using the tags for the two end members, separated by a hyphen. If you don’t set a tag for an end member, the app will pick up the first six characters of the corresponding mineral’s name. For example, the command

```
solid_solution Calcite contains Strontianite
```

produces a solid solution named “Calcit-Stront”, whereas typing

```
solid_solution Calcite tag Cc contains Strontianite tag Str
```

yields the name “Cc-Str”. To redefine the properties of a solid solution, enter the name followed by the properties to be changed:

```
solid_solution my_ss a0 = 1 a1 = 2
```

(You may prefer to use the “alter” command to change solutions defined within the thermo database; the two commands are functionally equivalent.) The command

```
solid_solution remove my_ss
```

discards a previously defined solution.



## 8.99 sorbate

```
sorbate <exclude | include>
```

```
Config → Options...
```

Use the “sorbate” command to tell the program, when considering sorption onto surfaces (see the “surface\_data” command), whether to include or exclude sorbed species in figuring the composition of the initial system. By default, the program does not include sorbed species in this calculation. If you set the Ca<sup>++</sup> concentration to 15 mg/kg, for example, the initial system would contain that amount in the fluid and an additional amount sorbed onto mineral surfaces. If you type the command “sorbate include”, however, that amount would apply to the sum of the Ca<sup>++</sup> sorbed and in solution.

## 8.100 span

```
span <value | min | ?> <unit> <to> <value | max | ?> <unit>  
span <off | on | reset>
```

```
Config → Options...
```

The “span” command lets you set a temperature range across which stability must be defined for each species considered in the calculation. The lower limit may be identified with keyword “from” or “Tmin”, and the upper denoted by “to” or “Tmax”. A value of “?” unsets the bound in question.

With *T*-table datasets, simply setting “span = on” without specifying a range forces the program to load species’ log *K*s in polynomial mode. In other words, it will take log *K*s from polynomial fits against temperature to the stability constants in the thermo dataset, even if the calculation is performed entirely at one of the dataset’s principal temperatures. Otherwise, an isothermal calculation run at one of the principal temperatures (commonly 0°C, 25°C, 60°C, ...) takes log *K*s for aqueous species, minerals, and so on directly from the values given in the thermo dataset.

Temperature ranges of validity can also be set in the thermo dataset for the virial coefficients used to calculate “Pitzer” and SIT activity coefficients. The “span” command works in a similar manner in such cases, loading only virial coefficients with validity that spans the calculation’s temperature range.

As an example of using the “span” command, you may wish to compare results calculated at 25°C with those figured at 30°C. Since log *K* for a number of species in the thermo dataset is given at only 25°C, unless the “extrap” option is set, the former calculation may well take into account a somewhat larger set of species than the latter. Entering the command

```
span 25 C to 30 C
```

assures the same set of species is loaded in each case, and that both runs take log  $K$ s from polynomial fits.

Set the bounds to specific temperatures, or to “min” or “max”, which respectively represent the lowest and highest temperatures considered in the thermo dataset, as loaded at run time. If you set the temperature range directly, it will be bracketed to that of the thermo dataset. For example, the command

```
span -1000 C to 1000 C
```

is functionally the same as

```
span 0 C to 300 C
```

when the “thermo.tdat” dataset is loaded, since the dataset’s range is 0°C to 300°C.

The temperature bounds set here are the same as those set with the “heat\_source” command: The command

```
span 20 C to 100 C
```

may be equivalently expressed

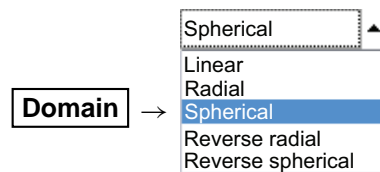
```
heat_source Tmin = 20 C, Tmax = 100 C
```

as long as the heat source option is enabled.

Keyword “off” disables the feature, leaving values for the limits intact, and keyword “on” re-enables the feature. The “reset” keyword clears the limits, disabling the feature. By default, the span feature is disabled.

### 8.101 spherical

```
spherical <r1 = value> <unit> <r2 = value> <unit>  
          <angle = value> <unit> <normal|reverse>  
spherical <off|on>
```



Use the “spherical” command (also: “spike”) to set a spherical flow system extending from radius “r1” to radius “r2”, where “r1” is less than “r2”. Radii may be entered in units of distance (see the **Units Recognized** chapter in the **GWB Reference Manual**); “cm” is the default. Keyword “angle” gives the angle of divergence of the medium, in

“radians” (the default) or “degrees”. By default, this value is 0.1 radians. To return to a linear geometry, type “spherical off”. The “normal” keyword sets the domain to narrow from right to left; the “reverse” keyword sets it narrowing from left to right. See also the “radial” command.

### 8.102 start\_date

```
start_date <value | off>
```

**Config** → **Stepping...**

Use the “start\_date” command to set an explicit starting date of the reaction. This can be used to coordinate the plotting of dated scatter data samples stored in a **GSS** spreadsheet on the reaction path in **Xtplot**. The date should be in the format “MM/DD/YYYY”. Use “off” to return to the default of not set.

```
start_date 10/30/2008
start_date off
```

### 8.103 start\_time

```
start_time <value | off>
```

**Config** → **Stepping...**

Use the “start\_time” command to set an explicit starting time of the reaction. This can be used to coordinate the plotting of timed scatter data samples stored in a **GSS** spreadsheet on the reaction path in **Xtplot**. The time should be in the format “HH:MM:SS”. Use “off” to return to the default of not set.

```
start_time 11:30:00
start_time off
```

### 8.104 step\_increase

```
step_increase = <value | ?>
```

**Config** → **Stepping...**

Use the “step\_increase” command to set the greatest proportional increase, from one step to the next, in the size of the time step. By default, this variable is set to 2.0. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 8.105 step\_max

```
step_max = <value | ?>
```

**Config** → **Stepping...**

Use the “step\_max” command to limit the number of reaction steps the program may take to trace a simulation. Use a “?” to restore the default state, which is no prescribed limit. To see the current setting, type “show variables”.

## 8.106 suffix

```
suffix <string>
```

**Config** → **Output...**

Use the “suffix” command to alter the names of the output datasets (“X1t\_output.txt” and “X1t\_plot.xtp”) by adding a trailing string. Example:

```
suffix _run2
```

produces output datasets with names such as “X1t\_output\_run2.txt”.

## 8.107 sulfur-34

```
sulfur-34 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “sulfur-34” command (also, “34-S”) to set the <sup>34</sup>S isotopic composition of the initial fluid, boundary fluids, reactant species (aqueous species, minerals, end members, gases, or oxides), or segregated minerals. The composition may be set on any scale (e.g., CDT), but you must be consistent throughout the calculation. Example:

```
scope initial
  sulfur-34 fluid = -10
scope inlet-1
  sulfur-34 fluid = +4
scope
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
sulfur-34 remove
sulfur-34 off
```

clear all settings for  $^{34}\text{S}$  isotopes in the current scope.

See also the “<isotope>” section above, and the “carbon-13”, “hydrogen-2”, and “oxygen-18” commands.

## 8.108 suppress

```
suppress <species, minerals, solid solutions, gases, surface species | ALL>
```

**Config** → **Suppress...**

Use the “suppress” command (also: “kill”) to prevent certain aqueous species, surface species, minerals, solid solutions, or gases from being considered in a calculation. Example:

```
suppress H3SiO4- Quartz "Maximum Microcline"
```

prevents the three entries listed from being loaded from the database. Typing “suppress ALL” suppresses all of the minerals and solid solutions in the thermodynamic database, as well as any solid solutions defined locally.

The “unsuppress” command reverses the process. To suppress all but a few minerals, you could type

```
suppress ALL
unsuppress Quartz Muscovite Kaolinite
```

## 8.109 surface\_capacitance

```
surface_capacitance = <value | ?>
```

```
surface_capacitance C1 = <value | ?> C2 = <value | ?>
```

```
surface_capacitance on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant capacitance  ▾ F/m<sup>2</sup>  
or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_capacitance”) to set, in units of F/m<sup>2</sup>, the capacitance or capacitances of a sorbing surface.

For a two-layer complexation model, when you set a capacitance with the command, or if a value for capacitance is set in the header section of the surface reaction dataset,

**X1t** will model surface complexation for the surface in question using the constant capacitance model, rather than the full two-layer model.

A triple-layer or CD-MUSIC complexation model, on the other hand, requires two capacitances,  $C_1$  and  $C_2$ . Default capacitances are set in the header of the surface dataset, but you may use the “C1” and “C2” arguments to override the default settings from the command line.

If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_capacitance on HFO = 2
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface.

## 8.110 surface\_data

```
surface_data <sorption dataset>
surface_data remove <sorption dataset | surface type>
surface_data OFF
```

File → Open → Sorbing Surfaces... →

File → Open → Sorbing Surfaces... → *select surface(s)* →

or Config → Sorbing Surfaces...

Use the “surface\_data” command (abbrev.: “surf\_data”) to specify an input dataset of surface sorption reactions to be considered in the calculation. The dataset name should be enclosed in quotes if it contains any unusual characters. Use the “remove” argument to eliminate a surface dataset, specified by name or surface type (e.g., “HFO”), from consideration. The argument “OFF” disables consideration of all surface complexes.

You can specify more than one sorbing surface in a model by repeating the “surface\_data” command for different datasets (a dataset of surface reactions for sorption onto hydrous ferric oxide, several triple-layer and CD-MUSIC datasets, as well as example datasets for the ion exchange,  $K_d$ , Freundlich, and Langmuir models are distributed with the software). To remove a dataset of surface reactions from consideration, you use commands such as

```
surface_data remove FeOH.sdat
surface_data remove HFO
surface_data OFF
```

The latter command removes all of the surface datasets that have been loaded.

## 8.111 surface\_potential

```
surface_potential = <value | ?>
surface_potential on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant potential  ▼ mV  
 or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_potential”) to set, in units of mV, the electrical potential for a two-layer sorbing surface. When you set this value (or if a value is set in the header section of the surface reaction dataset), **X1t** will model surface complexation for the surface in question using the constant potential method, rather than invoking the full two-layer model.

If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_potential on HFO = 0
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface.

## 8.112 swap

```
swap <new basis> <for> <basis species>
```

**Initial** or  **Fluids** → 

Use the “swap” command to change the set of basis entries. All reactions are written internally in terms of a set of basis species that you can alter to constrain the composition of the initial system. An aqueous species, mineral, gas, or activity ratio can be swapped into the basis in place of one of the original basis species listed in the database. The swap operates on either the initial system, a boundary fluid, or globally on all fluids, depending on the current input scope (see the “scope” command). Examples:

```
swap CO3--          for HCO3-
swap Quartz         for SiO2(aq)
swap CO2(g)         for H+
swap O2(g)          for O2(aq)
swap Ca++/Na+^2    for Ca++
```

Each end member of a continuous solid solution can similarly be swapped into the basis. Use the syntax <solid solution>::<end member tag>. For example, the sodium end member Albite (tag “Ab”) of the plagioclase feldspar solid solution (name “Plag”) can be swapped into the basis simultaneously with the calcium end member Anorthite (tag “An”) as follows:

```
swap Plag::Ab for Na+
swap Plag::An for Al+++
```

The new species must contain in its composition the original basis species being swapped out (you can’t swap lead for gold). For example, CO<sub>2</sub>(g) is composed of HCO<sub>3</sub><sup>-</sup>, H<sup>+</sup>, and water in “thermo.tdat”. The reactions in the thermo dataset (once reduced to the set of basis and redox species and modified to reflect enabled redox couples) show the basis entries for which a species may be swapped. For a list of original basis species, type “show basis”. To reverse a swap, type “unswap <species>”.

### 8.113 system

```
system <system command>
```

Use the “system” command to execute a command on the computer’s operating system (a “DOS command”) without leaving **X1t**. Example:

```
system "copy X1t_output.txt X1t_output2.txt"
```

Enclose multiword system commands in quotes. An alternative way to fork a system command is to begin the **X1t** command with a “\$”:

```
$copy X1t_output.txt X1t_output2.txt
```

In this case, no quotation marks are needed.

When a “system” command is executed, a “Command Prompt” window will appear briefly on your screen. Due to limitations of the Windows operating system, you cannot fork a command that requires user input, and you will not be able to see any output (including error messages) that might be generated by the command.

### 8.114 TDS

```
TDS = <value | ?>
```

```
 Initial or  Fluids →  advanced → TDS  ▼ mg/kg
```



Use the “TDS” command to set in mg/kg the total dissolved solids for the current input scope, if you don’t want the program to calculate this value automatically. The program uses the TDS when needed to convert input constraints into molal units. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command).

To restore automatic calculation of the TDS, type the command with no argument or with an argument of “?”. To see the variable’s current setting, type “show variables”.

## 8.115 temperature

```

temperature = <field variable> <unit>
temperature = <value> <unit>
temperature initial = <field variable> <unit> <boundary fluid> = <value> <unit>
temperature isothermal
temperature constant = <on | off>
temperature reset

```

**Initial** → temperature  +  ▼ C ▼

**Fluids** → temperature  ▼ C ▼

Use the “temperature” command (also: “T”) to set the temperature of the initial system or the boundary fluids. The **Units Recognized** chapter in the **GWB Reference Manual** lists possible units, which default to “C”. By default, the initial system and boundary fluids are set to 25°C and, in the absence of a heat source, the simulation proceeds isothermally.

In its simple usage, the command sets the temperature of the current scope:

```

scope initial
  temperature = 25 C
scope inlet_fluid1
  temperature = 100 C

```

but you can also specify scope as a keyword on the command line

```

temperature initial = 25 C, inlet_fluid1 = 100 C

```

In the example, hot fluid invades an initially cool domain.

Keyword “isothermal” sets the boundary fluid temperature(s) to the initial temperature for the domain. Keyword “constant” causes the program to hold temperature at each nodal block invariant over the course of the simulation, and the “reset” keyword returns temperature variables to their default state.

You should enter temperature values within the range of the thermo dataset in use. For the default dataset “thermo.tdat”, the range is 0°C to 300°C.

## 8.116 thermal\_cond

**thermal\_cond** = <field variable | ?> <unit> <steady | transient>

**Medium** → thermal conductivity   ▼ cal/cm/s/C ▼

Use the “thermal\_cond” command (abbrev.: “tcond”) to set the thermal conductivity of the fluid-saturated medium. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/cm/sec/°C” is the default. This value is used during polythermal simulations in calculating the effects of advective heat transport.

By default, this variable is set to 0.004 cal/cm/s/°C. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show system”.

## 8.117 theta

**theta** = <value | ?>

**Config** → **Stepping...**

Use the “theta” command to set the time weighting variable used in evaluating kinetic rate laws. The value may vary from zero (full weighting at the old time level) to one (full weighting at the new time level). By default, this variable is set to 0.6. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 8.118 threads

**threads** = <value | ?>

**Config** → **Stepping...**

Use the “threads” command to control the number of threads the program spawns when running a simulation. Using more than one thread on multicore or multiprocessor computers, the program runs in parallel, dividing the computing work among the computing cores available. By default (use the “?” argument), the program spawns one thread for each computing core. Typing the command without an argument returns the current setting and the number of computing cores available.

## 8.119 timax

```
timax = <value | ?>
```

**Config** → **Iteration...**

The “timax” command sets in molal units the maximum value of ionic strength used in calculating species’ activity coefficients when Debye-Hückel methods are employed. The limiting value is also used when water activity is calculated according to Wolery (1992b). By default, this variable is set to 3 molal. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 8.120 time

```
time start = <value> <unit> <interval> = <value | ?> <unit> \
end = <value> <unit>
```

**Intervals** →

start	at	<input type="text"/>	▼ day ▼
interval_1	at	<input type="text"/>	▼ day ▼
end	at	<input type="text"/>	▼ day ▼

Use the “time” command (abbrev.: “t”) to set the time span (keywords “start” and “end”) of the simulation, and the start times for individual simulation intervals. The starting time for the run, by default, is zero, and the default end time is 1 day. Unit choices are listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “day” is the default. Examples:

```
time end = 100 years
time start 10 days, end 20 days
time phase2 = 1 year, phase3 = 2 years, end = 10 years
```

Note that starting times and the end time are commonly and perhaps more clearly specified instead on the “intervals” command.

## 8.121 title

```
title <character string>
```

**Config** → **Output...**

Use the “title” command to set a title to be passed to the graphics program. Example:

```
title "Yucca Mountain Groundwater"
```

Be sure to put multiword titles in quotes.

### 8.122 unalter

```
unalter <species | mineral | solid solution | gas | surface species | ALL>
```

```
Config → Alter Log Ks... → select entry(ies) → 
```

Use the “unalter” command to reverse the effect of having changed the log  $K$ ’s for a species, mineral, gas, or surface reaction, the selectivity coefficient for an exchange species, the  $K_d$  for a sorbed species, the  $K_f$  and  $n_f$  for a Freundlich species, or the properties of a solid solution in the current dataset. Example:

```
unalter Quartz
```

In this case, the log  $K$  values for quartz revert to those in the current thermodynamic dataset. The argument “ALL” resets the log  $K$ ’s, selectivity coefficients,  $K_d$ ’s, or  $K_f$ ’s and  $n_f$ ’s for all species, minerals, gases, and surface species, as well as the properties of all solid solutions in the dataset.

### 8.123 unsegregate

```
unsegregate <mineral(s) | ALL>
```

```
Config → Isotopes... → select entry(ies) → 
```

Use the “unsegregate” command to remove minerals from the list of minerals to be segregated isotopically.

### 8.124 unsuppress

```
unsuppress <species, minerals, solid solutions, gases, surface species | ALL>
```

```
Config → Suppress...
```

Use the “unsuppress” command (also: “include”) to include in the calculation aqueous species, surface species, minerals, solid solutions, or gases that have previously been suppressed. Examples:

```

unsuppress Quartz Albite "Albite low"
unsuppress ALL

```

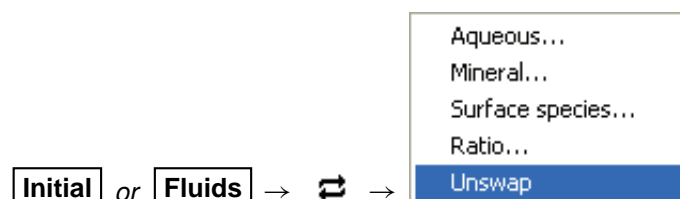
The argument “ALL” clears any species, minerals, solid solutions, or gases that have been suppressed.

## 8.125 unswap

```

unswap <species | ALL>

```



Use the “unswap” command to reverse a basis swap. The swap reversal operates on either the initial system, a boundary fluid, or globally on all fluids, depending on the current input scope (see the “scope” command). Example:

```

unswap Quartz (or unswap SiO2(aq))

```

to reverse the effect of the previous command

```

swap Quartz for SiO2(aq)

```

At this point, SiO<sub>2</sub>(aq) is back in the basis. The “ALL” argument reverses all basis swaps.

## 8.126 usgovt

```

usgovt

```

```

Help → About X1t...

```

Display the restricted rights legend that governs use, duplication, or disclosure by the U.S. Government.

## 8.127 width

**width** = <value> <unit>

**Domain** → width (y)  ▼ cm ▼

Use the “width” command to set the width along  $y$  of the medium. You can use any unit of distance listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cm” is the default. This command has no effect when modeling flow in radial coordinates. Domain width defaults to 1 cm in **X1t**. See also the “length” and “height” commands.

## 8.128 xtplot

**xtplot** <xtplot options>

**Run** → **Xtplot**

The “xtplot” command launches program **Xtplot** from the **X1t** command line, invoking input from the current graphics interface dataset (by default, “X1t\_plot.xtp”; see the “suffix” command). For example,

```
xtplot -graph xyplot
```

starts **Xtplot** initially showing an  $x$ - $y$  plot.

## 8.129 Xstable

**Xstable** = <value>

**Config** → **Stepping...**

Use the “Xstable” command to control how the stability criterion for dispersive transport and thermal conduction is applied. A value of one sets the theoretically largest stable time step for an ideal situation. **X1t** simulations are not necessarily ideal (for example, the solute may react and the medium may not be uniform), so this limiting time step may in fact be too large to be stable. Setting “Xstable” to a value smaller than one results in a more stringent constraint on the time step, and hence greater stability. The default value for this variable is 1.0, the theoretical limit. See also the “Courant” command.

# X2t Commands

## 9.1 <unit>

<field variable> <free> <unit> <as element symbol> <basis entry>

**Initial** or **Fluids** →  ▼ free kg ▼

To constrain chemical composition, enter a command containing only the above entries. Entries may appear in any order. The qualifier “free” specifies that the constraint applies to the free rather than to the bulk basis entry. Settings constrain the composition of the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). **X2t** recognizes the following units:

*By mass or volume:*

mol	mmol	umol	nmol	
kg	g	mg	ug	ng
eq	meq	ueq	neq	
cm3	m3	km3	l	

*By concentration:*

mol/kg	mmol/kg	umol/kg	nmol/kg	
molal	mmolal	umolal	nmolal	
mol/l	mmol/l	umol/l	nmol/l	
g/kg	mg/kg	ug/kg	ng/kg	
wt%	"wt fraction"			
g/l	mg/l	ug/l	ng/l	
eq/kg	meq/kg	ueq/kg	neq/kg	
eq/l	meq/l	ueq/l	neq/l	

*By carbonate alkalinity (applied to bicarbonate component):*

eq_acid	meq_acid	ueq_acid	neq_acid	
eq_acid/kg	meq_acid/kg	ueq_acid/kg	neq_acid/kg	
eq_acid/l	meq_acid/l	ueq_acid/l	neq_acid/l	
g/kg_as_CaCO3	mg/kg_as_CaCO3	ug/kg_as_CaCO3	ng/kg_as_CaCO3	
wt%_as_CaCO3				
g/l_as_CaCO3	mg/l_as_CaCO3	ug/l_as_CaCO3	ng/l_as_CaCO3	
mol/kg_as_Ca...	mmol/kg_as_Ca...	umol/kg_as_Ca...	nmol/kg_as_CaCO3	
mol/l_as_CaCO3	mmol/l_as_CaCO3	umol/l_as_CaCO3	nmol/l_as_CaCO3	

## GWB Command Reference

### *Per volume of the system:*

mol/cm <sup>3</sup>	mmol/cm <sup>3</sup>	umol/cm <sup>3</sup>	nmol/cm <sup>3</sup>
kg/cm <sup>3</sup>	g/cm <sup>3</sup>	mg/cm <sup>3</sup>	ug/cm <sup>3</sup>
ng/cm <sup>3</sup>			
mol/m <sup>3</sup>	mmol/m <sup>3</sup>	umol/m <sup>3</sup>	nmol/m <sup>3</sup>
kg/m <sup>3</sup>	g/m <sup>3</sup>	mg/m <sup>3</sup>	ug/m <sup>3</sup>
ng/m <sup>3</sup>			
volume%	"vol. fract."		

### *By activity:*

activity	fugacity	ratio
pH	V	pe

### *By partial pressure:*

Pa	MPa	atm	bar
psi			

Activity and fugacity may be abbreviated to "a" or "f". Keyword "total" reverses a setting of "free".

Use the "as" keyword to constrain mass in terms of elemental equivalents. For example, the command

```
CH3COO- = 10 umol/kg as C
```

specifies 5 umol/kg of acetate ion, since each acetate contains two carbons, whereas

```
20 mg/kg SO4-- as S
```

would specify 59.9 mg/kg of sulfate, since the ion's mole weight is about 3 times that of sulfur itself.

Examples:

```
55 mmol/kg HCO3-  
Na+ = 1 molal  
1 ug/kg U++++  
10 volume% Dolomite  
pH = 8  
Eh = .550 V  
log f O2(g) = -60  
HCO3- = 30 mg/kg as C
```



You may in a similar fashion constrain the concentration of a kinetic aqueous or surface complex in the initial system, a boundary fluid, or globally. For example, the commands

```
kinetic AIF++
AIF++ = 1 umol/kg
scope ambient_water
  AIF++ = 10 umol/kg
```

set the concentration of the kinetic complex AIF<sup>++</sup> to 1  $\mu\text{mol kg}^{-1}$  in all fluids, except fluid “ambient\_water”, where its concentration is ten times greater.

## 9.2 <isotope>

```
<isotope | symbol> <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the name of any isotope system loaded in the isotope dataset (also, the isotope’s symbol) to set the isotopic composition of the initial fluid, boundary fluids, reactant species (aqueous species, minerals, end members, gases, or oxides), or segregated minerals. The composition may be set on any scale (e.g., SMOW, PDB, ...), but you must be consistent throughout the calculation.

For example, if the <sup>17</sup>O isotope system has been added to the isotope dataset, you could enter:

```
scope initial
  oxygen-17 fluid = -10
scope inlet-1
  oxygen-17 fluid = +2
scope
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
oxygen-17 remove
oxygen-17 off
```

clear all settings for <sup>17</sup>O isotopes from the calculation.

## 9.3 activity

**activity** <species> = <field variable>

or  →  ▼ activity ▼

Use the “activity” command (abbrev.: “a”) to constrain the activity of an aqueous species. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). Examples:

```
activity Na+ = 0.3
log a H+ = -5
```

See also the “pH”, “Eh”, “pe”, “ratio”, “fugacity”, “fix”, and “slide” commands.

## 9.4 add

**add** <basis species>

or  →

Use the “add” command to include a basis species in the calculation. Example:

```
add HCO3-
```

See also the “swap”, “activity”, “fugacity”, “pH”, “pe”, and “Eh” commands.

## 9.5 alkalinity

**alkalinity** = <field variable> <unit>

or  → HCO3-  ▼ mg/kg\_as\_CaCO3 ▼

Use the “alkalinity” command to constrain the total concentration of  $\text{HCO}_3^-$  to reflect the solution’s carbonate alkalinity. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “mg/kg\_as\_CaCO3” is the default. To use this option, the solution pH must be set explicitly.

## 9.6 alter

```

alter <species | mineral | gas> <log Ks>
alter <species | mineral | gas> <poly coefs> TminK = <value> TmaxK = <value>
alter <surface species> <poly coefs> TminK = <value> TmaxK = <value>
alter <surface species> logK = <value> dlogK/dT = <value>
alter <exchange species> beta = <value>
alter <sorbed species> Kd = <value>
alter <sorbed species> Kf = <value> nf = <value>
alter <solid solution> <type> <discrete | continuous> \
  <from <value> to <value> step <value>> <<variable> = <value>>

Config → Alter Log Ks...
Config → Solid Solutions...

```

Use the “alter” command to change the temperature expansion for a species’, mineral’s, or gas’ log  $K$ , to adjust the stability of a surface species, or to change the properties of a solid solution.

Temperature expansions for log  $K$ s are given in the thermo dataset either as  $T$ -tables, or by up to six coefficients of a polynomial. When the current thermo dataset uses  $T$ -tables, you list replacement values at each of the principal temperatures specified in the dataset, most commonly 0°C, 25°C, 60°C, 100°C, 150°C, 200°C, 250°C, and 300°C. Example:

```
alter Anhydrite -4.3009 -4.4199 -4.7126 -5.1758 -6.2299 500 500 500
```

Values of “500” represent a lack of data at the corresponding temperature.

For a thermo dataset constructed from polynomial expansions, on the other hand, list instead up to six polynomial coefficients

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4
```

You can optionally append a temperature range of validity, in Kelvins, for the polynomial:

```
alter Anhydrite 4186 2.475 -0.001305 -85377 0 -794.4 TminK= 293 TmaxK= 383
```

Absent a range, the polynomial is taken to span the principal temperatures. You may adjust a reaction’s temperature range without specifying polynomial coefficients, but setting only coefficients resets the temperature range.

For a surface complex, use the “alter” command to set log  $K$ , its temperature derivative, or both

```
alter >(w)FeOCa+ logK = 6.0 dlogK/dT = 0.02
```

whereas for an ion exchange reaction, you set the selectivity coefficient  $\beta$

```
alter >X2:Ca beta = .033
```

directly, rather than as a logarithm. For a sorbing species, set  $K_d$  or, for a Freundlich species,  $K_f$ ,  $n_f$ , or both, as follows

```
alter >Pb++ Kd = .03  
alter >Sr++ Kf = .015 nf = .8
```

For a solid solution, use the “alter” command to set the activity coefficient model, discrete or continuous behavior, the composition range in terms of the mole fraction of the most recently specified end member, and parameters for the chosen activity model, as described for the “solid\_solution” command. A solid solution “my\_ss” might be set as a subregular Guggenheim solution with the command

```
alter my_ss guggenheim a0 = 1 a1 = 2
```

You may prefer to use the “alter” command to adjust an entry from the thermo database, and the “solid\_solution” command to reconfigure a user-defined solution, but the two commands work interchangeably for this purpose.

Type “show alter” to list altered species or solutions and their settings; the “unalter” command reverses the process.

## 9.7 audit

```
audit <off>
```

**Config** → **Stepping...**

The program performs a global mass balance audit over the course of each simulation. By default, the program is silent unless it detects a mass imbalance, in which case it writes a warning message to the **Results** pane, as well as the bottom of the print-format output, if the print output stream is open.

You can use the “audit” option to cause the program to report at the end of the simulation the complete results of the mass balance audit for each original basis entry.

To turn audit reporting on and off, use:

```
audit  
audit off
```

## 9.8 b-dot

**b-dot**

**File** → **Open** → **Thermo Data...** → thermo.tdat

The “b-dot” command (formerly: “debye-huckel”) causes the program to read the “thermo.tdat” thermodynamic dataset, which invokes the “B-dot” form of the extended Debye-Hückel equation to calculate activity coefficients for aqueous species.

## 9.9 balance

**balance** <on> <basis entry>

**Initial** or **Fluids** →  ▼ unit ▼ → Balance species

Use the “balance” command to specify the basis entry to be used for electrical charge balancing. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). The basis entry must be a charged aqueous species. By default, **X2t** balances on  $\text{Cl}^-$ . Example:

balance on Na+

## 9.10 bottom

**bottom fluid** = <boundary fluid>

**bottom** <interval> **fluid** = <boundary fluid>

**bottom** = <normal | inlet | outlet>

**Flow** → boundaries: bottom  ▼

In simulations in which the flow field is imported, and only such simulations, use the “bottom” command to control the boundary condition at the bottom side of the domain: the fluid to appear at the boundary, and whether solutes can diffuse and disperse, and heat conduct, across the boundary.

If the command contains the name of a simulation interval, subsequent settings on the command apply to that interval, rather than the simulation as a whole. This behavior continues to the end of the command, or until the name of another interval is encountered.

You use the “fluid” keyword to specify the boundary fluid to appear on the bottom boundary, for the entire simulation or an individual interval. For example, the commands

```
bottom fluid = ambient_water  
bottom phase2 fluid = pit_water
```

set fluid “ambient\_water” on the bottom boundary over the course of the simulation, except during interval “phase2”, in which case fluid “pit\_water” is found on the boundary.

The program by default uses “inlet/free outlet” boundary conditions: where fluid flows into the domain, solute mass can advect, diffuse, and disperse across the boundary, but where it exits the domain, transport occurs by advection alone. Similarly, heat can conduct across an inlet, but not an outlet.

You can force the program to treat the bottom of the domain as an inlet or free outlet boundary, regardless of the direction of fluid migration. Keyword “inlet” causes the program to allow diffusion, hydrodynamic dispersion and heat conduction across the bottom bound, whereas keyword “outlet” creates an insulated boundary sealed against transport by these processes:

```
bottom outlet
```

In either case, the advective transport of solute mass and heat depends on the direction and rate of fluid migration, and is not affected by the command setting

See also the “left”, “top”, “right”, “discharge”, “head drop”, and “interval” commands.

### 9.11 carbon-13

```
carbon-13 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “carbon-13” command (also, “13-C”) to set the <sup>13</sup>C isotopic composition of the initial fluid, boundary fluids, reactant species (aqueous species, minerals, end members, gases, or oxides), or segregated minerals. The composition may be set on any scale (e.g., PDB), but you must be consistent throughout the calculation. Example:

```
scope initial  
  carbon-13 fluid = -10  
scope inlet-1  
  carbon-13 fluid = +4  
scope
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
carbon-13 remove  
carbon-13 off
```

clear all settings for  $^{13}\text{C}$  isotopes in the current scope.

See also the “<isotope>” section above, and the “hydrogen-2”, “oxygen-18”, and “sulfur-34” commands.

## 9.12 chdir

```
chdir <directory> <ALL>
```

**File** → **Working Directory...**

Use the “chdir” command (abbrev.: “work\_dir”, “cd”) to change the working directory. The program reads input scripts relative to the current working directory and writes output into it. Typing the command “chdir” without an argument causes the program to display the name of the working directory. The “ALL” argument changes the working directories of all of the GWB apps. The command

```
chdir ~
```

changes to the user’s home directory, if one is defined by the operating system.

## 9.13 clear

```
clear
```

**View** → **Clear Screen**

Use the “clear” command (abbrev.: “c”) or right-click in the text window and choose **Clear Screen** to clear it.

## 9.14 clipboard

```
clipboard
```

**Edit** → **Copy**

The “clipboard” command causes the contents of the **Results** pane to be copied to the clipboard, from where it can be pasted into various types of documents.

## 9.15 conductivity

**conductivity** <conductivity dataset>

**File** → **Open** → **Conductivity Data...**

Use the “conductivity” command to change the input file of coefficients used to calculate electrical conductivity. Example:

```
conductivity "..\my_conductivity.dat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. Beginning with GWB11, the applications compute electrical conductivity using either of two different approaches, the USGS and APHA methods; the USGS method is the default. The required coefficients are defined in the files “conductivity-USGS.dat” and “conductivity-APHA.dat”, respectively, which are installed in the same directory as the thermo datasets (commonly “\Program Files\GWB\Gtdata”).

## 9.16 couple

**couple** <redox species | element(s) | **ALL**>

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “couple” command to enable any redox coupling reactions that have been disabled with the “decouple” command. You specify one or more redox species or elements. For example, the command

```
couple Carbon
```

couple all redox reactions involving the element carbon. Argument “ALL” enables all of the coupling reactions in the thermo dataset.

## 9.17 Courant

**Courant** = <value | ?>

**Config** → **Stepping...**

Use the “Courant” command to constrain the time step according to the Courant condition. You enter a value for the Courant number, which is the ratio of the distance fluid travels over a time step to the length of the nodal blocks. If you set a Courant number of one, then **X2t** will select a time step over which the fluid will exactly traverse the nodal blocks. For a value of 0.5, the fluid will move halfway across the nodal



blocks, and so on. Values greater than one for the Courant number typically give unstable solutions and are therefore not recommended. By default, **X2t** assumes a Courant number of 1.0. The “?” argument resets the default value.

## 9.18 cpr

**cpr** = <field variable | ?> <unit> <steady | transient>

**Medium** → cpr (minerals)  +  cal/g/C ▼

Use the “cpr” command to set the heat capacity of the rock (mineral) framework. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/g°C” is the default. This value is used during polythermal simulations in calculating the effects of advective heat transport. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function.

By default, this variable is set to 0.2 cal/g°C. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 9.19 cpu\_max

**cpu\_max** = <value | ?>

**Config** → **Stepping...**

Use the “cpu\_max” command to limit the amount of computing time a simulation may take. You set the maximum computing time in seconds, or use a “?” to restore the default state, which is no prescribed limit. To see the current setting, type “show variables”.

## 9.20 cpw

**cpw** = <field variable | ?> <unit> <steady | transient>

**Medium** → cpw (fluid)  +  cal/g/C ▼

Use the “cpw” command to set in cal/g°C the heat capacity of the fluid. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/g°C” is the default. This value is used during polythermal simulations in calculating the effects of advective heat transport. The “transient” keyword causes the

model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function.

By default, this variable is set to 1.0 cal/g°C. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

### 9.21 data

```
data <thermo dataset> <verify>
```

**File** → **Open** → **Thermo Data...**

Use the “data” command to change the input file of thermodynamic data. Example:

```
data "..\my_thermo.tdat"
```

The dataset name may need to be enclosed in quotes if it contains unusual characters. The “verify” option causes the program to read the named dataset only if it has not already been read.

### 9.22 decouple

```
decouple <redox species | element(s) | ALL>
```

**Config** → **Redox Couples...** → *select couple(s)* → **select with...**

Use the “decouple” command to disable the coupling reactions for one or more redox species, in order to calculate a model assuming redox disequilibrium. The redox species then become available for use as basis species and may be constrained independently of the original basis entries. You can disable as many coupling reactions as you want.

You specify either one or more redox species or elements. For example, the command

```
decouple Carbon
```

decouples all redox reactions involving the element carbon. Argument “ALL” disables all of the coupling reactions in the thermo dataset. Use the “couple” command to enable coupling reactions, once they have been disabled.

## 9.23 delQ

**delQ** = <value | ?>

**Config** → **Stepping...**

Use the “delQ” command to control the lengths of time steps taken in a simulation accounting for reaction kinetics. The program limits how much the ion activity product  $Q$  can change over a step, for each kinetic reaction considered. The setting for “delQ” is the projected change  $\Delta Q/Q$  allowed in the relative value of the activity product. You can set a larger value to permit longer time steps, or a smaller value to improve stability. The default setting is 0.1. Type the command with no argument or with an argument of “?” to restore the default. To see the current setting, type “show variables”.

## 9.24 deltar

**deltar** = <field variable> <unit>

**Domain** →  delta r  +  ▼ cm ▼

The “deltar” (also: “delta\_r”) command is a synonym for the “deltax” command, described below. It is provided for clarity when working with radial domains.

## 9.25 deltax

**deltax** = <field variable> <unit>

**Domain** →  delta x  +  ▼ cm ▼

The “deltax” (also: “delta\_x”) command sets the length  $\Delta x$  of nodal blocks in the finite difference grid. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cm” is the default.

When you set a value for  $\Delta x$ , the program ignores any setting of the “length” command and figures the domain length from the values of  $\Delta x$  and  $N_x$ , the number of nodal blocks specified along  $x$ . For example, the commands

$N_x = 10$   
deltax = 1 m

are equivalent to

$N_x = 10$   
Length = 10 m

You can set  $\Delta x$  as a field variable, in order to create grids of varying node density. For example, the commands

```
Nx = 10
deltax = { 1 1 .5 .5 .25 .25 .5 .5 1 1 } m
```

set a grid in which nodal blocks near the center of the domain are smaller than those near the left and right sides. The domain in this case will be 7 m long. See also the “deltay”, “length”, and “Nx” commands.

### 9.26 deltay

```
deltay = <field variable> <unit>
```

```
Domain →  delta y  +  ▼ cm ▼
```

The “deltay” (also: “delta\_y”) command sets the width  $\Delta y$  of nodal blocks in the finite difference grid. You can specify one of the units listed in the **Units Recognized** chapter; **GWB Reference Manual** “cm” is the default.

When you set a value for  $\Delta y$ , the program ignores any setting of the “width” command and figures the domain length from the values of  $\Delta y$  and  $N_y$ , the number of nodal blocks specified along  $y$ . For example, the commands

```
Ny = 10
deltay = 1 m
```

are equivalent to

```
Ny = 10
width = 10 m
```

You can set  $\Delta y$  as a field variable, in order to create grids of varying node density. For example, the commands

```
Ny = 10
deltay = { 1 | 1 | .5 | .5 | .25 .25 | .5 | .5 | 1 | 1 } m
```

set a grid in which nodal blocks near the center of the domain are smaller than those near the bottom and top. (Notice the use of vertical bars to separate entries in the field variable, since the  $\Delta y$  vector runs vertically.) The domain in this case will be 7 m wide. See also the “deltax”, “width”, and “Ny” commands.

## 9.27 delxi

**delxi** = <value | ?>

**Config** → **Stepping...**

Use the “delxi” command to set the maximum length (in terms of reaction progress, which varies from zero to one over the course of the simulation) of the time step. By default, this variable is set to 0.01. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 9.28 density

**density** = <value | ?>

**density** = <TDS | chlorinity>

**density** = <batzle-wang | phillips>

**Initial** →  advanced → density  ▼ g/cm<sup>3</sup>

**Fluids** →  advanced → density  ▼ g/cm<sup>3</sup>

**Config** → **Options...**

You can use the “density” command to set in g/cm<sup>3</sup> the fluid density the program uses to convert compositional constraints to molality, the concentration unit it carries internally. If you set the initial Na<sup>+</sup> composition in mg/l, for example, the program needs to know the density of the initial fluid to determine Na<sup>+</sup> molality. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command).

The program by default converts units using a density value it calculates automatically, as discussed below. This value is sufficient for most purposes, and hence it is generally not necessary to set fluid density explicitly. You might, however, want to set the density if you are working at high temperature, but your analysis is expressed per liter of solution at room temperature.

You can also use the “density” command to tell the program how to calculate the default density it uses to convert units, and the fluid density it reports in the simulation results. The program by default uses the Batzle-Wang equation to figure density, but you can use the command “density = phillips” to select the method of Phillips et al., instead.

As well, the program normally figures density as that of an NaCl solution with the same TDS as the fluid in question, at the temperature of interest. With the command “density = chlorinity” you can tell the program to instead use the density of an NaCl solution of equivalent chlorinity.

To restore automatic calculation, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 9.29 diffusion\_coef

**diffusion\_coef** = <field variable | ?> <unit> <steady | transient>

**Medium** → diffusion coefficient

Use the “diffusion\_coef” command (abbrev.: “diff\_coef”) to set a value for the diffusion coefficient in one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function. By default, **X2t** assumes a diffusion coefficient of  $10^{-6}$  cm<sup>2</sup>/s. The “?” argument resets the default value.

## 9.30 discharge

**discharge** = <value | open | closed | ?> <unit>  
**discharge** <interval> = <value | open | closed | ?> <unit>  
**discharge** <left | right> = <value | open | closed | ?> <unit>  
**discharge** <left | right> <interval> = <value | open | closed | ?> <unit>

**discharge fluid** = <boundary fluid>  
**discharge** <interval> **fluid** = <boundary fluid>  
**discharge** <left | right> **fluid** = <boundary fluid>  
**discharge** <left | right> <interval> **fluid** = <boundary fluid>

**discharge x** = <field variable> <unit> **y** = <field variable> <unit>  
**discharge x** <interval> = <field variable> <unit>  
**discharge y** <interval> = <field variable> <unit>  
**discharge** <bottom | top> **fluid** = <boundary fluid>  
**discharge** <bottom | top> <interval> **fluid** = <boundary fluid>

**Flow** →  interval 1:  discharge

Use the “discharge” command to set a value for the specific discharge (fluid volume crossing a unit area per unit time) across the domain’s left or right boundary, or both, and to set the fluid to appear at the boundary or boundaries. Alternatively, you can use this command to import a groundwater flow field from another program.

You use the “left” or “right” keyword to specify the boundary in question. If neither side is specified, the command sets flow across both boundaries, which can not uncommonly lead to contradictory boundary conditions. For example, there is no flow field that can honor boundary conditions in which the right and left discharges are the equal, if there is an injecting or producing well within the domain. For this reason, you typically set discharge across the left or the right boundary, but not both.

Keyword “closed” sets a zero flux across the boundary or boundaries in question, and “open” returns the boundary or boundaries to the default state.

If the “discharge” command contains the name of a simulation interval, subsequent settings on the command apply to that interval, rather than the simulation as a whole. This behavior continues to the end of the command, or until the name of another interval is encountered.

By default, both boundaries are held open to flow and the discharge across them is determined from the permeability of the domain, the fluid viscosity, the driving force for flow (set using the “head\_drop” command), and the rates at which fluid is produced from or injected into any wells. A value of “?” returns the setting for the boundary in question, or both boundaries, to the default condition.

When, on the other hand, you specify the discharge across one or, less commonly, both of the boundaries, the program uses this value (or values) together with permeability, viscosity, and the well production and injection rates to determine the flow pattern.

A value for discharge may be set in any of the units of velocity or specific discharge listed in the **Units Recognized** chapter in the **GWB Reference Manual**. The default is cm/s, which is equivalent to  $\text{cm}^3/\text{cm}^2/\text{s}$ . In addition, you may use the unit “pore\_volumes” (or “pv”) to specify discharge in terms of the number of times the original pore volume is to be replaced over the course of the reaction model.

Examples:

```
discharge left = 15 m/day
discharge right = -100 pore_volumes
discharge right = ?
discharge right start = 10 m/yr interval-2 = 20 m/yr
```

A positive value represents flow from left to right, and a negative value denotes flow in the opposite direction.

You can also specify the boundary fluid to appear on either or both the left and right boundaries, for the entire simulation or an individual interval. To do so, you use the “fluid” keyword. In the examples

```
discharge left = .002 m/s fluid = ambient_water
discharge left phase2 fluid = pit_water
```

fluid “ambient\_water” flows across the left boundary at a discharge of  $.002 \text{ m s}^{-1}$  over the course of the simulation, except during interval “phase2”, in which case fluid “pit\_water” flows into the domain. In a simulation employing an imported flow field, and only such a simulation, you may as well set the fluids to appear at the bottom and top of the domain.

Alternatively, you can use this command together with the “x” and “y” keywords to import a flow field from another program. To do so, prepare a table of the specific discharge along  $x$  from node to node in the domain, and a second table containing similar values for flow along  $y$ . For information on preparing tables, see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**.

When importing a flow field, the  $x$ -direction table should contain  $N_x+1$  columns and  $N_y$  rows of values. The first table column holds values for flow across the left boundary into or out of the first column of nodes. The second table column represents flow from the first column of nodes to the second, and so on, until the last column, which holds values for flow across the right boundary. Similarly, the  $y$ -direction table has  $N_x$  columns and  $N_y+1$  rows, the first row representing flow across the bottom bound and the last for flow across the top.

You can use the “modflow” command to import flow fields directly from the results of a MODFLOW model. In this case, settings made using the “discharge” and “head\_drop” commands are ignored. For more information, see the **Importing From MODFLOW** appendix to the **GWB Reactive Transport Modeling Guide**.

See also the “left”, “right”, and “head\_drop” commands.

### 9.31 dispersivity

```
dispersivity long = <field variable> | ?> <unit> <steady | transient> \
trans = <field variable> | ?> <unit> <steady | transient>
dispersivity = <field variable> | ?> <unit>
```

**Medium** → longitudinal dispersivity  +  ▼ cm ▼  
transverse dispersivity  +  ▼ cm ▼

Use the “dispersivity” command to set values for the longitudinal and transverse components of the hydrodynamic dispersivity. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cm” is the default. Example:

```
dispersivity long = 100 cm, trans = 10 cm
```

Note that the longitudinal component is greater than or equal to the transverse component. By default, **X2t** assumes a value equal to 1% of the length of the domain (see the “length” command) for the longitudinal component; the transverse component is set to 10% of the longitudinal value. The second form of the command sets both components to a single value. The “?” argument resets the default state.



## 9.32 dual\_porosity

```

dual_porosity = <on | off> <spheres | blocks | fractures> \
  <geometry = <spheres | blocks | fractures | ?>> \
  <volfrac = <field variable | ?>> \
  <Nsubnode = <value | ?>> \
  <<radius | half-width> = <field variable | ?> <unit>> \
  <diff_length = <field variable | ?> <unit>> \
  <porosity = <field variable | ?>> \
  <retardation = <field variable | ?>> <steady | transient> \
  <diff_coef = <field variable | ?> <unit>> <steady | transient> \
  <thermal_con = <field variable | ?> <unit>> <steady | transient> \
  <theta = <value | ?>> <reset | ?>

```

**Config** → **Dual Porosity...**

Use the “dual\_porosity” command (abbrev: “dual”) to configure stagnant zones in the simulation, using the dual porosity feature. Enable and disable the feature with the “on” and “off” keywords. Disabling the feature does not affect other settings, so re-enabling the feature returns the model to its most recent configuration.

With the “geometry” keyword, you configure the stagnant zone into spheres, blocks, or a fractured domain, the latter being slabs separated by fractures arrayed along the  $x$  direction. Alternatively, you can set the three configurations directly with keywords “spheres”, “blocks”, and “fractures”. Keyword “volfrac” sets the fraction of the nodal block’s bulk volume occupied by the stagnant zone.

The “Nsubnode” (or “nx”) keyword sets the number of nodes into which the stagnant zone within each node will be divided when solving for solute and temperature distributions. The “radius” (or “half-width”, for blocks and fractures) keyword sets the zone’s characteristic dimension, in units of distance (see the **Units Recognized** chapter in the **GWB Reference Manual**; default is cm), and keyword “diff\_length” sets the distance (same units) from the contact with the free-flowing zone over which the model will account for solute diffusion and heat conduction. Use keywords “porosity”, “retardation”, “diff\_coef”, and “thermal\_con” to set values for the porosity, retardation factor, diffusion coefficient, and thermal conductivity of the stagnant zone. The default unit for the diffusion coefficient is  $\text{cm}^2/\text{s}$ , and thermal conductivity defaults to  $\text{cal}/\text{cm}/\text{s}/^\circ\text{C}$  units; see the **Units Recognized** chapter for a list of options.

By default, the stagnant zone is configured in spheres divided into 5 subnodes. You must specify a value for the volume fraction of the stagnant zone, as well as one for the radius (or half-width); the diffusion length defaults to the latter value. The program uses whatever values are set for the free-flowing zone in the node in question as default values for the porosity, diffusion coefficient, and thermal conductivity of the stagnant zone; the retardation factor defaults to a value of one.

The “theta” keyword sets time weighting ( $0 \leq \theta \leq 1$ ) for the numerical solution of diffusive transport within the stagnant zone. A weight  $\theta = 0$  assigns the explicit method, and larger values invoke an implicit solution. By default, the program chooses  $\theta$

automatically, using the explicit method unless it would force too many more time steps than would otherwise be necessary. In that case, the program uses the implicit method ( $\theta = 0.6$ ), which requires more computing effort per time step, but can take long steps without becoming numerically unstable.

You can append the “transient” keyword when setting several of the parameters: the diffusion coefficient, thermal conductivity, and retardation factor. If the variable is defined by an equation, script, or external function, it will then be re-evaluated continuously over the course of the run.

As an example, the command

```
dual_porosity geometry = spheres, radius = 50 cm, volfrac = 75%
```

configures the stagnant zone into spheres of half-meter radius that occupy three-quarters of the domain. The command

```
dual_porosity reset
```

enables the feature after restoring default settings for each keyword.

### 9.33 dx\_init

```
dx_init = <value | ?>
```

**Config** → **Stepping...**

Use the “dx\_init” command to set the length of the initial time step. You set this value in terms of reaction progress, which varies from zero to one over the course of the simulation.

By default, the variable is ignored. To restore the default state, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

### 9.34 dxplot

```
dxplot = <value | ?> <linear | log> <exact | approx>
```

**Config** → **Output...**

Use the “dxplot” command to set the interval in reaction progress (which varies from zero to one over the course of the simulation) between entries in the “X2t\_plot.xtp” dataset, and to specify the entries be spaced on a linear or logarithmic scale. A value of zero causes the program to write the results after each time step, which can produce unwieldy amounts of output.

The command

```
dxplot = .1 linear
```

for example, causes the program to write entries for when reaction progress is .1, .2, .3, ..., and 1.0. Alternatively, the commands

```
dx_init = .001
dxplot = .5 log
```

produce entries at .001, .003, .01, .03, .1, .3, and 1.0 (see the “dx\_init” command).

The arguments “exact” and “approx” define whether the program will constrain the sizes of time steps to give entries that match the specified increment exactly (the default case), or will output entries whenever the time stepping meets or exceeds the increment.

By default, this variable is set to 0.02, “linear”, and “exact”. To restore the default settings, type the command with no argument or with an argument of “?”. To see the current settings, type “show variables”.

## 9.35 dxprint

```
dxprint = <value | ?> <linear | log> <exact | approx>
```

**Config** → **Output...**

Use the “dxprint” command to set the interval in reaction progress (which varies from zero to one over the course of the simulation) between entries in the “X2t\_output.txt” dataset, and to specify the entries be spaced on a linear or logarithmic scale. A value of zero causes the program to write the results after each time step, which can produce unwieldy amounts of output.

The command

```
dxprint = .1 linear
```

for example, causes the program to write entries for when reaction progress is .1, .2, .3, ..., and 1.0. Alternatively, the commands

```
dx_init = .001
dxprint = .5 log
```

produce entries at .001, .003, .01, .03, .1, .3, and 1.0 (see the “dx\_init” command).

The arguments “exact” and “approx” define whether the program will constrain the sizes of time steps to give entries that match the specified increment exactly (the default case), or will output entries whenever the time stepping meets or exceeds the increment.

By default, this variable is set to 0.1, “linear”, and “exact”. To restore the default settings, type the command with no argument or with an argument of “?”. To see the current settings, type “show variables”.

### 9.36 Eh

**Eh** = <field variable> <unit>

**Initial** or **Fluids** →  ▼ V ▼

Use the “Eh” command to set Eh in the initial system. Example:

Eh = 200 mV

sets the system’s oxidation state to correspond to an Eh of 0.2 volt. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). See also the “activity”, “pH”, “pe”, “fugacity”, “fix”, and “slide” commands.

### 9.37 epsilon

**epsilon** = <value | ?>

**Config** → **Iteration...**

Use the “epsilon” command to set the convergence criterion (dimensionless) for iterating to a solution of the equations representing the distribution of chemical mass. By default, this variable is set to  $5 \times 10^{-11}$ . To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

### 9.38 exchange\_capacity

**exchange\_capacity** = <field variable | ?> <units>

**exchange\_capacity on** <type> = <field variable | ?> <units>

**Config** → **Sorbing Surfaces...** → exchange capacity  ▼ eq ▼

or **File** → **Open** → **Sorbing Surfaces...**

Use the “exchange\_capacity” command (abbrev.: “ex\_capacity” or “exch\_capacity”) to set the exchange capacity (i.e., the CEC) of the system when modeling ion exchange reactions or sorption according to Langmuir isotherms. For ion exchange reactions, you set units of electrical equivalents (“eq”, “meq”, and so on) or equivalents per gram

of dry sediment (“eq/g”, “meq/g”, . . .). For Langmuir reactions, you similarly set a value in mole units: “mol”, “mmol”, “mol/g”, “mmol/g”. If you set units per gram of sediment, the program multiplies the value entered by the mass of rock in the system (including equilibrium and kinetic minerals as well as inert volume) to get the system’s capacity.

If you read in a surface that sorbs by ion exchange or Langmuir isotherms, you must set a value for its exchange capacity. If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example

```
exchange_capacity on IonEx = .0008 eq/g
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the surface represented by the sample dataset “IonEx.sdat”, for example, is “IonEx”. You can use the “show” command to display the “type” of each active surface. See also the “surface\_data” and “inert” commands.

### 9.39 explain

```
explain <species | mineral(s) | solid solution(s) | gas(es) | surface species>
```

**Config** → **Show...** → *select species*

Use the “explain” command to get more information (such as the mole weight of a species, a mineral’s formula, mole volume, and density, or a solid solution’s end-member minerals) about species, minerals, solid solutions, and gases in the dataset. Example:

```
explain Analcime
```

### 9.40 explain\_step

```
explain_step <off>
```

**Config** → **Stepping...**

The “explain\_step” option causes the program to report on the Results pane the factor controlling the length of each time step, whenever the step size is limited by the need to maintain numerical stability.

## 9.41 extrapolate

**extrapolate** <on | off>

**Config** → **Options...**

Use the “extrapolate” option to cause the program to extrapolate log  $K$ s for reactions forming aqueous species, minerals, and gases to temperatures beyond the data’s known range of validity. The option should be used with considerable care.

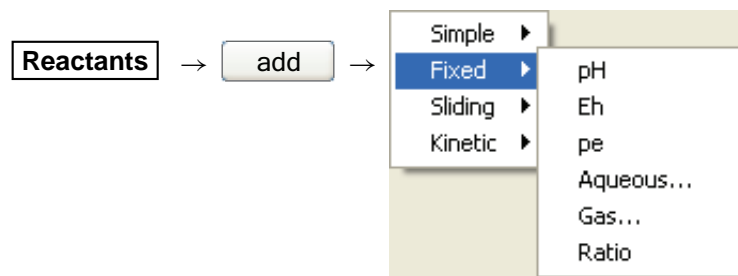
The temperature range of validity for a species’ log  $K$  is defined by its temperature expansion, taken from the thermo dataset. For  $T$ -table expansions, it is the span of principal temperatures at which log  $K$ s are not 500, whereas for polynomial expansions, the range is listed explicitly.

In normal operation, the program loads only species with ranges of validity encompassing the calculation’s temperature range. When the “extrapolate” option is “on”, in contrast, the program projects to the temperature of interest the log  $K$ s of species that would not otherwise be loaded.

Temperature ranges of validity can also be set in the thermo dataset for the virial coefficients used to calculate “Pitzer” and SIT activity coefficients. The “extrapolate on” option works in a similar manner in such cases, allowing virial coefficients to be used at temperatures beyond the coefficients’ known ranges of validity.

## 9.42 fix

**fix** <unit> <species | gas>



Use the “fix” command to hold the activity of a species, fugacity of a gas, or an activity ratio constant over the course of a run. The <unit> can be “activity” or “fugacity” (“a” or “f” for short), “ratio”, “pH”, “pe”, or “Eh”, or it can be omitted. Examples:

```
fix pH
fix a H+
fix f O2(g)
fix ratio Ca++/Na+^2
```

## 9.43 fugacity

**fugacity** <gas> = <field variable>

**Initial** or **Fluids** →  ▼ fugacity ▼

Use the “fugacity” command (abbrev.: “f”) to set gas fugacities (on an atm scale) in the initial system. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). Examples:

```
fugacity O2(g) = .2
f CO2(g) = 0.0003
log f S2(g) = -30
```

See also the “activity”, “ratio”, “pH”, “Eh”, “pe”, “fix”, and “slide” commands.

## 9.44 go

**go** <boundary | initial | single | ask>

**Run** → **Go**

**Run** → **Go Initial**

**Run** → **Go Single**

**Results** →

Use the “go” command to initiate calculation of a model based on the current configuration. Used with the “boundary” argument, the command causes the program to calculate only the composition of the boundary fluids. With the “initial” argument, the program calculates the initial state of the medium as well as the composition of the boundary fluids, but does not trace a reaction model. Typing the command with the argument “single” causes the simulation to run on a single processor, regardless of the setting for the “threads” command.

Upon completing a simulation, the program normally presents a dialog box asking if you would like to continue the simulation over an extended time span. This option appears if you have triggered the calculation by entering “go” from the keyboard, selecting “Go” from the menubar, or moving to the **Results** pane and clicking the **Run** button, but not if you are reading commands from a file or running a control script. Typing

```
go ask
```

causes the program to offer the option to continue the simulation in all cases.

## 9.45 grep

```
grep <string>
```

**File** → **View** → **.\X2t\_output.txt** → **Edit** → **Find**

Use the “grep” command to search for each occurrence of the given character string in the current output dataset. For example, use

```
grep Albite
```

to obtain a list of each line in “X2t\_output.txt” (or the current output dataset, as modified by the “suffix” command) containing the string “Albite”.

## 9.46 h-m-w

```
h-m-w
```

**File** → **Open** → **Thermo Data...** → **thermo\_hmw.tdat**

Use the “h-m-w” command (abbrev.: “hmw”) to set the program to calculate species’ activity coefficients by using the Harvie-Møller-Weare equations. Executing this command automatically sets the input dataset of thermodynamic data to “thermo\_hmw.tdat”. Note that dataset “thermo\_hmw.tdat” supports calculations at 25°C only.

## 9.47 head\_drop

```
head_drop = <value | ?> <unit>
```

```
head_drop <interval> = <value | ?> <unit>
```

```
head_drop fluid = <boundary fluid>
```

```
head_drop <interval> fluid = <boundary fluid>
```

```
head_drop <left | right> fluid = <boundary fluid>
```

```
head_drop <left | right> <interval> fluid = <boundary fluid>
```

```
head_drop <bottom | top> fluid = <boundary fluid>
```

```
head_drop <bottom | top> <interval> fluid = <boundary fluid>
```

**Flow** → flow field:  calculated →

interval 1: head drop  ▼ cm ▼

Use the “head\_drop” (also: “pot\_drop”) command to set a value for the decrease in hydraulic head (in length units) or hydraulic potential (in units of pressure) across



the medium. This decrease provides the driving force for fluid flow. The program sets hydraulic head or potential along the left boundary to the value specified in the command, and sets head or potential along the right boundary to zero.

At each step in the simulation, the program uses the head or potential values at the boundaries along with the permeability, fluid viscosity, and the rates of production or injection at any wells to calculate the flow field. Setting a positive head or potential drop drives flow from left to right, and a negative value drives flow in the opposite direction.

You set the head drop for a specific interval by including the interval name, or for all intervals by omitting it.

The “head\_drop” command also lets you specify the boundary fluids to set to the left and right sides of the domain, in the same fashion as you would use the “discharge” command. In a simulation employing an imported flow field, and only such a simulation, you may set the fluids to appear at the bottom and top of the domain.

The program recognizes hydraulic head in units of length, as listed in the **Units Recognized** chapter in the **GWB Reference Manual**, and hydraulic potential in units of pressure. By default, values are taken in atm.

Examples:

```
head_drop = 150 cm
pot_drop = 0.1 atm
head_drop start = 4 m phase2 = 2 m
```

By default, the head drop is set to zero. To return to the default value, enter the command

```
head_drop = ?
```

## 9.48 heat\_source

```
heat_source = <field variable | ?> <unit> <steady | transient> \
              <Tmin = <value | min | ?> <unit>> <Tmax = <value | max | ?> <unit>>
heat_source = <off | on | reset>
```

**Medium** →  internal heat source   cal/cm<sup>3</sup>/s ▼  
 min  C ▼ max  C ▼

Use the “heat\_source” command (also: “heat\_src”) to set the rate of internal heat production within the medium. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/cm<sup>3</sup>/s” is the default. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function. By default,

the program does not account for internal heat production. The “?” argument resets the default value of zero.

The “Tmin” and “Tmax” keywords (also: “temp\_max”, “temp\_min”) prescribe the allowable temperature range for the simulation. Set the keywords to specific temperatures, or to “min” or “max”, which respectively represent the lowest and highest temperatures considered in the thermo dataset, as loaded at run time. If you set the temperature range directly, it will be bracketed to that of thermo dataset. For example, the command

```
heat_source Tmin = -1000 C, Tmax = 1000 C
```

is functionally the same as

```
heat_source Tmin = 0 C, Tmax = 300 C
```

when the “thermo.tdat” dataset is loaded, since the dataset’s range is 0°C to 300°C.

The upper and lower temperature bounds serve two purposes. First, the simulation will give an error message and stop if temperature at any point in the domain falls more than 5°C less than the minimum value, or exceeds the maximum value by more than this amount. Second, unless the “extrap” option is set, the model will load for the simulation only those species for which log *K* values are available in the thermodynamic dataset over the specified temperature range.

The temperature bounds specified with the “Tmin” and “Tmax” keywords are the same as those set with the “span” command: The command

```
heat_source Tmin = 20 C, Tmax = 100 C
```

may be equivalently expressed

```
span 20 C to 100 C
```

as long as the heat source option is enabled.

Values for the keywords default to the temperature span of the thermodynamic database, as set in the database header. Keyword “off” disables the heat source, leaving the source rate and temperature range intact, and keyword “on” re-enables the heat source; “reset” disables the source and discards any settings for the source rate and temperature range.

## 9.49 height

```
height = <value> <unit>
```

```
Domain → height (z)  ▼ cm ▼
```

Use the “height” command to set the height along z of the medium. You can use any unit of distance listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cm” is the default.

By default, the domain is 1 cm in height. See also the “length” and “width” commands.

## 9.50 help

```
help <help | subject>
```

**Help**

Use the “help” command (also: “man”) to display online documentation for the GWB package. The command “help help” shows a list of the types of documentation available.

## 9.51 history

```
history
```

**Run** → **History...**

Use the “history” command to view a list of previously executed commands. You can re-execute commands in the **History...** dialog box by double-clicking on a command, selecting one or more commands, and then choosing **Execute**, or with a history substitution command, as explained in the **Command Line Interface** chapter in the **GWB Reference Manual**.

## 9.52 hydrogen-2

```
hydrogen-2 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “hydrogen-2” command (also, “2-H”) to set the  $^2\text{H}$  isotopic composition of the initial fluid, boundary fluids, reactant species (aqueous species, minerals, end members, gases, or oxides), or segregated minerals. The composition may be set on any scale (e.g., SMOW), but you must be consistent throughout the calculation. Example:

```
scope initial
  hydrogen-2 fluid = -10
scope inlet-1
  hydrogen-2 fluid = +4
scope
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
hydrogen-2 remove
hydrogen-2 off
```

clear all settings for <sup>2</sup>H isotopes in the current scope.

See also the “<isotope>” section above, and the “carbon-13”, “oxygen-18”, and “sulfur-34” commands.

## 9.53 inert

```
inert = <field variable | ?> <units>
```

**Medium** → inert volume   ▼ cm3 ▼

Use the “inert” command to set the volume of non-reacting space in the system. You may set a value in units of volume, including cm<sup>3</sup>, m<sup>3</sup>, and *l*, as well as volume% and “vol. fract.”. The default setting is zero and the default unit is cm<sup>3</sup>.

Assuming you have not set a value for the initial fluid fraction in the system using the “porosity” command, the program figures the porosity over the course of the calculation as a derived variable. Specifically, it divides the fluid volume by the sum of the fluid volume, mineral volume, and inert volume, and reports this value as a result.

When you have set a value for initial porosity with the “porosity” command, on the other hand, the program works in the contrary sense. In this case, it calculates the inert volume as that required to form a system of the specified initial porosity; the program now ignores any entry you may have set using the “inert” command.

To restore the default state, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show”.

## 9.54 interval

```
interval <interval> start = <value> <unit>
interval <interval> fluid = <boundary fluid>
interval <interval> <left | right | bottom | top> = <boundary fluid>
interval <interval> rename <new name>
```

**Intervals** →  ▼ at  ▼ day ▼ left  ▼ right  ▼

The “interval” command sets the properties of a named simulation interval, creating it if necessary. The first interval is named “start”, and the terminating point for the run is labeled “end”, but you can create any number of intervening intervals.

For each interval, you set the starting time with keyword “start”, and the boundary fluid to use during the simulation interval with keyword “fluid”. The default time is zero days, and you create fluids with the “scope” command. In the absence of a keyword, the command expects the starting time.

You can use keyword combinations “left fluid” and “right fluid” to set separate fluids on the left and right sides of the domain. In a simulation employing an imported flow field, and only such a simulation, you can also set the fluids to appear at the bottom and top of the domain.

You can also rename the interval, using keyword “rename”. You cannot, however, rename “start” or “end”.

For example, the commands

```
interval start fluid = fluid1
interval phase2 start = 1 year, fluid = fluid2
interval phase3 start = 4 years, left fluid = fluid3A, right fluid = fluid3B
interval end 10 years
```

configure a simulation spanning 10 years and consisting of three reaction intervals, each with a different boundary fluid, or fluids.

## 9.55 isotope\_data

```
isotope_data <dataset>
```

**File** → **Open** → **Isotope Data...**

Use the “isotope\_data” command (also: “idata”) to set the name of the database containing isotope fractionation factors for species, minerals, and gases. Example:

```
isotope_data Isotope.mydata
```

## 9.56 itmax

```
itmax = <value | ?>
```

**Config** → **Iteration...**

Use the “itmax” command to set the maximum number of iterations that may be taken in an attempt to converge to a solution for the equations representing the distribution of chemical mass. By default, this variable is set to 999. To restore the default value,

type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 9.57 Kd

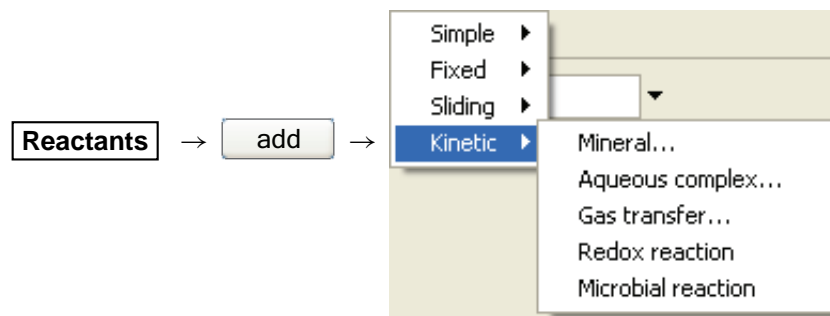
**Kd** <off>

**Config** → **Options...**

The “Kd” command controls whether the program calculates  $K_d$  distribution coefficients for sorbing components, in units of liters per kg sediment mass. This calculation requires that the mineral mass in the system (as specified for individual minerals and/or in terms of inert volume) be set correctly.

## 9.58 kinetic

**kinetic** <species | mineral | end member | gas> <variable> = <value>  
**kinetic** <species | mineral | end member | gas> <variable> = <field variable> \ <steady | transient>  
**kinetic** <species | mineral | end member | gas> <apower | mpower(species)> = <value>  
**kinetic** <species> <boundary fluid> = <value> <unit>  
**kinetic** <redox(label)> <variable> = <value>  
**kinetic** <redox(label)> <variable> = <field variable> <steady | transient>  
**kinetic** <redox(label)> <apower | mpower(species)> = <value>  
**kinetic** <microbe(label)> <variable> = <value>  
**kinetic** <microbe(label)> <variable> = <field variable> <steady | transient>  
**kinetic** <microbe(label)> <apower | mpower(species)> = <value>  
**kinetic** <microbe(label)> <apower | mpower(species)> = <value> \ <apowerA | mpowerA(species)> = <value> \ <apowerD | mpowerD(species)> = <value>



Use the “kinetic” command to set variables defining a kinetic rate law for (1) dissolution or precipitation of any mineral or end member in the initial system or reactant list, (2) the association or dissociation of any aqueous or surface complex in the system modeled,

(3) the transfer of gases into or out of an external reservoir, (4) a redox reaction, including those promoted by catalysis or enzymes, or (5) a microbial metabolism.

In the first three cases, you identify the kinetic reaction by the name of the mineral, end member, species, or gas involved. In the case of a redox reaction, you set a label that begins with the characters “redox”, such as “redox-1” or “redox-Fe”. For a microbial reaction, set a label that starts with “microbe”, such as “microbe-Ecoli”.

The rate law you specify in a “kinetic” command, by default, applies to the dissolution of a mineral or end member, dissociation of a complex, dissolution of a gas, or forward progress of a redox or microbial reaction. The synonymous keywords “forward”, “dissolution”, and “dissociation” set this behavior. Including in a “kinetic” command the keyword “reverse” or its synonyms “precipitation”, “complexation”, “association”, or “exsolution” invokes the opposite behavior. In this case, the rate law applies to the reverse reaction: mineral or end member precipitation, complex association, or gas exsolution.

You can append the “transient” keyword when setting the following field variables: rate constant, specific surface area, activation energy, pre-exponential factor, nucleus area, and critical saturation index. If the variable is defined by an equation, script, or external function, it will then be re-evaluated continuously over the course of the run.

See also the “react” and “remove reactant” commands.

The following paragraphs apply to all types of kinetic reactions. You set the rate constant either directly using the “rate\_con” keyword, or by setting an activation energy and pre-exponential factor with keywords “act\_en” and “pre-exp”. In the absence of promoting and inhibiting species (see next paragraph), you set the rate constant and preexponential factor in (1) mol/cm<sup>2</sup> s for mineral, end member, and gas transfer reactions, (2) molal/s or molal/cm<sup>2</sup> s (the latter when accounting for heterogeneous catalysis) for complexation and redox reactions, and (3) mol/mg s for microbial reactions. The activation energy is specified in J/mol. Example:

```
kinetic "Albite low" rate_con = 1e-15
```

You can set “rate\_con”, “act\_en”, and “pre-exp” as field variables (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

You use the “apower” or “mpower” (also “apow” or “mpow”) keyword to specify any promoting or inhibiting species in the kinetic rate law. Keyword “apower” sets the exponent of a species activity, and “mpower” the exponent of a species molality. Promoting species have positive powers, and the powers of inhibiting species are negative. For example, the command

```
kinetic "Albite low" apower(H+) = 1
```

sets H<sup>+</sup> as a promoting species, the activity of which is raised to a power of one.

You can use aqueous species, minerals (represented by activity, which is one, or molality), end members (activity or mole fraction), gas species (fugacity or partial pressure), surface complexes (molal concentration), and solvent water (activity) as

promoting and inhibiting species. When setting an end member, use keyword “xpower” to set the rate law in terms of mole fraction. When setting a gas, use keyword “fpower” to set the rate law in terms of fugacity, and “ppower” to use partial pressure, instead. The generic keyword “power” sets the activity of the solvent or an aqueous species, the activity of an end member, the fugacity of a gas, and the molality of a surface species.

The “order1” and “order2” keywords set nonlinear rate laws. Keyword “order1” represents the power of the  $Q/K$  term, and “order2” represents the power of the  $(1 - Q/K)$  term.

Use the “rate\_law” keyword to set the form of the kinetic rate law for a specific mineral, end member, species, gas, redox reaction, or microbial metabolism. You may set the keyword equal to (1) a character string containing the rate law, (2) the name of a file containing a basic-like script, or (3) the name of a function in a library. The name of a file containing a rate law script must end in “.bas”. To specify a function from a library, set the name of a dynamic link library (DLL) separated from the function name by a colon (“:”), such as “rate\_laws.dll:my\_ratelaw”; the library file must end in “.dll”. To return to the program’s built-in rate law, enter “rate\_law = off” or “rate\_law = ?”.

The following paragraphs apply to dissolution and precipitation reactions. You set the specific surface area of a kinetic mineral or end member (in  $\text{cm}^2/\text{g}$ ) with the “surface” keyword. For example,

```
kinetic "Albite low" surface = 1000
```

The “cross-affinity” option lets you use the saturation state of one mineral to model the reaction rate of another (or of an end member), as is sometimes useful for example in studying glass dissolution. To do so, you use the “xaffin” option. For (a hypothetical) example, the command

```
kinetic Quartz xaffin = Cristobalite
```

causes the program to calculate the reaction rate of quartz according to the fluid’s saturation state with respect to cristobalite. The command

```
kinetic Quartz xaffin = OFF
```

turns off the option.

Finally, you use the “nucleus” and “critSI” keywords to set the area available for nucleation (in  $\text{cm}^2/\text{cm}^3$  fluid volume) and the critical saturation index above which the mineral or end member can nucleate. Each of these values, by default, is zero.

Keywords “surface”, “nucleus”, and “critSI” can be set as field variables (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

The following paragraphs apply to reactions for aqueous and surface complexes. When you specify a kinetic reaction for the association of an aqueous complex or



surface complex, or its dissociation, you can set the complex's initial concentration directly. The concentration can be set heterogeneously, as a field variable. If you do not specify an initial concentration, or set an entry of "?", the program takes the complex at the start of the simulation to be in equilibrium with the initial fluid.

You specify the initial concentration within a "kinetic" command or as a separate command line. For example, the commands

```
kinetic AIF++ rate_con = 3.3e-6, mpow(AIF++) = 1
AIF++ = 1 umol/kg
```

are equivalent to

```
kinetic AIF++ 1 umol/kg rate_con = 3.3e-6, mpow(AIF++) = 1
```

Either case defines a kinetic reaction for decomposition of the  $\text{AIF}^{++}$  ion pair, setting it initially to a free concentration of  $1 \mu\text{mol kg}^{-1}$ .

In **X1t** and **X2t**, you can similarly set the species' concentration within the boundary fluids directly

```
kinetic AIF++
scope fluid-1
  AIF++ = 4 umol/kg
scope fluid-2
  AIF++ = 8 umol/kg
```

or on a "kinetic" command

```
kinetic AIF++ fluid-1 = 4 umol/kg, fluid-2 = 8 umol/kg
```

If you do not set a value for a boundary fluid, or set it to "?", the program takes the species to be in equilibrium with the fluid.

The following paragraphs apply to gas transfer reactions. Use the "f\_ext" keyword to specify the fugacity of the gas in question in the external reservoir, or keyword "P\_ext" to set its partial pressure. In the latter case, you may append a pressure unit; the default is bar. Keyword "contact" sets the contact area between fluid and external reservoir, in  $\text{cm}^2/\text{kg}$  of water. Example:

```
kinetic CO2(g) f_ext = 10^-3.5, contact = 10
```

Both values can be set as field variables, as described in the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**.

If you do not set a value for the gas' external fugacity, or set "f\_ext = ?", the program uses the fugacity in the initial fluid, at the start of the simulation, and the external fugacity.

## GWB Command Reference

---

The following paragraphs apply to redox reactions. You set the form of the redox reaction to be considered as a character string, using the “rxn” (or “reaction”) keyword. For example,

```
kinetic redox-1 rxn = "Fe++ + 1/4 O2(aq) + H+ -> Fe+++ + 1/2 H2O"
```

To specify that the reaction be promoted by a heterogeneous catalyst, set keyword “catalyst” to the name of the catalyzing mineral, or simply to “on”. In the former case, you use keyword “surface” to set the specific surface area of the catalytic mineral (in  $\text{cm}^2/\text{g}$ ). If you have set “catalyst = on”, however, you use the “surface” keyword to set total catalytic area, in  $\text{cm}^2$ . Setting “catalyst = off” disables the catalysis feature.

To set an enzymatically promoted reaction, set keyword “me” to the name of the aqueous species serving as the enzyme, or simply to the value to be used as the enzyme’s molality. In the former case, the program tracks the enzyme molality  $m_E$  over the course of the simulation from the calculated distribution of species. If you have set a numeric value for  $m_E$  using the “mE” keyword, the program uses this value directly. You may alternatively specify the enzyme species or its activity  $a_E$  using keyword “aE”, in which case variables  $m_E$ ,  $m_A$ , and  $m_P$  in the rate law are replaced by the activities  $a_E$ ,  $a_A$ , and  $a_P$ .

For an enzymatic reaction, you further set the half-saturation constants  $K_A$  and  $K_P$  for the forward and reverse reactions in molal with the “KA” and “KP” keywords. You must set a value for  $K_A$ , but may omit  $K_P$ , in which case the  $m_P/K_P$  term in the rate law will be ignored. Setting “enzyme = off” disables the enzyme feature.

The following paragraphs apply to microbial reactions. You set the form of the metabolic reaction using the “rxn” (or “reaction”) keyword, in the same manner as with redox reactions. For example,

```
kinetic microbe-1 rxn = "CH4(aq) + 2 O2(aq) -> HCO3- + H+ + H2O"
```

Set the half-saturation constants  $K_D$  and  $K_A$  for the electron donating and accepting reactions with the “KD” and “KA” keywords. These values default to zero.

You set the powers of species in the numerator of the rate law with the “mpower” keyword, as with other types of kinetic reactions. Use keywords “mpowerD” and “mpowerA” (or “mpowD” and “mpowA”) to set the powers  $p_D$ , etc., of species from the electron donating and accepting reactions, respectively, within the product functions in the rate law’s denominator. For example,

```
kinetic microbe-1 mpower(CH4(aq)) = 1, mpowerD(CH4(aq)) = 1
```

sets the power of the electron-donating species  $\text{CH}_4(\text{aq})$  to one in both the rate law numerator and denominator. Keywords “PKD” and “PKA” set the overall powers  $p_{KD}$  and  $p_{KA}$  of the electron donating and accepting terms in the denominator of the rate law; by default, these are one.

You set the free energy  $\Delta G_{\text{ATP}}$  of ATP hydrolysis (in kJ/mol) with the “ATP\_energy” keyword, and the value of  $n_{\text{ATP}}$  with keyword “ATP\_number”. These values default to zero.

Use the “biomass” keyword to set the initial biomass concentration, in mg/kg. You can set this value as a field variable (see the **Heterogeneity** appendix to the **GWB Reactive Transport Modeling Guide**).

The “growth\_yield” keyword sets the microbe’s growth yield in mg biomass/mol of reaction progress, and “decay\_con” sets its decay constant in  $\text{s}^{-1}$ ; both values default to zero.

## 9.59 left

```
left fluid = <boundary fluid>
left <interval> fluid = <boundary fluid>
left = <open | closed>
left = <interval> <open | closed>
left = <normal | inlet | outlet>
```

**Flow** → flow field:  calculated →

interval 1:  
left boundary

open

closed

discharge  ▼ cm/s ▼

**Flow** → boundaries: left  ▼

Use the “left” command to control the boundary condition at the left side of the domain: the fluid to appear at the boundary, whether the boundary is open or closed, and whether solutes can diffuse and disperse, and heat conduct, across the boundary.

If the command contains the name of a simulation interval, subsequent settings on the command apply to that interval, rather than the simulation as a whole. This behavior continues to the end of the command, or until the name of another interval is encountered.

You use the “fluid” keyword to specify the boundary fluid to appear on the left boundary, for the entire simulation or an individual interval. For example, the commands

```
left fluid = ambient_water
left phase2 fluid = pit_water
```

set fluid “ambient\_water” on the left boundary over the course of the simulation, except during interval “phase2”, in which case fluid “pit\_water” is found on the boundary.

Keywords “open” and “closed” set the left boundary of the domain open or closed to flow. By default, the boundary is open. The commands

```
left = open  
left = closed
```

are equivalent to the commands

```
discharge left = ?  
discharge left = 0
```

The program by default uses “inlet/free outlet” boundary conditions: where fluid flows into the domain, solute mass can advect, diffuse, and disperse across the boundary, but where it exits the domain, transport occurs by advection alone. Similarly, heat can conduct across an inlet, but not an outlet.

You can force the program to treat the left side of the domain as an inlet or free outlet boundary, regardless of the direction of fluid migration. Keyword “inlet” causes the program to allow diffusion, hydrodynamic dispersion and heat conduction across the left bound, whereas keyword “outlet” creates an insulated boundary sealed against transport by these processes:

```
left outlet
```

In either case, the advective transport of solute mass and heat depends on the direction and rate of fluid migration, and is not affected by the command setting

See also the “right”, “discharge”, “head drop”, and “interval” commands.

### 9.60 length

```
length = <value> <unit>
```

```
Domain →  length (x)  ▼ cm ▼
```

Use the “length” command to set the length along  $x$  of the medium. You can use any unit of distance listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cm” is the default. By default, the domain is 100 cm long. See also the “width” and “height” commands.

### 9.61 log

```
log <variable> = <value>
```

Use the “log” command to set variables on a logarithmic scale. Examples:

```
log fugacity O2(g) = -65
log activity U++++ = -10
```

## 9.62 mobility

```
mobility = <surface_type> = <field variable> <steady | transient>
mobility = <surface_type> = <value>
mobility = <surface_type> <initial = <field variable> <steady | transient>> \
    <boundary_fluid> = <value>
```

**Config** → **Sorbing Surfaces...** → mobility  +  ▼  
 + fluids → fluid  ▼

or **File** → **Open** → **Sorbing Surfaces...**

Use the “mobility” command to set up a complexing surface in your model as a mobile colloid. A mobile colloid is composed of the mineral (or minerals) associated with a complexing surface, as well as the ion complexes present on that surface. Only datasets with model type “two-layer”, “three-layer”, or “cd-music” as set in the dataset header are surface complexation models, and hence only those datasets can be used to form a mobile colloid.

Mobility refers to the fraction of the surface in question that can move in the model by advection and dispersion. A surface with a mobility of one moves freely, whereas a mobility of zero sets the surface to be stationary. Intermediate values arise, for example, when some of the surface is attached to the medium, or when colloid motion is impeded by electrostatic interactions. By default in the software the mobility of any surface is zero.

To set a mobile colloid, begin by reading in a surface complexation dataset using the “surface\_data” command. Then, use the “mobility” command, referencing the surface’s label, to set the colloid’s mobility. The label is given at the head of the surface dataset, on a line beginning “Surface type”. The label in dataset “FeOH.sdat”, for example, is “HFO”. If you omit the label, the program will assume you are referring to the surface complexation dataset most recently read.

You can define the mobility of the domain (i.e., “initial” scope) as a field variable, which means you can have the program calculate mobility using an equation, script, or compiled function you provide. When you set the “transient” keyword, the program upon undertaking each time step in the simulation evaluates mobility at each nodal block. In the “steady” case, which is the default, the program evaluates mobility at each block just once, at the start of the run.

## GWB Command Reference

---

You can further define colloid mobility in a boundary fluid. If you do not set a value, mobility in an inlet stream is taken as the initial value assigned the first interior node in the simulation.

In its simplest form, the command sets mobility for the current scope, either the initial system or a boundary fluid. You can, however, set mobility for one or more scopes with a single command, by citing each scope's name.

For example, since "HFO" is the label for the surface defined by dataset "FeOH.sdat", the commands

```
surface_data FeOH.sdat
scope initial
  mobility HFO = 100%
scope inlet_fluid1
  mobility HFO = 0%
```

are equivalent to

```
surface_data FeOH.sdat
mobility HFO initial = 100%, inlet_fluid1 = 0%
```

Restore the default behavior of immobility by entering a command such as

```
mobility HFO ?
```

## 9.63 modflow

```
modflow <discretization dataset> <short> <long>
```

**Flow** → flow field:  imported from modflow →  
 all intervals: set modflow file

Use the "modflow" command to import a groundwater flow field from the results of a MODFLOW simulation. You specify the name of the discretization file, such as "MyRun.dis". **X2t** will read data from this file as well as a budget file with the same root name ("MyRun.bud"). In the example

```
modflow "My Flow Field.dis"
```

**X2t** will expect to find the MODFLOW data in files "My Flow Field.dis" and "My Flow Field.bud".

The dataset name should be enclosed in quotes if it contains spaces or unusual characters. Alternatively, you may specify only the root of the file name. For example, if you type

```
modflow MyModflow
```

**X2t** will open files “MyModflow.dis” and “MyModflow.bud”.

The “short” and “long” options control the level of detail in the MODFLOW log file “mf\_log.txt”, which **X2t** creates in the user’s profile directory (found by typing %appdata% in the Windows Explorer Address bar, e.g., “C:\Documents and Settings\*userName*\Application Data\GWB”). For details, refer to the **Importing from MODFLOW** appendix to the **GWB Reactive Transport Modeling Guide**. By default, **X2t** assumes the “short” option.

## 9.64 no-precip

```
no-precip <off>
```

**Config** → **Options...**

Use the “no-precip” command (also: “noprecip”) to prevent new minerals from precipitating over the course of a simulation. By default, they are allowed to precipitate. Use:

```
no-precip
no-precip off
```

See also the “precip” command.

## 9.65 nswap

```
nswap = <value | ?>
```

**Config** → **Iteration...**

Use the “nswap” command to set the maximum number of times that the program may swap entries in the basis in an attempt to converge to a stable mineral assemblage. By default, this variable is set to 30. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 9.66 Nx

```
Nx = <value | ?>
```

**Domain** → nodes (Nx)  ▼

Use the “Nx” command to set in the simulation the number of nodal blocks along the  $x$  direction. By default, the program divides the medium into ten blocks along  $x$ .

## 9.67 Ny

**Ny** = <value | ?>

**Domain** → nodes (Ny)  ▼

Use the “Ny” command to set in the simulation the number of nodal blocks along the  $y$  direction. By default, the program divides the medium into ten blocks along  $y$ .

## 9.68 off-diag

**off-diag** <off>

**Medium** → mass transport  include off-diagonal terms in dispersion tensor

Use the “off-diag” command to enable or disable the inclusion of the “off-diagonal” dispersion coefficients (i.e.,  $D_{xy}$  and  $D_{yx}$ ) when calculating the effects of mass transport. By default, these terms are not included. The terms come into play when modeling hydrodynamic dispersion where groundwater flow is not aligned with the  $x$  or  $y$  coordinate axis.

## 9.69 oxygen-18

**oxygen-18** <fluid | reactant | segregated mineral> = <value>

**Config** → **Isotopes...**

Use the “oxygen-18” command (also, “18-O”) to set the  $^{18}\text{O}$  isotopic composition of the initial fluid, boundary fluids, reactant species (aqueous species, minerals, end members, gases, or oxides), and segregated minerals. The composition may be set on any scale (e.g., SMOW), but you must be consistent throughout the calculation. Example:

```
scope initial
  oxygen-18 fluid = -10
scope inlet-1
  oxygen-18 fluid = +4
scope
```



Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
oxygen-18 remove
oxygen-18 off
```

clear all settings for  $^{18}\text{O}$  isotopes in the current scope.

See also the “<isotope>” section above, and the “carbon-13”, “hydrogen-2”, and “sulfur-34” commands.

## 9.70 pause

```
pause
```

Use the “pause” command to cause **X2t** to pause temporarily during input. This command is useful when you are debugging scripts.

## 9.71 pe

```
pe = <field variable>
```

```
Initial or Fluids →  ▼ pe ▼
```

Use the “pe” command to set oxidation state in the initial system in terms of pe. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). Example:

```
pe = 10
```

is equivalent to

```
log activity e- = -10
```

where “e-” is the electron. See also the “activity”, “Eh”, “pH”, “fugacity”, “fix”, and “slide” commands.

## 9.72 permeability

**permeability** <intercept = field variable | ?> <unit> <steady | transient> \

<porosity = field variable | ?> <steady | transient> \

<kxky = field variable | ?> <steady | transient> \

<mineral = field variable | ?> <steady | transient>

**Medium** → A (porosity)  +  ▼

B (intercept)  +  ▼ darcy ▼

kx/ky (anisotropy)  +  ▼

You use the “permeability” command to set the correlation by which the program calculates sediment permeability. The correlation gives log permeability in any of the units listed in the **Units Recognized** chapter of the **GWB Reference Manual** (darcys by default) as a linear function of the porosity (expressed as a volume fraction) of a nodal block and, optionally, the volume fractions of one or more minerals. The “transient” keyword causes the model to evaluate the coefficient in question continuously over the course of the simulation, if it is set with an equation, script, or function.

Examples:

```
permeability intercept = -11 cm2 porosity = 15
permeability Kaolinite = -8
```

The latter command adds a term for the mineral Kaolinite to the existing correlation. To remove a term from the correlation, set a value of “?”. The entry

```
permeability Kaolinite = ?
```

for example, removes the correlation entry for that mineral.

The default correlation is

$$\log k = -5 + 15\phi$$

where  $k$  is permeability in darcys and  $\phi$  is porosity (expressed as a fraction).

## 9.73 pH

pH = <field variable>

**Initial** or **Fluids** →  ▼ pH ▼

Use the “pH” command to set pH in the initial system. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). Example:

pH = 5

is equivalent to

log activity H+ = -5

See also the “activity”, “Eh”, “pe”, “fugacity”, “fix”, and “slide” commands.

## 9.74 phrqpitz

**phrqpitz**

**File** → **Open** → **Thermo Data...** → thermo\_phrqpitz.tdat

Use the “phrqpitz” command to set the program to calculate species’ activity coefficients using the Harvie-Møller-Weare equations, as implemented in the USGS program PHRQPITZ. Executing this command automatically sets the input dataset of thermodynamic data to “thermo\_phrqpitz.tdat”. Note that dataset “thermo\_phrqpitz.tdat” is primarily intended to support calculations at or near 25°C.

## 9.75 pitz\_dgamma

**pitz\_dgamma** = <value | ?>

**Config** → **Iteration...**

Use the “pitz\_dgamma” command to control the relative change in an activity coefficient’s value the program allows during each Newton-Raphson iteration, when the Harvie-Møller-Weare activity model has been invoked. By default, the program allows a 10% change, which corresponds to a value of 0.1.

## 9.76 pitz\_precon

```
pitz_precon = <value | ?>
```

**Config** → **Iteration...**

Use the “pitz\_precon” command to control the maximum number of passes the program takes through the pre-conditioning loop before beginning a Newton-Raphson iteration, when a virial activity model (Harvie-Møller-Weare or SIT) has been invoked. By default, the program makes up to 10 passes. In cases of difficult convergence, counter-intuitively, it can sometimes be beneficial to decrease this value.

## 9.77 pitz\_relax

```
pitz_relax = <value | ?>
```

**Config** → **Iteration...**

The “pitz\_relax” command controls under-relaxation when evaluating the Harvie-Møller-Weare equations. The program at each Newton-Raphson iteration assigns activity coefficients as a weighted average of the newly calculated value and the corresponding value at the previous iteration level. Setting pitz\_relax to zero eliminates under-relaxation, so the newly calculated values are used directly; a value of one, in contrast, should be avoided because it would prevent the activity coefficients from being updated. By default, the program carries an under-relaxation factor of 0.5.

## 9.78 plot

```
plot <character | binary> <on | off>
```

**Config** → **Output...**

Use the “plot” command to set the format of the “X2t\_plot.xtp” dataset. The dataset is written in XML, a standard format that is easy to parse for use with alternative plotting programs. Numerical data in the dataset can be represented in either standard decimal notation (keyword “character”) for user readability or a binary encoding (keyword “binary”) that maintains full precision of data. The default format, XML with binary encoded data, also zips the output file to reduce output size and improve file opening speed. The command “plot off” causes **X2t** to bypass writing calculation results to the “X2t\_plot.xtp” dataset, which is used to pass input to **Xtplot**. By default, the program writes output to the dataset. The command “plot on” (or just “plot”) re-enables the output. To see the current setting, type “show print”.

## 9.79 pluses

**pluses** <off | on | multicore | banner>

**Config** → **Stepping...**

The “pluses” command controls whether the program writes a plus sign (“+”) to the screen each time it converges to a solution of the chemical equations at a nodal block (keyword “on”); by default, it does not (keyword “off”). You can also set the program to write a distinct symbol for each thread active in tracing the simulation (keyword “multicore”), allowing you to observe the model as it runs in parallel. You can also specify the program write a banner giving details such as the number of iterations required for convergence and the final residuals (keyword “banner”). The latter option produces unwieldy amounts of output, except for simple simulations.

## 9.80 polymorphs

**polymorphs** <mineral>

**Config** → **Show...** → Minerals → *select mineral*

Use the “polymorphs” command to list each polymorph (mineral identical in composition to the specified mineral) in the current database. Example:

```
polymorphs Albite
```

## 9.81 porosity

**porosity** = <field variable | ?>

**Medium** → porosity

Use the “porosity” command to set (as a volume fraction) the initial porosity of the medium. Porosity, the fraction of a nodal block occupied by fluid, is the ratio of fluid volume to the sum of fluid, mineral, and inert volume.

The examples

```
porosity = 0.30
porosity = 30%
```

are equivalent.

When you specify the porosity, the program will at each nodal block figure the difference between the volume taken up initially by minerals and fluid, and the block's bulk volume. The program assigns this difference as inert, non-reactive volume (see the "inert" command). In this case, the program ignores any settings that may have been made with the "inert" command.

When you do not specify an initial porosity with the "porosity" command, on the other hand, the program calculates it from volumes in the nodal block of fluid, minerals, and inert space. To restore this default behavior, enter the command with an argument of "?".

### 9.82 precip

```
precip <off>
```

**Config** → **Options...**

Use the "precip off" command to prevent new minerals from precipitating over the course of a simulation. By default, they are allowed to precipitate. Use

```
precip  
precip off
```

See also the "no-precip" command.

### 9.83 press\_model

```
press_model <Tsonopoulos | Peng-Robinson | Spycher-Reed | default | off>
```

**Config** → **Options...**

The "press\_model" command (also: "pressure\_model") lets you control the method used to calculate fugacity coefficients and gas partial pressures. Three nonideal pressure models are coded in the software: Tsonopoulos, Peng-Robinson, and Spycher-Reed, as described in the **GWB Essentials Guide**.

By default, the pressure model is taken from the header lines of the thermo dataset in use, but you can use the "press\_model" command to override the default setting. Keywords "Tsonopoulos", "Peng-Robinson", and "Spycher-Reed" set the pressure model directly (you need only enter the first three letters), whereas "default" returns to the setting in the thermo dataset, and "off" (also: "ideal") disables the feature, forcing all fugacity coefficients to one.

Examples:

```
press_model Peng-Robinson
press_model default
```

## 9.84 print

```
print <option> = <long | short | none>
print <off | on>
print <numeric | alphabetic>
```

**Config** → **Output...**

Use the “print” command (also: “printout”) to control the amount of detail to be written into the “X2t\_output.txt” dataset. For example, the dataset can contain information about each aqueous species, information on only species with concentrations greater than  $10^{-8}$  molal, or no species information.

By default, the program does not produce these datasets, which can become quite lengthy. The command “print on” (or just “print”) causes them to be written, and “print off” disables them.

Options, which may be abbreviated to three letters, and their default settings are:

species	short
surfaces	long
saturation	short
gases	long
basis	none
orig_basis	long
elements	long
reactions	none
stagnant	none

The “stagnant” option includes in the printout the diffusive profile within the stagnant zone, when the dual porosity feature is in use.

The “print” command can also be used to arrange entries in the output dataset either numerically or alphabetically:

```
print numeric
print alphabetic
```

To see the current print settings, type “show print”.

## 9.85 pwd

**pwd**

**File** → **Working Directory...**

The “pwd” command returns the name of the current working directory. The command has the same effect as typing “show directory”. See the “chdir” command.

## 9.86 quit

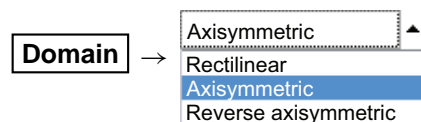
**quit**

**File** → **Quit**

Use the “quit” command (also: “end”, “q”) to exit the program. Touching **Ctrl+Q** serves the same purpose as the “quit” command.

## 9.87 radial

**radial** <r1 = value> <unit> <r2 = value> <unit>  
 <angle = value> <unit> <normal | reverse>  
**radial** <off | on>



Use the “radial” command (also: “axisymmetric” or “wedge”) to set a 2-D radial (i.e., axisymmetric) flow system extending from radius “r1” to radius “r2”, where “r1” is less than “r2”. Radii may be entered in units of distance (see the **Units Recognized** chapter in the **GWB Reference Manual**); “cm” is the default. Keyword “angle” gives the angle of divergence of the medium, in “radians” (the default) or “degrees”. By default, this value is 0.1 radians. To return to a linear geometry, type “radial off”. The “normal” keyword sets the domain to narrow from right to left; the “reverse” keyword sets it narrowing from left to right.

## 9.88 ratio

**ratio** <species ratio> = <value>

**Initial** or **Fluids** →  ▼ ratio ▼



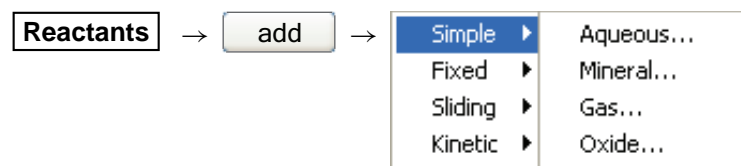
Use the “ratio” command to constrain an activity ratio in the initial system. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command). Example:

```
swap Ca++/Na+2 for Ca++
ratio Ca++/Na+2 = 0.2
```

See also the “activity”, “pH”, “Eh”, “pe”, “fugacity”, “fix”, and “slide” commands.

## 9.89 react

```
react <amount (field variable)> <unit> <as <element symbol>> \
  <species | mineral | end member | gas | oxide> <cutoff> = <field variable>
```



Use the “react” command (abbrev.: “rct”) to define the reactants for the current simulation. To set a kinetic rate law for a reactant, use the “kinetic” command.

Units for the amount of reactant to add over a reaction path can be:

mol	mmol	umol	nmol	
kg	g	mg	ug	ng
eq	meq	ueq	neq	
cm <sup>3</sup>	m <sup>3</sup>	km <sup>3</sup>	l	
mol/kg	mmol/kg	umol/kg	nmol/kg	
molal	mmolal	umolal	nmolal	
mol/l	mmol/l	umol/l	nmol/l	
g/kg	mg/kg	ug/kg	ng/kg	
wt%	"wt fraction"			
g/l	mg/l	ug/l	ng/l	
eq/kg	meq/kg	ueq/kg	neq/kg	
eq/l	meq/l	ueq/l	neq/l	

Units of mass or volume can be expressed per volume of the porous medium. Examples:

mol/cm <sup>3</sup>	g/cm <sup>3</sup>
mmol/m <sup>3</sup>	ug/m <sup>3</sup>
volume%	"vol. fract"

Units of mass or volume can be set as absolute rates by appending “/s”, “/day”, “/yr”, or “/m.y.”. For example,

mmol/s	g/day	cm <sup>3</sup> /yr	
mol/kg/s	mg/kg/day	cm <sup>3</sup> /kg/yr	
mol/cm <sup>3</sup> /s	kg/m <sup>3</sup> /yr	m <sup>3</sup> /m <sup>3</sup> /yr	volume%/m.y.

The latter units refer to rates per cm<sup>3</sup> or m<sup>3</sup> of porous medium, per unit time.

Use the “as” keyword to specify reactant masses as elemental equivalents. For example, the command

```
react 10 umol/kg CH3COO- as C
```

specifies 5 umol/kg of acetate ion, since each acetate contains two carbons, whereas

```
react 20 mg/kg SO4-- as S
```

would cause the program to add 59.9 mg/kg of sulfate, since the ion’s mole weight is about 3 times that of sulfur itself.

You can set a cutoff to limit the amount of a reactant. For example, if you set the amount of a reactant to two moles and set a cutoff of one, then **X2t** will add one mole of the reactant over the first half of the path and none over the second half. Enter the cutoff value in the same units as the amount of reactant. Examples:

```
react 10 grams Quartz
react 1e-2 mol Muscovite cutoff = .5e-2
react .01 mol/day HCl
```

See also the “kinetic” and “remove reactant” commands.

## 9.90 read

```
read <dataset>
```

**File** → **Open** → **Read Script...**

Use the “read” command to begin reading commands from a script stored in a dataset. Example:

```
read Seawater
```

Control returns to the user after the script has been read, unless the script contains a “quit” command. You can also use the “read” command in place of the “data” or “surface\_data” command to read a thermo or surface reaction dataset.

When typing a “read” command, you can use the spelling completion feature to complete dataset names: touch “[tab]” or “[esc]” to cycle through the possible completions, or **Ctrl+D** to list possible completions.

## 9.91 remove

```

remove <basis specie(s)> <solid solution(s)> <reactant(s)> <interval(s)> \
        <boundary fluid(s)> <well(s)> <well interval(s)>
remove basis <basis specie(s)>
remove solid_solution <solid solution(s)>
remove reactant <reactant(s)>
remove interval <interval(s)>
remove scope <boundary fluid(s)>
remove well <well(s)>
remove well <well> interval <interval(s)>
remove wells

```

**Initial** or **Fluids** or **Reactants** or **Intervals** or **Wells** or

**Config** → **Solid Solutions...** →

*select entry(ies)* →

Use the “remove” command (also: “rm”) to eliminate one or more basis entries or reactants from consideration in the calculation. Example:

```

remove Na+
remove Quartz Calcite
remove reactant H2O

```

Components can be reentered into the basis using the “swap”, “add”, “activity”, and “fugacity” commands. You can also use the “remove” command to remove solid solutions configured in a run (but not those set in the thermo dataset):

```

remove solid_solution mySS

```

You can also use the “remove” command to remove boundary fluids and simulation intervals

```

remove scope inlet_fluid1
remove interval phase2

```

from the run configuration. Finally, you can remove wells, or well intervals:

```

remove well injection
remove well extraction-8 interval pumping-2

```

The command

```

remove wells

```

removes all the wells that have been defined.

### 9.92 report

```
report <option>  
report set_digits <value>  
report set_node <x-index> <y-index>
```

Once the program has completed a calculation, you can use the “report” command to return aspects of the calculation results. You can use this command from the **Command** pane to query the program interactively. More significantly, the “report” command provides a means of communicating calculation results to control scripts and external programs running GWB programs by remote control.

For arguments available, see the **Report Command** chapter in the **GWB Reference Manual**. For details about control scripts and the plug-in feature, see the **Control Scripts** and **Plug-in Feature** chapters of the **GWB Reference Manual**. Use the “set\_digits” argument to set the number of significant digits in the values reported, and the “set\_node” argument to identify the node of interest.

### 9.93 reset

```
reset  
reset system  
reset reactants  
reset variables
```

**File** → **Reset Configuration**

Use the “reset” command to begin defining the chemical system again with a clean slate. Your current settings will be lost, and all options will be returned to their default states. The command, however, does not alter the setting for the thermo dataset. The “reset system” command resets only the initial system. Similarly, typing “reset reactants” resets the reactant system, and “reset variables” sets each settable variable to its default value.

### 9.94 resume

```
resume
```

**File** → **Resume**

Use the “resume” command to restore the configuration from when you last exited the program. The program configuration is stored in a file “x2t\_resume.x2t” in your profile directory (e.g., “c:\Documents and Settings\jones\Application Data\GWB”). Your current configuration is discarded.

## 9.95 right

```

right fluid = <boundary fluid>
right <interval> fluid = <boundary fluid>
right = <open | closed>
right = <interval> <open | closed>
right = <normal | inlet | outlet>

```

**Flow** → flow field:  calculated →

interval 1:  
 right boundary  
 open  
 closed  
 discharge  ▼ cm/s ▼

**Flow** → boundaries: right  ▼

Use the “right” command to control the boundary condition at the right side of the domain: the fluid to appear at the boundary, whether the boundary is open or closed, and whether solutes can diffuse and disperse, and heat conduct, across the boundary.

If the command contains the name of a simulation interval, subsequent settings on the command apply to that interval, rather than the simulation as a whole. This behavior continues to the end of the command, or until the name of another interval is encountered.

You use the “fluid” keyword to specify the boundary fluid to appear on the right boundary, for the entire simulation or an individual interval. For example, the commands

```

right fluid = ambient_water
right phase2 fluid = pit_water

```

set fluid “ambient\_water” on the right boundary over the course of the simulation, except during interval “phase2”, in which case fluid “pit\_water” is found on the boundary.

Keywords “open” and “closed” set the right boundary of the domain open or closed to flow. By default, the boundary is open. The commands

```

right = open
right = closed

```

are equivalent to the commands

```
discharge right = ?  
discharge right = 0
```

The program by default uses “inlet/free outlet” boundary conditions: where fluid flows into the domain, solute mass can advect, diffuse, and disperse across the boundary, but where it exits the domain, transport occurs by advection alone. Similarly, heat can conduct across an inlet, but not an outlet.

You can force the program to treat the right side of the domain as an inlet or free outlet boundary, regardless of the direction of fluid migration. Keyword “inlet” causes the program to allow diffusion, hydrodynamic dispersion and heat conduction across the right bound, whereas keyword “outlet” creates an insulated boundary sealed against transport by these processes:

```
right outlet
```

In either case, the advective transport of solute mass and heat depends on the direction and rate of fluid migration, and is not affected by the command setting

See also the “left”, “discharge”, “head drop”, and “interval” commands.

### 9.96 save

```
save <dataset> <hex>
```

**File** → **Save As...**

Use the “save” command to write the current chemical system into a dataset in **X2t** format commands. The dataset can be used as an **X2t** input script. Examples:

```
save  
save kspar.x2t
```

If you don’t specify a dataset name, **X2t** will ask for one. The optional keyword “hex” causes the program to output numbers as hexadecimal values.

## 9.97 scope

```
scope <initial | boundary fluid>
scope <initial | boundary fluid> = <initial | boundary fluid>
scope <boundary fluid> rename <new name>
```

**Initial** or **Fluids**

**Initial** or **Fluids** →

The “scope” command defines whether subsequent user commands apply to the initial condition of the medium, to a boundary fluid, or globally, to all fluids that have been defined. The scope is initially set global. When you enter a scope command with “initial” as an argument

```
scope initial
```

the scope is set to the initial system. For any other argument

```
scope my_fluid
```

the scope is set to a boundary fluid of that name, creating it if necessary.

Entering the command without an argument

```
scope
```

returns the scope global, as does entering any command of global meaning, such as “react” or “kinetic”. To see the current scope, type “show scope”.

The “scope” command can also be used to copy constraints on fluid composition and temperature from one aspect of the calculation to another. For example, the command

```
scope my_fluid2 = my_fluid1
```

copies the basis constraints, temperature, TDS, and density from a boundary fluid “my\_fluid1” to another “my\_fluid2”.

```
scope inlet = initial
```

sets the inlet fluid to be the same as the fluid in the initial system.

You can rename a boundary fluid, but not the initial system, using the “rename” keyword.

## 9.98 script

```
script
script end
```

Use the “script” command to mark the beginning, and optionally the end, of a control script. Control scripts differ from standard input files in that they can contain not only **X2t** commands, but control structures such as loops and if-else branches. Control scripts follow the Tcl syntax, described in [www.tcl.tk](http://www.tcl.tk) and [mini.net/tcl](http://mini.net/tcl), as well as several widely available textbooks.

Within a control script, filenames are written with double rather than single backslashes. For example, a “read” command might appear as

```
read GWB_files\\My_file.x2t
```

within a control script.

## 9.99 segregate

```
segregate <mineral(s)>
segregate <mineral> <value>
segregate <mineral> <initial value> <final value>
segregate <mineral> <value> Xi = <value> <value> Xi = <value>
```

**Config** → **Isotopes...** →

**Config** → **Isotopes...** → *select entry(ies)* →

The “segregate” command causes minerals to be isolated from isotopic exchange over the course of a reaction path. By default, a mineral in the equilibrium system remains in isotopic equilibrium with the fluid and other minerals. A segregated mineral, on the other hand, changes in isotopic composition only when it precipitates from solution; it alters the system’s composition only if it dissolves. Example:

```
segregate Quartz Calcite "Maximum Microcline"
```

Optionally, a fraction of a mineral’s mass may be isotopically segregated, and that fraction may vary linearly with reaction progress. Examples:

```
segregate Quartz 100%, Muscovite 7/10
segregate Ca-Saponite 100% 0%
segregate Ca-Saponite 80% Xi = .3, 20% Xi = .7
```



In the latter example, the program segregates 80% of the mass of Ca-Saponite until the reaction progress variable  $X_i$  reaches .3, decreases the segregated fractionation until it reaches 20% when  $X_i$  equals .7, and then holds the value constant until the end of the path. To display the isotopically segregated minerals, type “show isotopes”.

Note, you use the name of the corresponding mineral to segregate a solid solution end member.

## 9.100 show

```
show <option>
show <aqueous | minerals | solid_solutions | gases | oxides | surfaces> \
  <with | w/> <basis entry | string>
```

**Config** → **Show...**

Use the “show” command to display specific information about the current system or database. The options are:

show	show altered	show aqueous
show basis	show boundary	show commands
show couples	show directory	show elements
show gases	show initial	show isotopes
show minerals	show modflow	show oxides
show printout	show reactants	show scope
show show	show suppressed	show surfaces
show system	show variables	show wells
show system	show variables	show wells

The command “show show” gives a list of show command options. When you type “show aqueous” (or “minerals”, “solid\_solutions”, “gases”, “oxides”, or “surfaces”), the program lists all entries of that type in the thermo database. The “solid\_solutions” option additionally includes solutions defined locally. A long form lets you limit the query to entries composed of a particular basis species or containing a text string in the name:

```
show aqueous with Al+++
show minerals w/ chal
```

There is also a compound form of the “show couples” command:

```
show coupling reactions
```

This command produces a complete list of the redox couples, in reaction form.

## 9.101 simax

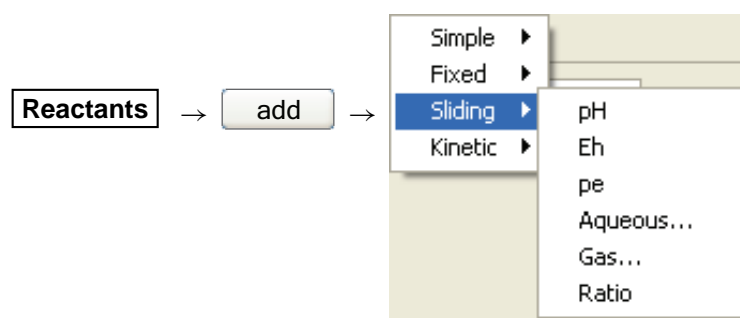
**simax** = <value | ?>

**Config** → Iteration...

The “simax” command sets in molal units the maximum value of the stoichiometric ionic strength used in calculating water activity when Helgeson’s B-dot Debye-Hückel model is employed. By default, this variable is set to 3 molal. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 9.102 slide

**slide** <unit> <species | gas> **to** <field variable>



Use the “slide” command to linearly adjust the activity of the specified species, fugacity of the gas, or an activity ratio toward <value>, which is attained at the end of the path. Note that the interpolation is made linearly on the logarithm of activity or fugacity if <value> is set as a log, and that <unit> can be “activity” or “fugacity” (“a” or “f” for short), “ratio”, “pH”, “pe”, or “Eh”, or omitted. Examples:

```
slide pH to 5
slide activity Cl- to 2/3
slide f CO2(g) to 10^-3.5
slide log f O2(g) to -65
```

## 9.103 solid\_solution

```

solid_solution <name = <label>> <type> <discrete | continuous> \
  <<mineral> tag = <label>> <contains> <<mineral> tag = <label>> \
  <from <value>> <to <value>> <step = <value>> <<variable> = <value>>
solid_solution <solid solution> <options>
solid_solution remove <solid solution>

```

**Config** → **Solid Solutions...** →

**Config** → **Solid Solutions...** → *select entry(ies)* →

Use the “solid\_solution” command (abbrev.: “solid\_soln”) to define binary solid solutions in addition to any provided in the current thermo database. Supply a name with the “name” keyword, or simply set one as the first argument.

You can select the type of solid solution, either “ideal” (the default), “site\_mixing”, “guggenheim”, “regular”, “cubic”, or “third\_order”. For a site-mixing solution, set the site number “bsite”, which defaults to 1. For a Guggenheim solution, set dimensionless parameters “a0”, “a1”, and “a2” or a temperature expansion for their J/mol counterparts using coefficients “p1”–“p9”, each of which defaults to zero. For the other nonideal models, similarly, set coefficients “p1”, “p2”, etc., as appropriate. For more information, see **Solid solutions** under **Configuring the programs** in the **GWB Essentials Guide** and the **Thermo Datasets** chapter in the **GWB Reference Manual**. You can as well choose whether the solid solution is to be treated as continuous, the default option, or as a sequence of “discrete” minerals.

You need to specify two end members for a solid solution, each associated with an abbreviated “tag”. The tag cannot be the name of another entry in the database. You can set a composition range in terms of the mole fraction of the most recently referenced end member, using keywords “from” and “to”, as well as a compositional “step” separating tranches of a discrete solution; by default, composition ranges from zero to one and the step is one-twentieth of the range. Keyword “contains” may be inserted for clarity. Examples:

```

solid_solution name = my_ss Calcite contains Strontianite from 0% to 10% a0= 0.05
solid_solution my_ss Calcite Strontianite from 0 to .1 a0= 0.05
solid_solution Biotite site_mixing Annite Phlogopite bsite= 3

```

If you don’t specify a solid solution’s name, the app will create one automatically using the tags for the two end members, separated by a hyphen. If you don’t set a tag for an end member, the app will pick up the first six characters of the corresponding mineral’s name. For example, the command

```

solid_solution Calcite contains Strontianite

```

produces a solid solution named “Calcit-Stront”, whereas typing

```
solid_solution Calcite tag Cc contains Strontianite tag Str
```

yields the name “Cc-Str”. To redefine the properties of a solid solution, enter the name followed by the properties to be changed:

```
solid_solution my_ss a0 = 1 a1 = 2
```

(You may prefer to use the “alter” command to change solutions defined within the thermo database; the two commands are functionally equivalent.) The command

```
solid_solution remove my_ss
```

discards a previously defined solution.

### 9.104 sorbate

```
sorbate <exclude | include>
```

**Config** → **Options...**

Use the “sorbate” command to tell the program, when considering sorption onto surfaces (see the “surface\_data” command), whether to include or exclude sorbed species in figuring the composition of the initial system. By default, the program does not include sorbed species in this calculation. If you set the  $\text{Ca}^{++}$  concentration to 15 mg/kg, for example, the initial system would contain that amount in the fluid and an additional amount sorbed onto mineral surfaces. If you type the command “sorbate include”, however, that amount would apply to the sum of the  $\text{Ca}^{++}$  sorbed and in solution.

### 9.105 span

```
span <value | min | ?> <unit> <to> <value | max | ?> <unit>
```

```
span <off | on | reset>
```

**Config** → **Options...**

The “span” command lets you set a temperature range across which stability must be defined for each species considered in the calculation. The lower limit may be identified with keyword “from” or “Tmin”, and the upper denoted by “to” or “Tmax”. A value of “?” unsets the bound in question.

With *T*-table datasets, simply setting “span = on” without specifying a range forces the program to load species’ log *K*s in polynomial mode. In other words, it will

take log  $K$ s from polynomial fits against temperature to the stability constants in the thermo dataset, even if the calculation is performed entirely at one of the dataset's principal temperatures. Otherwise, an isothermal calculation run at one of the principal temperatures (commonly 0°C, 25°C, 60°C, ...) takes log  $K$ s for aqueous species, minerals, and so on directly from the values given in the thermo dataset.

Temperature ranges of validity can also be set in the thermo dataset for the virial coefficients used to calculate "Pitzer" and SIT activity coefficients. The "span" command works in a similar manner in such cases, loading only virial coefficients with validity that spans the calculation's temperature range.

As an example of using the "span" command, you may wish to compare results calculated at 25°C with those figured at 30°C. Since log  $K$  for a number of species in the thermo dataset is given at only 25°C, unless the "extrap" option is set, the former calculation may well take into account a somewhat larger set of species than the latter. Entering the command

```
span 25 C to 30 C
```

assures the same set of species is loaded in each case, and that both runs take log  $K$ s from polynomial fits.

Set the bounds to specific temperatures, or to "min" or "max", which respectively represent the lowest and highest temperatures considered in the thermo dataset, as loaded at run time. If you set the temperature range directly, it will be bracketed to that of the thermo dataset. For example, the command

```
span -1000 C to 1000 C
```

is functionally the same as

```
span 0 C to 300 C
```

when the "thermo.tdat" dataset is loaded, since the dataset's range is 0°C to 300°C.

The temperature bounds set here are the same as those set with the "heat\_source" command: The command

```
span 20 C to 100 C
```

may be equivalently expressed

```
heat_source Tmin = 20 C, Tmax = 100 C
```

as long as the heat source option is enabled.

Keyword "off" disables the feature, leaving values for the limits intact, and keyword "on" re-enables the feature. The "reset" keyword clears the limits, disabling the feature. By default, the span feature is disabled.

## 9.106 start\_date

```
start_date <value | off>
```

**Config** → **Stepping...**

Use the “start\_date” command to set an explicit starting date of the reaction. This can be used to coordinate the plotting of dated scatter data samples stored in a **GSS** spreadsheet on the reaction path in **Xtplot**. The date should be in the format “MM/DD/YYYY”. Use “off” to return to the default of not set.

```
start_date 10/30/2008  
start_date off
```

## 9.107 start\_time

```
start_time <value | off>
```

**Config** → **Stepping...**

Use the “start\_time” command to set an explicit starting time of the reaction. This can be used to coordinate the plotting of timed scatter data samples stored in a **GSS** spreadsheet on the reaction path in **Xtplot**. The time should be in the format “HH:MM:SS”. Use “off” to return to the default of not set.

```
start_time 11:30:00  
start_time off
```

## 9.108 step\_increase

```
step_increase = <value | ?>
```

**Config** → **Stepping...**

Use the “step\_increase” command to set the greatest proportional increase, from one step to the next, in the size of the time step. By default, this variable is set to 2.0. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 9.109 step\_max

```
step_max = <value | ?>
```

**Config** → **Stepping...**

Use the “step\_max” command to limit the number of reaction steps the program may take to trace a simulation. Use a “?” to restore the default state, which is no prescribed limit. To see the current setting, type “show variables”.

## 9.110 suffix

```
suffix <string>
```

**Config** → **Output...**

Use the “suffix” command to alter the names of the output datasets (“X2t\_output.txt” and “X2t\_plot.xtp”) by adding a trailing string. Example:

```
suffix _run2
```

produces output datasets with names such as “X2t\_output\_run2.txt”.

## 9.111 sulfur-34

```
sulfur-34 <fluid | reactant | segregated mineral> = <value>
```

**Config** → **Isotopes...**

Use the “sulfur-34” command (also, “34-S”) to set the <sup>34</sup>S isotopic composition of the initial fluid, boundary fluids, reactant species (aqueous species, minerals, end members, gases, or oxides), or segregated minerals. The composition may be set on any scale (e.g., CDT), but you must be consistent throughout the calculation. Example:

```
scope initial
  sulfur-34 fluid = -10
scope inlet-1
  sulfur-34 fluid = +4
scope
```

Note that you use the name of the corresponding mineral to set the isotopic composition of an end member.

The commands

```
sulfur-34 remove
sulfur-34 off
```

clear all settings for <sup>34</sup>S isotopes in the current scope.

See also the “<isotope>” section above, and the “carbon-13”, “hydrogen-2”, and “oxygen-18” commands.

## 9.112 suppress

```
suppress <species, minerals, solid solutions, gases, surface species | ALL>
```

**Config** → **Suppress...**

Use the “suppress” command (also: “kill”) to prevent certain aqueous species, surface species, minerals, solid solutions, or gases from being considered in a calculation. Example:

```
suppress H3SiO4- Quartz "Maximum Microcline"
```

prevents the three entries listed from being loaded from the database. Typing “suppress ALL” suppresses all of the minerals and solid solutions in the thermodynamic database, as well as any solid solutions defined locally.

The “unsuppress” command reverses the process. To suppress all but a few minerals, you could type

```
suppress ALL
unsuppress Quartz Muscovite Kaolinite
```

## 9.113 surface\_capacitance

```
surface_capacitance = <value | ?>
surface_capacitance C1 = <value | ?> C2 = <value | ?>
surface_capacitance on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant capacitance  ▼ F/m<sup>2</sup>  
 or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_capacitance”) to set, in units of F/m<sup>2</sup>, the capacitance or capacitances of a sorbing surface.

For a two-layer complexation model, when you set a capacitance with the command, or if a value for capacitance is set in the header section of the surface reaction dataset,



**X2t** will model surface complexation for the surface in question using the constant capacitance model, rather than the full two-layer model.

A triple-layer or CD-MUSIC complexation model, on the other hand, requires two capacitances,  $C_1$  and  $C_2$ . Default capacitances are set in the header of the surface dataset, but you may use the “C1” and “C2” arguments to override the default settings from the command line.

If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_capacitance on HFO = 2
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface.

## 9.114 surface\_data

```
surface_data <sorption dataset>
surface_data remove <sorption dataset | surface type>
surface_data OFF
```

**File** → **Open** → **Sorbing Surfaces...** →

**File** → **Open** → **Sorbing Surfaces...** → *select surface(s)* →

or **Config** → **Sorbing Surfaces...**

Use the “surface\_data” command (abbrev.: “surf\_data”) to specify an input dataset of surface sorption reactions to be considered in the calculation. The dataset name should be enclosed in quotes if it contains any unusual characters. Use the “remove” argument to eliminate a surface dataset, specified by name or surface type (e.g., “HFO”), from consideration. The argument “OFF” disables consideration of all surface complexes.

You can specify more than one sorbing surface in a model by repeating the “surface\_data” command for different datasets (a dataset of surface reactions for sorption onto hydrous ferric oxide, several triple-layer and CD-MUSIC datasets, as well as example datasets for the ion exchange,  $K_d$ , Freundlich, and Langmuir models are distributed with the software). To remove a dataset of surface reactions from consideration, you use commands such as

```
surface_data remove FeOH.sdat
surface_data remove HFO
surface_data OFF
```

The latter command removes all of the surface datasets that have been loaded.

## 9.115 surface\_potential

```
surface_potential = <value | ?>
surface_potential on <type> = <value | ?>
```

**Config** → **Sorbing Surfaces...** → constant potential  ▼ mV  
 or **File** → **Open** → **Sorbing Surfaces...**

Use this command (abbrev.: “surf\_potential”) to set, in units of mV, the electrical potential for a two-layer sorbing surface. When you set this value (or if a value is set in the header section of the surface reaction dataset), **X2t** will model surface complexation for the surface in question using the constant potential method, rather than invoking the full two-layer model.

If you have set more than one sorbing surface (using the “surface\_data” command), you identify the surface in question by its “type”. For example,

```
surface_potential on HFO = 0
```

The “type” associated with each surface is listed at the top of each dataset of surface reactions. The “type” of the hydrous ferric oxide surface represented by the dataset “FeOH.sdat”, for example, is “HFO”. You can use the “show” command to display the “type” of each active surface.

## 9.116 swap

```
swap <new basis> <for> <basis species>
```

**Initial** or  **Fluids** → 

Use the “swap” command to change the set of basis entries. All reactions are written internally in terms of a set of basis species that you can alter to constrain the composition of the initial system. An aqueous species, mineral, gas, or activity ratio can be swapped into the basis in place of one of the original basis species listed in the database. The swap operates on either the initial system, a boundary fluid, or globally on all fluids, depending on the current input scope (see the “scope” command). Examples:

```
swap CO3--      for HCO3-
swap Quartz     for SiO2(aq)
swap CO2(g)     for H+
swap O2(g)      for O2(aq)
swap Ca++/Na+^2 for Ca++
```

Each end member of a continuous solid solution can similarly be swapped into the basis. Use the syntax <solid solution>::<end member tag>. For example, the sodium end member Albite (tag “Ab”) of the plagioclase feldspar solid solution (name “Plag”) can be swapped into the basis simultaneously with the calcium end member Anorthite (tag “An”) as follows:

```
swap Plag::Ab for Na+
swap Plag::An for Al+++
```

The new species must contain in its composition the original basis species being swapped out (you can’t swap lead for gold). For example, CO<sub>2</sub>(g) is composed of HCO<sub>3</sub><sup>-</sup>, H<sup>+</sup>, and water in “thermo.tdat”. The reactions in the thermo dataset (once reduced to the set of basis and redox species and modified to reflect enabled redox couples) show the basis entries for which a species may be swapped. For a list of original basis species, type “show basis”. To reverse a swap, type “unswap <species>”.

## 9.117 system

```
system <system command>
```

Use the “system” command to execute a command on the computer’s operating system (a “DOS command”) without leaving **X2t**. Example:

```
system "copy X2t_output.txt X2t_output2.txt"
```

Enclose multiword system commands in quotes. An alternative way to fork a system command is to begin the **X2t** command with a “\$”:

```
$copy X2t_output.txt X2t_output2.txt
```

In this case, no quotation marks are needed.

When a “system” command is executed, a “Command Prompt” window will appear briefly on your screen. Due to limitations of the Windows operating system, you cannot fork a command that requires user input, and you will not be able to see any output (including error messages) that might be generated by the command.

## 9.118 TDS

```
TDS = <value | ?>
```

```
Initial or Fluids → + advanced → TDS [ ] ▼ mg/kg
```

Use the “TDS” command to set in mg/kg the total dissolved solids for the current input scope, if you don’t want the program to calculate this value automatically. The program uses the TDS when needed to convert input constraints into molal units. The setting applies to the initial system, a boundary fluid, or globally to all fluids, depending on the current input scope (see the “scope” command).

To restore automatic calculation of the TDS, type the command with no argument or with an argument of “?”. To see the variable’s current setting, type “show variables”.

## 9.119 temperature

```

temperature = <field variable> <unit>
temperature = <value> <unit>
temperature initial = <field variable> <unit> <boundary fluid> = <value> <unit>
temperature isothermal
temperature constant = <on | off>
temperature reset
    
```

**Initial** → temperature   C ▼

**Fluids** → temperature  C ▼

Use the “temperature” command (also: “T”) to set the temperature of the initial system or the boundary fluids. The **Units Recognized** chapter in the **GWB Reference Manual** lists possible units, which default to “C”. By default, the initial system and boundary fluids are set to 25°C and, in the absence of a heat source, the simulation proceeds isothermally.

In its simple usage, the command sets the temperature of the current scope:

```

scope initial
  temperature = 25 C
scope inlet_fluid1
  temperature = 100 C
    
```

but you can also specify scope as a keyword on the command line

```

temperature initial = 25 C, inlet_fluid1 = 100 C
    
```

In the example, hot fluid invades an initially cool domain.

Keyword “isothermal” sets the boundary fluid temperature(s) to the initial temperature for the domain. Keyword “constant” causes the program to hold temperature at each nodal block invariant over the course of the simulation, and the “reset” keyword returns temperature variables to their default state.

You should enter temperature values within the range of the thermo dataset in use. For the default dataset “thermo.tdat”, the range is 0°C to 300°C.

## 9.120 thermal\_cond

```
thermal_cond = <field variable | ?> <unit> <steady | transient>
```

**Medium** → thermal conductivity   cal/cm/s/C ▼

Use the “thermal\_cond” command (abbrev.: “tcond”) to set the thermal conductivity of the fluid-saturated medium. You can specify one of the units listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cal/cm/sec/°C” is the default. This value is used during polythermal simulations in calculating the effects of advective heat transport.

By default, this variable is set to 0.004 cal/cm/s/°C. The “transient” keyword causes the model to evaluate the field variable continuously over the course of the simulation, if it is set with an equation, script, or function. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show system”.

## 9.121 theta

```
theta = <value | ?>
```

**Config** → **Stepping...**

Use the “theta” command to set the time weighting variable used in evaluating kinetic rate laws. The value may vary from zero (full weighting at the old time level) to one (full weighting at the new time level). By default, this variable is set to 0.6. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 9.122 threads

```
threads = <value | ?>
```

**Config** → **Stepping...**

Use the “threads” command to control the number of threads the program spawns when running a simulation. Using more than one thread on multicore or multiprocessor computers, the program runs in parallel, dividing the computing work among the computing cores available. By default (use the “?” argument), the program spawns one thread for each computing core. Typing the command without an argument returns the current setting and the number of computing cores available.

## 9.123 timax

```
timax = <value | ?>
```

**Config** → **Iteration...**

The “timax” command sets in molal units the maximum value of ionic strength used in calculating species’ activity coefficients when Debye-Hückel methods are employed. The limiting value is also used when water activity is calculated according to Wolery (1992b). By default, this variable is set to 3 molal. To restore the default value, type the command with no argument or with an argument of “?”. To see the current setting of this variable, type “show variables”.

## 9.124 time

```
time start = <value> <unit> <interval> = <value | ?> <unit> \
end = <value> <unit>
```

**Intervals** →

start	at	<input type="text"/>	▼ day ▼
interval_1	at	<input type="text"/>	▼ day ▼
end	at	<input type="text"/>	▼ day ▼

Use the “time” command (abbrev.: “t”) to set the time span (keywords “start” and “end”) of the simulation, and the start times for individual simulation intervals. The starting time for the run, by default, is zero, and the default end time is 1 day. Unit choices are listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “day” is the default. Examples:

```
time end = 100 years
time start 10 days, end 20 days
time phase2 = 1 year, phase3 = 2 years, end = 10 years
```

Note that starting times and the end time are commonly and perhaps more clearly specified instead on the “intervals” command.

## 9.125 title

```
title <character string>
```

**Config** → **Output...**

Use the “title” command to set a title to be passed to the graphics program. Example:

```
title "Yucca Mountain Groundwater"
```

Be sure to put multiword titles in quotes.

## 9.126 top

```
top fluid = <boundary fluid>
top <interval> fluid = <boundary fluid>
top = <normal | inlet | outlet>
```

**Flow** → boundaries: top  ▼

In simulations in which the flow field is imported, and only such simulations, use the “top” command to control the boundary condition at the top side of the domain: the fluid to appear at the boundary, and whether solutes can diffuse and disperse, and heat conduct, across the boundary.

If the command contains the name of a simulation interval, subsequent settings on the command apply to that interval, rather than the simulation as a whole. This behavior continues to the end of the command, or until the name of another interval is encountered.

You use the “fluid” keyword to specify the boundary fluid to appear on the top boundary, for the entire simulation or an individual interval. For example, the commands

```
top fluid = ambient_water
top phase2 fluid = pit_water
```

set fluid “ambient\_water” on the top boundary over the course of the simulation, except during interval “phase2”, in which case fluid “pit\_water” is found on the boundary.

The program by default uses “inlet/free outlet” boundary conditions: where fluid flows into the domain, solute mass can advect, diffuse, and disperse across the boundary, but where it exits the domain, transport occurs by advection alone. Similarly, heat can conduct across an inlet, but not an outlet.

You can force the program to treat the top of the domain as an inlet or free outlet boundary, regardless of the direction of fluid migration. Keyword “inlet” causes the program to allow diffusion, hydrodynamic dispersion and heat conduction across the top bound, whereas keyword “outlet” creates an insulated boundary sealed against transport by these processes:

```
top outlet
```

In either case, the advective transport of solute mass and heat depends on the direction and rate of fluid migration, and is not affected by the command setting

See also the “left”, “bottom”, “right”, “discharge”, “head drop”, and “interval” commands.

## 9.127 unalter

**unalter** <species | mineral | solid solution | gas | surface species | **ALL**>

**Config** → **Alter Log Ks...** → *select entry(ies)* →

Use the “unalter” command to reverse the effect of having changed the log  $K$ ’s for a species, mineral, gas, or surface reaction, the selectivity coefficient for an exchange species, the  $K_d$  for a sorbed species, the  $K_f$  and  $n_f$  for a Freundlich species, or the properties of a solid solution in the current dataset. Example:

```
unalter Quartz
```

In this case, the log  $K$  values for quartz revert to those in the current thermodynamic dataset. The argument “ALL” resets the log  $K$ ’s, selectivity coefficients,  $K_d$ ’s, or  $K_f$ ’s and  $n_f$ ’s for all species, minerals, gases, and surface species, as well as the properties of all solid solutions in the dataset.

## 9.128 unsegregate

**unsegregate** <mineral(s) | **ALL**>

**Config** → **Isotopes...** → *select entry(ies)* →

Use the “unsegregate” command to remove minerals from the list of minerals to be segregated isotopically.

## 9.129 unsuppress

**unsuppress** <species, minerals, solid solutions, gases, surface species | **ALL**>

**Config** → **Suppress...**

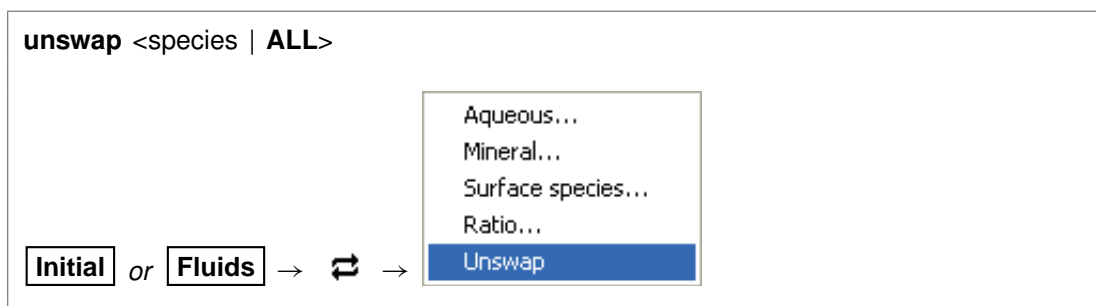
Use the “unsuppress” command (also: “include”) to include in the calculation aqueous species, surface species, minerals, solid solutions, or gases that have previously been suppressed. Examples:

```
unsuppress Quartz Albite "Albite low"
unsuppress ALL
```

The argument “ALL” clears any species, minerals, solid solutions, or gases that have been suppressed.



## 9.130 unswap



Use the “unswap” command to reverse a basis swap. The swap reversal operates on either the initial system, a boundary fluid, or globally on all fluids, depending on the current input scope (see the “scope” command). Example:

```
unswap Quartz (or unswap SiO2(aq))
```

to reverse the effect of the previous command

```
swap Quartz for SiO2(aq)
```

At this point, SiO<sub>2</sub>(aq) is back in the basis. The “ALL” argument reverses all basis swaps.

## 9.131 usgovt



Display the restricted rights legend that governs use, duplication, or disclosure by the U.S. Government.

## 9.132 well

```

well <name> <x = <value> <unit>> <y = <value> <unit>> \
    <<rate | inject> = <value> <unit>> \
    <<head | potential> = <value> <unit>> \
    <start = <value> <unit>> <end = <value> <unit>> \
    <fluid = <boundary fluid>>
well <name> interval <interval> \
    <<rate | inject> = <value> <unit>> \
    <<head | potential> = <value> <unit>> \
    <start = <value> <unit>> <end = <value> <unit>> \
    <fluid = <boundary fluid>>
well <name> rename <new name>
well <name> interval <interval> rename <new name>
well
    
```

→   ▼ x  ▼ cm y  ▼ cm

▼ fluid  ▼ start  ▼ day ▼ end  ▼ day ▼

▼  ▼ cm<sup>3</sup>/s ▼

or

▼  ▼ atm ▼

Use the “well” command to define wells and set well properties. Create a well by issuing the “well” command with a new well name. If you omit the well name from the command line, the program assumes the command applies to the most recently referenced well, the name of which is given by typing “well” without arguments. Type “show wells” to see a list of wells and their properties.

To define a well, you need to set its position along the  $x$  and  $y$  directions, with the “ $x$ ” and “ $y$ ” keywords. You set position in units of distance (see the **Units Recognized** chapter in the **GWB Reference Manual**); cm is the default. The  $x$  position can vary from zero to the length of the domain, and the  $y$  position can vary from zero to the domain’s width.

You need to set also either the well’s injection rate with the “rate” or “inject” keyword, or its hydraulic head or hydraulic potential with the “head” or “potential” keyword. You set injection rates in units of flow rate (see the **Units Recognized** chapter) such as cm<sup>3</sup>/s, the default; head in distance units, defaulting to cm; and hydraulic potential in pressure units, which default to atm. A negative rate sets a producing rather than injecting well.

You then use the “fluid” keyword to associate a fluid with the well. Specifically, you need to set a fluid for each injecting well, and each well set at known head. Producing wells, however, need not be associated with a fluid.

For example, in the commands

```
well well-1 x = 13 m, y = 22 m, rate = 14 m3/day, fluid = injected_fluid
well well-2 x = 1.9 km, y = .38 km, head = 210 m, fluid = fluid-2
```

well “well-1” injects  $14 \text{ m}^3 \text{ day}^{-1}$  of the fluid “injected\_fluid”, whereas “well-2” operates at known head, and as such may either inject fluid “fluid-2” or produce water from the domain, depending on the flow field calculated by **X2t**.

A well by default operates over the course of the **X2t** simulation, but you can specify a specific time range with the “start” and “end” keywords. For example, the commands

```
time start 0 years, end 10 years
well well-1 x = 2 km, y = 3 km
well well-2 x = 1.5 km, y = 8 km, start = 2 years, end = 3 years
```

set well “well=1” to operate from 0 years to 10 years, whereas “well-2” operates over only part of the simulation, from 2 years to 3 years.

You can further define discrete operating intervals for a single well, using the “interval” keyword. For example, executing the commands

```
well well-1 x = 1.2 km, y = 13.7 km
well well-1 interval period-1 end 120 days rate = 0.2 m3/min fluid = fluid-1
well well-1 interval period-2 start 120 days rate = 0.3 m3/min fluid = fluid-2
```

well “well-1” will inject fluid-1 at  $0.2 \text{ m}^3 \text{ min}^{-1}$  until the simulation reaches 120 days, after which it will inject fluid-2 at  $0.3 \text{ m}^3 \text{ min}^{-1}$ . Note there may be any number of intervals, and you may leave hiatuses among them, during which the well does not operate.

Finally, you can use the “rename” keyword to recast the name of a well or a well interval. For example, after running the commands

```
well your_well rename my_well
well my_well interval your_interval rename my_interval
```

interval “your\_interval” of well “your\_well” is known as “my\_interval”, and the well is now “my\_well”.

See also the “remove” command.

### 9.133 width

**width** = <value> <unit>

**Domain** → width (y)  ▼ cm ▼

Use the “width” command to set the width along  $y$  of the medium. You can use any unit of distance listed in the **Units Recognized** chapter in the **GWB Reference Manual**; “cm” is the default. Domain width defaults to 100 cm in **X2t**. See also the “length” and “height” commands.

### 9.134 xtplot

**xtplot** <xtplot options>

**Run** → **Xtplot**

The “xtplot” command launches program **Xtplot** from the **X2t** command line, invoking input from the current graphics interface dataset (by default, “X2t\_plot.xtp”; see the “suffix” command). For example,

```
xtplot -graph map
```

starts **Xtplot** initially showing a map view plot.

### 9.135 Xstable

**Xstable** = <value>

**Config** → **Stepping...**

Use the “Xstable” command to control how the stability criterion for dispersive transport and thermal conduction is applied. A value of one sets the theoretically largest stable time step for an ideal situation. **X2t** simulations are not necessarily ideal (for example, the solute may react and the medium may not be uniform), so this limiting time step may in fact be too large to be stable. Setting “Xstable” to a value smaller than one results in a more stringent constraint on the time step, and hence greater stability. The default value for this variable is 1.0, the theoretical limit. See also the “Courant” command.

# Index

---

- Act2 commands, 27
- activity, 3, 27, 57, 87, 129, 191, 252, 322
- add, 3, 27, 57, 87, 130, 192, 252, 322
- alkalinity, 88, 130, 192, 252, 322
- alter, 4, 28, 58, 88, 130, 192, 253, 323
- audit, 132, 254, 324
  
- b-dot, 89, 132, 194, 255, 325
- background, 29, 59
- balance, 90, 132, 194, 255, 325
- banner, 29, 59
- block, 5
- bottom, 325
  
- carbon-13, 90, 133, 194, 255, 326
- chdir, 5, 29, 59, 90, 133, 195, 256, 327
- clear, 6, 30, 60, 91, 134, 195, 256, 327
- clipboard, 6, 30, 60, 91, 134, 195, 256, 327
- color\_by\_sample, 30, 60
- conductivity, 91, 134, 196, 257, 328
- couple, 6, 30, 60, 92, 134, 196, 257, 328
- Courant, 257, 328
- cpr, 135, 196, 258, 329
- cpu\_max, 135, 197, 258, 329
- cpw, 135, 197, 258, 329
- custom, 31, 61
  
- data, 7, 31, 61, 92, 136, 197, 259, 330
- debye-huckel, 89, 132, 194, 255, 325
- decouple, 7, 31, 61, 92, 136, 198, 259, 330
- delQ, 137, 198, 260, 331
- deltar, 260, 331
- deltax, 260, 331
- deltay, 332
- delxi, 137, 198, 261, 333
- density, 93, 138, 199, 261, 333
- diagram, 32, 62
  
- diffusion\_coef, 262, 334
- discharge, 262, 334
- dispersivity, 263, 336
- dual\_porosity, 94, 139, 264, 337
- dump, 140, 200
- dx\_init, 140, 200, 265, 338
- dxplot, 141, 265, 338
- dxprint, 141, 201, 266, 339
  
- Eh, 7, 32, 62, 94, 141, 201, 267, 340
- end-dump, 142
- epsilon, 95, 142, 201, 267, 340
- exchange\_capacity, 95, 142, 202, 267, 340
- explain, 8, 33, 63, 96, 143, 202, 268, 341
- explain\_step, 143, 203, 268, 341
- extrapolate, 33, 63, 96, 143, 203, 269, 342
  
- factor, 8
- Field variables, 1
- fix, 144, 204, 269, 342
- flash, 144, 204
- flow-through, 145, 205
- flush, 145, 205
- font, 33, 63
- fugacity, 8, 34, 64, 96, 145, 205, 270, 343
  
- gases, 34, 64
- go, 9, 35, 65, 97, 146, 206, 270, 343
- grep, 35, 65, 97, 146, 206, 271, 344
- gridlines, 35, 65
- gtplot, 97, 146
- GWB commands, 1
  
- h-m-w, 98, 147, 206, 271, 344
- head\_drop, 271, 344
- heat\_source, 147, 207, 272, 345
- height, 273, 346

## Index

---

- help, 9, 35, 65, 98, 148, 208, 274, 347
- history, 9, 36, 66, 98, 148, 208, 274, 347
- hydrogen-2, 98, 149, 208, 274, 347
  
- inert, 99, 149, 209, 275, 348
- interval, 275, 348
- ionic\_strength, 10
- isotope, 86, 129, 191, 251, 321
- isotope\_data, 99, 150, 209, 276, 349
- itmax, 100, 150, 210, 276, 349
  
- Kd, 100, 150, 210, 277, 350
- kinetic, 151, 211, 277, 350
  
- labels, 36, 66
- left, 282, 355
- legend, 36, 66
- length, 283, 356
- line, 37, 67
- log, 10, 38, 68, 100, 155, 215, 283, 356
- long, 10
  
- minerals, 38, 68
- mobility, 101, 156, 216, 283, 357
- modflow, 358
- molality, 11
- mosaic\_bounds, 38, 68
- mosaic\_labels, 39, 69
  
- no-precip, 157, 217, 285, 359
- nswap, 157, 217, 285, 359
- Nx, 217, 285, 359
- Ny, 217, 360
  
- off-diag, 360
- output, 11
- oxygen-18, 102, 157, 218, 285, 360
  
- p2plot, 218
- paper, 39, 69
- pause, 11, 39, 69, 102, 158, 219, 286, 361
- pe, 11, 40, 70, 102, 158, 219, 286, 361
- permeability, 103, 159, 219, 287, 362
- pH, 12, 40, 70, 104, 159, 220, 288, 363
- Phase2 commands, 189
- phrqpitz, 104, 160, 220, 288, 363
- pickup, 160
- pitz\_dgamma, 104, 161, 221, 288, 363
- pitz\_precon, 105, 161, 221, 289, 364
- pitz\_relax, 105, 161, 221, 289, 364
  
- plot, 105, 162, 222, 289, 364
- pluses, 162, 222, 290, 365
- polymorphs, 12, 40, 70, 106, 162, 222, 290, 365
- porosity, 106, 163, 223, 290, 365
- precip, 163, 223, 291, 366
- press\_model, 106, 164, 224, 291, 366
- pressure, 41, 71
- print, 41, 71, 107, 164, 224, 292, 367
- pwd, 13, 41, 71, 108, 165, 225, 293, 368
  
- quit, 13, 42, 72, 108, 165, 225, 293, 368
  
- radial, 293, 368
- ratio, 13, 42, 72, 108, 166, 225, 294, 368
- react, 13, 166, 226, 294, 369
- React commands, 127
- reactants, 167, 227
- read, 14, 42, 72, 109, 168, 228, 295, 370
- remove, 14, 43, 73, 109, 168, 228, 296, 371
- report, 15, 110, 169, 297, 372
- reset, 15, 43, 73, 110, 169, 229, 297, 372
- resume, 15, 43, 73, 110, 169, 229, 297, 372
- reverse, 16
- right, 298, 373
- Rxn commands, 3
  
- save, 16, 44, 74, 111, 170, 229, 299, 374
- scatter, 44, 74
- scope, 230, 299, 375
- script, 16, 44, 74, 111, 170, 230, 300, 376
- segregate, 112, 171, 231, 301, 376
- short, 17
- show, 17, 45, 75, 112, 171, 231, 301, 377
- simax, 113, 172, 232, 302, 378
- slide, 173, 233, 303, 378
- solid\_solution, 18, 46, 76, 114, 173, 233, 303, 379
- sorbate, 115, 175, 235, 305, 380
- sorbed\_mass, 19
- span, 47, 77, 115, 175, 235, 305, 380
- SpecE8 commands, 85
- speciate, 48, 78
- spherical, 306
- start\_date, 176, 236, 307, 382
- start\_time, 177, 237, 307, 382
- step\_increase, 177, 237, 307, 382
- step\_max, 177, 237, 308, 383
- stoich\_IS, 19

suffix, 20, 48, 78, 116, 177, 238, 308, 383  
sulfur-34, 117, 178, 238, 308, 383  
suppress, 49, 78, 117, 178, 238, 309, 384  
surface\_capacitance, 20, 118, 179, 239,  
309, 384  
surface\_charge, 21  
surface\_data, 21, 118, 180, 240, 310, 385  
surface\_potential, 22, 119, 180, 240, 311,  
386  
swap, 22, 49, 79, 119, 181, 241, 311, 386  
system, 23, 50, 79, 120, 182, 242, 312, 387

Tact commands, 57  
TDS, 121, 182, 242, 312, 387  
temperature, 24, 50, 121, 183, 243, 313, 388  
thermal\_cond, 314, 389  
theta, 183, 243, 314, 389  
threads, 244, 314, 389  
timax, 121, 184, 244, 315, 390  
time, 184, 244, 315, 390  
title, 51, 80, 122, 184, 245, 315, 390  
top, 391  
trace, 51, 80  
truncate, 52, 81

unalter, 24, 52, 81, 122, 185, 245, 316, 392  
unit, 85, 127, 189, 249, 319  
unsegregate, 122, 185, 245, 316, 392  
unspeciate, 52, 82  
unsuppress, 53, 82, 123, 185, 246, 316, 392  
unswap, 25, 53, 83, 123, 186, 246, 317, 393  
usgovt, 25, 54, 83, 123, 186, 246, 317, 393

volume, 124, 186, 247

water\_limits, 54, 83  
well, 394  
width, 318, 396

x-axis, 54, 84  
X1t commands, 249  
X2t commands, 319  
xss, 25  
Xstable, 318, 396  
xtplot, 318, 396

y-axis, 55, 84