

Ecosystem

Version 0.0 beta

Generated by Doxygen 1.8.3.1

Thu Sep 17 2015 12:01:02

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	File Index	5
3.1	File List	5
4	Class Documentation	7
4.1	ARNOLDI_DATA Struct Reference	7
4.1.1	Member Data Documentation	7
4.1.1.1	beta	7
4.1.1.2	e1	7
4.1.1.3	Hkp1	7
4.1.1.4	hp1	7
4.1.1.5	iter	7
4.1.1.6	k	7
4.1.1.7	Output	7
4.1.1.8	sum	7
4.1.1.9	v	7
4.1.1.10	Vk	8
4.1.1.11	w	8
4.1.1.12	yk	8
4.2	Atom Class Reference	8
4.2.1	Constructor & Destructor Documentation	9
4.2.1.1	Atom	9
4.2.1.2	~Atom	9
4.2.1.3	Atom	9
4.2.1.4	Atom	9
4.2.2	Member Function Documentation	9
4.2.2.1	AtomCategory	9

4.2.2.2	AtomicNumber	9
4.2.2.3	AtomicWeight	9
4.2.2.4	AtomName	9
4.2.2.5	AtomState	9
4.2.2.6	AtomSymbol	9
4.2.2.7	BondingElectrons	9
4.2.2.8	DisplayInfo	9
4.2.2.9	editAtomicWeight	9
4.2.2.10	editElectrons	9
4.2.2.11	editNeutrons	9
4.2.2.12	editOxidationState	9
4.2.2.13	editProtons	9
4.2.2.14	editValence	10
4.2.2.15	Electrons	10
4.2.2.16	Neutrons	10
4.2.2.17	OxidationState	10
4.2.2.18	Protons	10
4.2.2.19	Register	10
4.2.2.20	Register	10
4.2.2.21	removeElectron	10
4.2.2.22	removeNeutron	10
4.2.2.23	removeProton	10
4.2.3	Member Data Documentation	10
4.2.3.1	atomic_number	10
4.2.3.2	atomic_weight	10
4.2.3.3	Category	10
4.2.3.4	electrons	10
4.2.3.5	Name	10
4.2.3.6	NaturalState	10
4.2.3.7	neutrons	10
4.2.3.8	oxidation_state	10
4.2.3.9	protons	10
4.2.3.10	Symbol	10
4.2.3.11	valence_e	10
4.3	BACKTRACK_DATA Struct Reference	10
4.3.1	Member Data Documentation	11
4.3.1.1	alpha	11
4.3.1.2	constRho	11
4.3.1.3	Fk	11
4.3.1.4	lambdaMin	11

4.3.1.5	normFkp1	11
4.3.1.6	rho	11
4.3.1.7	xk	11
4.4	BiCGSTAB_DATA Struct Reference	11
4.4.1	Member Data Documentation	12
4.4.1.1	alpha	12
4.4.1.2	bestres	12
4.4.1.3	bestx	12
4.4.1.4	beta	12
4.4.1.5	breakdown	12
4.4.1.6	iter	12
4.4.1.7	maxit	12
4.4.1.8	omega	12
4.4.1.9	omega_old	12
4.4.1.10	Output	12
4.4.1.11	p	12
4.4.1.12	r	12
4.4.1.13	r0	12
4.4.1.14	relres	12
4.4.1.15	relres_base	12
4.4.1.16	res	12
4.4.1.17	rho	12
4.4.1.18	rho_old	12
4.4.1.19	s	12
4.4.1.20	t	12
4.4.1.21	tol_abs	12
4.4.1.22	tol_rel	12
4.4.1.23	v	13
4.4.1.24	x	13
4.4.1.25	y	13
4.4.1.26	z	13
4.5	CGS_DATA Struct Reference	13
4.5.1	Member Data Documentation	13
4.5.1.1	alpha	13
4.5.1.2	bestres	13
4.5.1.3	bestx	13
4.5.1.4	beta	13
4.5.1.5	breakdown	14
4.5.1.6	c	14
4.5.1.7	iter	14

4.5.1.8	maxit	14
4.5.1.9	Output	14
4.5.1.10	p	14
4.5.1.11	r	14
4.5.1.12	r0	14
4.5.1.13	relres	14
4.5.1.14	relres_base	14
4.5.1.15	res	14
4.5.1.16	rho	14
4.5.1.17	sigma	14
4.5.1.18	tol_abs	14
4.5.1.19	tol_rel	14
4.5.1.20	u	14
4.5.1.21	v	14
4.5.1.22	w	14
4.5.1.23	x	14
4.5.1.24	z	14
4.6	Document Class Reference	14
4.6.1	Constructor & Destructor Documentation	16
4.6.1.1	Document	16
4.6.1.2	~Document	16
4.6.1.3	Document	16
4.6.1.4	Document	16
4.6.1.5	Document	16
4.6.1.6	Document	16
4.6.1.7	Document	16
4.6.2	Member Function Documentation	16
4.6.2.1	addHeadKey	16
4.6.2.2	addPair	16
4.6.2.3	addPair	16
4.6.2.4	begin	16
4.6.2.5	begin	16
4.6.2.6	changeKey	16
4.6.2.7	clear	16
4.6.2.8	copyAnchor2Alias	16
4.6.2.9	DisplayContents	16
4.6.2.10	end	16
4.6.2.11	end	16
4.6.2.12	getAlias	16
4.6.2.13	getAnchoredHeader	16

4.6.2.14	getDataMap	16
4.6.2.15	getHeader	16
4.6.2.16	getHeadFromSubAlias	16
4.6.2.17	getHeadMap	16
4.6.2.18	getName	16
4.6.2.19	getState	17
4.6.2.20	isAlias	17
4.6.2.21	isAnchor	17
4.6.2.22	operator()	17
4.6.2.23	operator()	17
4.6.2.24	operator=	17
4.6.2.25	operator[]	17
4.6.2.26	operator[]	17
4.6.2.27	resetKeys	17
4.6.2.28	revalidateAllKeys	17
4.6.2.29	setAlias	17
4.6.2.30	setName	17
4.6.2.31	setNameAliasPair	17
4.6.2.32	setState	17
4.6.2.33	size	17
4.6.3	Member Data Documentation	17
4.6.3.1	Head_Map	17
4.7	DOGFISH_DATA Struct Reference	17
4.7.1	Member Data Documentation	18
4.7.1.1	DirichletBC	18
4.7.1.2	end_time	18
4.7.1.3	eval_DI	18
4.7.1.4	eval_kf	18
4.7.1.5	eval_qs	18
4.7.1.6	eval_R	18
4.7.1.7	fiber_diameter	18
4.7.1.8	fiber_length	18
4.7.1.9	finch_dat	18
4.7.1.10	NonLinear	18
4.7.1.11	NumComp	18
4.7.1.12	OutputFile	18
4.7.1.13	param_dat	18
4.7.1.14	Print2Console	18
4.7.1.15	Print2File	18
4.7.1.16	t_counter	18

4.7.1.17	t_print	18
4.7.1.18	time	18
4.7.1.19	time_old	18
4.7.1.20	total_sorption	19
4.7.1.21	total_sorption_old	19
4.7.1.22	total_steps	19
4.7.1.23	user_data	19
4.8	DOGFISH_PARAM Struct Reference	19
4.8.1	Member Data Documentation	19
4.8.1.1	film_transfer_coeff	19
4.8.1.2	initial_sorption	19
4.8.1.3	intraparticle_diffusion	19
4.8.1.4	sorbed_molefraction	19
4.8.1.5	species	19
4.8.1.6	surface_concentration	19
4.9	EX01_DATA Struct Reference	19
4.9.1	Member Data Documentation	20
4.9.1.1	b	20
4.9.1.2	M	20
4.10	EX02_DATA Struct Reference	20
4.10.1	Member Data Documentation	20
4.10.1.1	b	20
4.10.1.2	M	20
4.11	EX04_DATA Struct Reference	20
4.11.1	Member Data Documentation	20
4.11.1.1	b	20
4.11.1.2	M	20
4.12	EX09_DATA Struct Reference	21
4.12.1	Member Data Documentation	21
4.12.1.1	h	21
4.12.1.2	k	21
4.12.1.3	M	21
4.12.1.4	N	21
4.12.1.5	p	21
4.12.1.6	s	21
4.12.1.7	x	21
4.13	EX15_DATA Struct Reference	21
4.13.1	Member Data Documentation	21
4.13.1.1	b	21
4.13.1.2	m	21

4.13.1.3 N	22
4.14 FINCH_DATA Struct Reference	22
4.14.1 Member Data Documentation	24
4.14.1.1 beta	24
4.14.1.2 callroutine	24
4.14.1.3 CC_E	24
4.14.1.4 CC_I	24
4.14.1.5 CheckMass	24
4.14.1.6 CL_E	24
4.14.1.7 CL_I	24
4.14.1.8 CN	24
4.14.1.9 CR_E	24
4.14.1.10 CR_I	24
4.14.1.11 d	24
4.14.1.12 DIC	24
4.14.1.13 Dirichlet	24
4.14.1.14 discretize	24
4.14.1.15 Dn	24
4.14.1.16 Dnp1	24
4.14.1.17 Do	24
4.14.1.18 dt	24
4.14.1.19 dt_old	24
4.14.1.20 dz	24
4.14.1.21 evalprecon	24
4.14.1.22 evalres	24
4.14.1.23 ExplicitFlux	24
4.14.1.24 fC_E	25
4.14.1.25 fC_I	25
4.14.1.26 fL_E	25
4.14.1.27 fL_I	25
4.14.1.28 Fn	25
4.14.1.29 Fnp1	25
4.14.1.30 fR_E	25
4.14.1.31 fR_I	25
4.14.1.32 gE	25
4.14.1.33 gl	25
4.14.1.34 Iterative	25
4.14.1.35 kfn	25
4.14.1.36 kfnp1	25
4.14.1.37 kIC	25

4.14.1.38 kn	25
4.14.1.39 knp1	25
4.14.1.40 ko	25
4.14.1.41 L	25
4.14.1.42 lambda_E	25
4.14.1.43 lambda_I	25
4.14.1.44 LN	25
4.14.1.45 max_iter	25
4.14.1.46 ME	25
4.14.1.47 MI	25
4.14.1.48 NE	25
4.14.1.49 NI	25
4.14.1.50 nl_method	25
4.14.1.51 NormTrack	25
4.14.1.52 OE	26
4.14.1.53 OI	26
4.14.1.54 param_data	26
4.14.1.55 picard_dat	26
4.14.1.56 pjfnk_dat	26
4.14.1.57 pres	26
4.14.1.58 res	26
4.14.1.59 resettime	26
4.14.1.60 RIC	26
4.14.1.61 Rn	26
4.14.1.62 Rnp1	26
4.14.1.63 Ro	26
4.14.1.64 s	26
4.14.1.65 setbcs	26
4.14.1.66 setic	26
4.14.1.67 setparams	26
4.14.1.68 setpostprocess	26
4.14.1.69 setpreprocess	26
4.14.1.70 settime	26
4.14.1.71 Sn	26
4.14.1.72 Snp1	26
4.14.1.73 solve	26
4.14.1.74 SteadyState	26
4.14.1.75 T	26
4.14.1.76 t	26
4.14.1.77 t_old	26

4.14.1.78 tol_abs	26
4.14.1.79 tol_rel	26
4.14.1.80 total_iter	27
4.14.1.81 u_star	27
4.14.1.82 uAvg	27
4.14.1.83 uAvg_old	27
4.14.1.84 ubest	27
4.14.1.85 uIC	27
4.14.1.86 un	27
4.14.1.87 unm1	27
4.14.1.88 unp1	27
4.14.1.89 uo	27
4.14.1.90 Update	27
4.14.1.91 uT	27
4.14.1.92 uT_old	27
4.14.1.93 uz_l_E	27
4.14.1.94 uz_l_I	27
4.14.1.95 uz_lm1_E	27
4.14.1.96 uz_lm1_I	27
4.14.1.97 uz_lp1_E	27
4.14.1.98 uz_lp1_I	27
4.14.1.99 vIC	27
4.14.1.100vn	27
4.14.1.101vnp1	27
4.14.1.102vo	27
4.15 GCR_DATA Struct Reference	27
4.15.1 Member Data Documentation	28
4.15.1.1 alpha	28
4.15.1.2 bestres	28
4.15.1.3 bestx	28
4.15.1.4 beta	28
4.15.1.5 breakdown	28
4.15.1.6 c	28
4.15.1.7 c_temp	28
4.15.1.8 iter_inner	28
4.15.1.9 iter_outer	28
4.15.1.10 maxit	28
4.15.1.11 Output	28
4.15.1.12 r	28
4.15.1.13 relres	28

4.15.1.14 relres_base	28
4.15.1.15 res	29
4.15.1.16 restart	29
4.15.1.17 tol_abs	29
4.15.1.18 tol_rel	29
4.15.1.19 total_iter	29
4.15.1.20 transpose_dat	29
4.15.1.21 u	29
4.15.1.22 u_temp	29
4.15.1.23 x	29
4.16 GMRESLP_DATA Struct Reference	29
4.16.1 Member Data Documentation	29
4.16.1.1 arnoldi_dat	29
4.16.1.2 bestres	29
4.16.1.3 bestx	29
4.16.1.4 iter	29
4.16.1.5 maxit	30
4.16.1.6 Output	30
4.16.1.7 r	30
4.16.1.8 relres	30
4.16.1.9 relres_base	30
4.16.1.10 res	30
4.16.1.11 restart	30
4.16.1.12 steps	30
4.16.1.13 tol_abs	30
4.16.1.14 tol_rel	30
4.16.1.15 x	30
4.17 GMRESR_DATA Struct Reference	30
4.17.1 Member Data Documentation	31
4.17.1.1 arg	31
4.17.1.2 gcr_abs_tol	31
4.17.1.3 gcr_dat	31
4.17.1.4 gcr_maxit	31
4.17.1.5 GCR_Output	31
4.17.1.6 gcr_rel_tol	31
4.17.1.7 gcr_restart	31
4.17.1.8 gmres_dat	31
4.17.1.9 gmres_maxit	31
4.17.1.10 GMRES_Output	31
4.17.1.11 gmres_restart	31

4.17.1.12 gmres_tol	31
4.17.1.13 iter_inner	31
4.17.1.14 iter_outer	31
4.17.1.15 matvec	31
4.17.1.16 matvec_data	31
4.17.1.17 N	31
4.17.1.18 term_precon	31
4.17.1.19 terminal_precon	31
4.17.1.20 total_iter	31
4.18 GMRESRP_DATA Struct Reference	31
4.18.1 Member Data Documentation	32
4.18.1.1 bestres	32
4.18.1.2 bestx	32
4.18.1.3 e0	32
4.18.1.4 e0_bar	32
4.18.1.5 H	32
4.18.1.6 H_bar	32
4.18.1.7 iter_inner	32
4.18.1.8 iter_outer	32
4.18.1.9 iter_total	32
4.18.1.10 maxit	32
4.18.1.11 Output	32
4.18.1.12 r	32
4.18.1.13 relres	32
4.18.1.14 relres_base	32
4.18.1.15 res	33
4.18.1.16 restart	33
4.18.1.17 sum	33
4.18.1.18 tol_abs	33
4.18.1.19 tol_rel	33
4.18.1.20 v	33
4.18.1.21 Vk	33
4.18.1.22 w	33
4.18.1.23 x	33
4.18.1.24 y	33
4.19 GPAST_DATA Struct Reference	33
4.19.1 Member Data Documentation	33
4.19.1.1 gama_inf	33
4.19.1.2 He	33
4.19.1.3 Plo	33

4.19.1.4	po	33
4.19.1.5	poi	33
4.19.1.6	present	34
4.19.1.7	q	34
4.19.1.8	qo	34
4.19.1.9	x	34
4.19.1.10	y	34
4.20	GSTA_DATA Struct Reference	34
4.20.1	Member Data Documentation	34
4.20.1.1	dHo	34
4.20.1.2	dSo	34
4.20.1.3	m	34
4.20.1.4	qmax	34
4.21	GSTA_OPT_DATA Struct Reference	34
4.21.1	Member Data Documentation	35
4.21.1.1	all_pars	35
4.21.1.2	best_par	35
4.21.1.3	Fobj	35
4.21.1.4	iso	35
4.21.1.5	Kno	35
4.21.1.6	n_par	35
4.21.1.7	norms	35
4.21.1.8	opt_qmax	35
4.21.1.9	P	35
4.21.1.10	q	35
4.21.1.11	qmax	35
4.21.1.12	total_eval	35
4.22	Header Class Reference	35
4.22.1	Constructor & Destructor Documentation	37
4.22.1.1	Header	37
4.22.1.2	~Header	37
4.22.1.3	Header	37
4.22.1.4	Header	37
4.22.1.5	Header	37
4.22.1.6	Header	37
4.22.1.7	Header	37
4.22.2	Member Function Documentation	37
4.22.2.1	addPair	37
4.22.2.2	addPair	37
4.22.2.3	addSubKey	37

4.22.2.4	begin	37
4.22.2.5	begin	37
4.22.2.6	changeKey	37
4.22.2.7	clear	37
4.22.2.8	copyAnchor2Alias	37
4.22.2.9	DisplayContents	37
4.22.2.10	end	37
4.22.2.11	end	37
4.22.2.12	getAlias	37
4.22.2.13	getAnchoredSub	37
4.22.2.14	getDataMap	37
4.22.2.15	getName	37
4.22.2.16	getState	37
4.22.2.17	getSubHeader	37
4.22.2.18	getSubMap	37
4.22.2.19	isAlias	38
4.22.2.20	isAnchor	38
4.22.2.21	operator()	38
4.22.2.22	operator()	38
4.22.2.23	operator=	38
4.22.2.24	operator[]	38
4.22.2.25	operator[]	38
4.22.2.26	resetKeys	38
4.22.2.27	setAlias	38
4.22.2.28	setName	38
4.22.2.29	setNameAliasPair	38
4.22.2.30	setState	38
4.22.2.31	size	38
4.22.3	Member Data Documentation	38
4.22.3.1	Sub_Map	38
4.23	KeyValueMap Class Reference	38
4.23.1	Constructor & Destructor Documentation	39
4.23.1.1	KeyValueMap	39
4.23.1.2	~KeyValueMap	39
4.23.1.3	KeyValueMap	39
4.23.1.4	KeyValueMap	39
4.23.1.5	KeyValueMap	39
4.23.2	Member Function Documentation	39
4.23.2.1	addKey	39
4.23.2.2	addPair	39

4.23.2.3	addPair	39
4.23.2.4	addPair	39
4.23.2.5	assertType	40
4.23.2.6	begin	40
4.23.2.7	begin	40
4.23.2.8	clear	40
4.23.2.9	DisplayMap	40
4.23.2.10	editValue4Key	40
4.23.2.11	editValue4Key	40
4.23.2.12	end	40
4.23.2.13	end	40
4.23.2.14	findAllTypes	40
4.23.2.15	findType	40
4.23.2.16	getBool	40
4.23.2.17	getDouble	40
4.23.2.18	getInt	40
4.23.2.19	getMap	40
4.23.2.20	getPair	40
4.23.2.21	getString	40
4.23.2.22	getType	40
4.23.2.23	getValue	40
4.23.2.24	operator=	40
4.23.2.25	operator[]	40
4.23.2.26	operator[]	40
4.23.2.27	size	40
4.23.3	Member Data Documentation	40
4.23.3.1	Key_Value	40
4.24	MAGPIE_DATA Struct Reference	41
4.24.1	Member Data Documentation	41
4.24.1.1	gpast_dat	41
4.24.1.2	gsta_dat	41
4.24.1.3	mspd_dat	41
4.24.1.4	sys_dat	41
4.25	MassBalance Class Reference	41
4.25.1	Constructor & Destructor Documentation	42
4.25.1.1	MassBalance	42
4.25.1.2	~MassBalance	42
4.25.2	Member Function Documentation	42
4.25.2.1	Display_Info	42
4.25.2.2	Eval_Residual	42

4.25.2.3	Get_Delta	42
4.25.2.4	Get_Name	42
4.25.2.5	Get_TotalConcentration	42
4.25.2.6	Initialize_List	42
4.25.2.7	Set_Delta	42
4.25.2.8	Set_Name	42
4.25.2.9	Set_TotalConcentration	42
4.25.2.10	Sum_Delta	42
4.25.3	Member Data Documentation	42
4.25.3.1	Delta	42
4.25.3.2	List	42
4.25.3.3	Name	42
4.25.3.4	TotalConcentration	42
4.26	MasterSpeciesList Class Reference	42
4.26.1	Constructor & Destructor Documentation	43
4.26.1.1	MasterSpeciesList	43
4.26.1.2	~MasterSpeciesList	43
4.26.1.3	MasterSpeciesList	43
4.26.2	Member Function Documentation	43
4.26.2.1	alkalinity	43
4.26.2.2	charge	43
4.26.2.3	DisplayAll	43
4.26.2.4	DisplayConcentrations	44
4.26.2.5	DisplayInfo	44
4.26.2.6	Eval_ChargeResidual	44
4.26.2.7	get_index	44
4.26.2.8	get_species	44
4.26.2.9	list_size	44
4.26.2.10	operator=	44
4.26.2.11	set_alkalinity	44
4.26.2.12	set_list_size	44
4.26.2.13	set_species	44
4.26.2.14	set_species	44
4.26.2.15	speciesName	44
4.26.3	Member Data Documentation	44
4.26.3.1	residual_alkalinity	44
4.26.3.2	size	44
4.26.3.3	species	44
4.27	Matrix< T > Class Template Reference	44
4.27.1	Constructor & Destructor Documentation	46

4.27.1.1	Matrix	46
4.27.1.2	Matrix	46
4.27.1.3	Matrix	46
4.27.1.4	~Matrix	46
4.27.2	Member Function Documentation	46
4.27.2.1	adjoint	46
4.27.2.2	cofactor	46
4.27.2.3	columnExtend	46
4.27.2.4	columnExtract	46
4.27.2.5	columnProjection	46
4.27.2.6	columnReplace	46
4.27.2.7	columns	46
4.27.2.8	columnShrink	46
4.27.2.9	columnVectorFill	46
4.27.2.10	ConstantICFill	46
4.27.2.11	determinate	46
4.27.2.12	diagonalSolve	46
4.27.2.13	dirichletBCFill	46
4.27.2.14	Display	46
4.27.2.15	edit	46
4.27.2.16	inner_product	46
4.27.2.17	IntegralAvg	46
4.27.2.18	IntegralTotal	46
4.27.2.19	inverse	46
4.27.2.20	ladshawSolve	46
4.27.2.21	lowerHessenberg2Triangular	46
4.27.2.22	lowerHessenbergSolve	47
4.27.2.23	lowerTriangularSolve	47
4.27.2.24	naturalLaplacian3D	47
4.27.2.25	norm	47
4.27.2.26	operator()	47
4.27.2.27	operator()	47
4.27.2.28	operator*	47
4.27.2.29	operator*	47
4.27.2.30	operator+	47
4.27.2.31	operator-	47
4.27.2.32	operator/	47
4.27.2.33	operator=	47
4.27.2.34	rowExtend	47
4.27.2.35	rowExtract	47

4.27.2.36 rowReplace	47
4.27.2.37 rows	47
4.27.2.38 rowShrink	47
4.27.2.39 set_size	47
4.27.2.40 SolnTransform	47
4.27.2.41 sphericalAvg	47
4.27.2.42 sphericalBCFill	47
4.27.2.43 sum	47
4.27.2.44 transpose	47
4.27.2.45 transpose_multiply	47
4.27.2.46 tridiagonalFill	47
4.27.2.47 tridiagonalSolve	47
4.27.2.48 tridiagonalVectorFill	48
4.27.2.49 upperHessenberg2Triangular	48
4.27.2.50 upperHessenbergSolve	48
4.27.2.51 upperTriangularSolve	48
4.27.2.52 zeros	48
4.27.3 Member Data Documentation	48
4.27.3.1 Data	48
4.27.3.2 num_cols	48
4.27.3.3 num_rows	48
4.28 Mechanism Class Reference	48
4.28.1 Member Data Documentation	48
4.28.1.1 List	48
4.28.1.2 reactions	48
4.28.1.3 species_index	48
4.28.1.4 weight	48
4.29 MIXED_GAS Struct Reference	48
4.29.1 Member Data Documentation	49
4.29.1.1 binary_diffusion	49
4.29.1.2 char_length	49
4.29.1.3 CheckMolefractions	49
4.29.1.4 gas_temperature	49
4.29.1.5 kinematic_viscosity	49
4.29.1.6 molefraction	49
4.29.1.7 N	49
4.29.1.8 Reynolds	49
4.29.1.9 species_dat	49
4.29.1.10 total_density	49
4.29.1.11 total_dyn_vis	49

4.29.1.12	total_molecular_weight	49
4.29.1.13	total_pressure	49
4.29.1.14	total_specific_heat	49
4.29.1.15	velocity	49
4.30	Molecule Class Reference	50
4.30.1	Constructor & Destructor Documentation	51
4.30.1.1	Molecule	51
4.30.1.2	~Molecule	51
4.30.1.3	Molecule	51
4.30.2	Member Function Documentation	51
4.30.2.1	calculateAvgOxiState	51
4.30.2.2	Charge	51
4.30.2.3	DisplayInfo	51
4.30.2.4	editAllOxidationStates	51
4.30.2.5	editCharge	51
4.30.2.6	editEnergy	51
4.30.2.7	editEnthalpy	51
4.30.2.8	editEntropy	51
4.30.2.9	editHS	51
4.30.2.10	editOneOxidationState	51
4.30.2.11	Energy	51
4.30.2.12	Enthalpy	51
4.30.2.13	Entropy	52
4.30.2.14	HaveEnergy	52
4.30.2.15	HaveHS	52
4.30.2.16	isRegistered	52
4.30.2.17	MolarWeight	52
4.30.2.18	MolecularFormula	52
4.30.2.19	MoleculeName	52
4.30.2.20	MoleculePhase	52
4.30.2.21	recalculateMolarWeight	52
4.30.2.22	Register	52
4.30.2.23	Register	52
4.30.2.24	removeAllAtoms	52
4.30.2.25	removeOneAtom	52
4.30.2.26	setFormula	52
4.30.2.27	setMolarWeigth	52
4.30.3	Member Data Documentation	52
4.30.3.1	atoms	52
4.30.3.2	charge	52

4.30.3.3	formation_energy	52
4.30.3.4	formation_enthalpy	52
4.30.3.5	formation_entropy	52
4.30.3.6	Formula	52
4.30.3.7	haveG	52
4.30.3.8	haveHS	52
4.30.3.9	molar_weight	52
4.30.3.10	Name	52
4.30.3.11	Phase	52
4.30.3.12	registered	53
4.31	MONKFISH_DATA Struct Reference	53
4.31.1	Member Data Documentation	54
4.31.1.1	avg_fiber_density	54
4.31.1.2	DirichletBC	54
4.31.1.3	dog_dat	54
4.31.1.4	domain_diameter	54
4.31.1.5	end_time	54
4.31.1.6	eval_ads	54
4.31.1.7	eval_Cex	54
4.31.1.8	eval_Dex	54
4.31.1.9	eval_eps	54
4.31.1.10	eval_kf	54
4.31.1.11	eval_Ret	54
4.31.1.12	eval_rho	54
4.31.1.13	finch_dat	54
4.31.1.14	haveMinMax	54
4.31.1.15	level	54
4.31.1.16	max_fiber_density	54
4.31.1.17	max_porosity	54
4.31.1.18	min_fiber_density	54
4.31.1.19	min_porosity	54
4.31.1.20	MultiScale	54
4.31.1.21	NonLinear	54
4.31.1.22	NumComp	54
4.31.1.23	Output	54
4.31.1.24	param_dat	54
4.31.1.25	Print2Console	54
4.31.1.26	Print2File	54
4.31.1.27	single_fiber_density	54
4.31.1.28	t_counter	55

4.31.1.29 t_print	55
4.31.1.30 time	55
4.31.1.31 time_old	55
4.31.1.32 total_sorption	55
4.31.1.33 total_sorption_old	55
4.31.1.34 total_steps	55
4.31.1.35 user_data	55
4.32 MONKFISH_PARAM Struct Reference	55
4.32.1 Member Data Documentation	55
4.32.1.1 avg_sorption	55
4.32.1.2 avg_sorption_old	55
4.32.1.3 exterior_concentration	55
4.32.1.4 exterior_transfer_coeff	55
4.32.1.5 film_transfer_coeff	55
4.32.1.6 initial_sorption	55
4.32.1.7 interparticle_diffusion	55
4.32.1.8 intraparticle_diffusion	56
4.32.1.9 sorbed_molefraction	56
4.32.1.10 sorption_bc	56
4.32.1.11 species	56
4.33 mSPD_DATA Struct Reference	56
4.33.1 Member Data Documentation	56
4.33.1.1 eMax	56
4.33.1.2 eta	56
4.33.1.3 gama	56
4.33.1.4 s	56
4.33.1.5 v	56
4.34 NUM_JAC_DATA Struct Reference	56
4.34.1 Member Data Documentation	57
4.34.1.1 dxj	57
4.34.1.2 eps	57
4.34.1.3 Fx	57
4.34.1.4 Fxp	57
4.35 OPTRANS_DATA Struct Reference	57
4.35.1 Member Data Documentation	57
4.35.1.1 Ai	57
4.35.1.2 li	57
4.36 PCG_DATA Struct Reference	57
4.36.1 Member Data Documentation	58
4.36.1.1 alpha	58

4.36.1.2	Ap	58
4.36.1.3	bestres	58
4.36.1.4	bestx	58
4.36.1.5	beta	58
4.36.1.6	iter	58
4.36.1.7	maxit	58
4.36.1.8	Output	58
4.36.1.9	p	58
4.36.1.10	r	58
4.36.1.11	r_old	58
4.36.1.12	relres	58
4.36.1.13	relres_base	58
4.36.1.14	res	58
4.36.1.15	tol_abs	58
4.36.1.16	tol_rel	58
4.36.1.17	x	58
4.36.1.18	z	58
4.36.1.19	z_old	58
4.37	PeriodicTable Class Reference	58
4.37.1	Constructor & Destructor Documentation	59
4.37.1.1	PeriodicTable	59
4.37.1.2	~PeriodicTable	59
4.37.1.3	PeriodicTable	59
4.37.1.4	PeriodicTable	59
4.37.1.5	PeriodicTable	59
4.37.2	Member Function Documentation	59
4.37.2.1	DisplayTable	59
4.37.3	Member Data Documentation	59
4.37.3.1	number_elements	59
4.37.3.2	Table	59
4.38	PICARD_DATA Struct Reference	60
4.38.1	Member Data Documentation	60
4.38.1.1	bestres	60
4.38.1.2	bestx	60
4.38.1.3	iter	60
4.38.1.4	maxit	60
4.38.1.5	Output	60
4.38.1.6	r	60
4.38.1.7	relres	60
4.38.1.8	relres_base	60

4.38.1.9	res	60
4.38.1.10	tol_abs	60
4.38.1.11	tol_rel	60
4.38.1.12	x0	60
4.39	PJFNK_DATA Struct Reference	60
4.39.1	Member Data Documentation	61
4.39.1.1	backtrack_dat	61
4.39.1.2	bestx	61
4.39.1.3	bicgstab_dat	61
4.39.1.4	Bounce	61
4.39.1.5	cgs_dat	61
4.39.1.6	eps	61
4.39.1.7	F	61
4.39.1.8	funeval	61
4.39.1.9	Fv	62
4.39.1.10	gcr_dat	62
4.39.1.11	gmreslp_dat	62
4.39.1.12	gmresr_dat	62
4.39.1.13	gmresrp_dat	62
4.39.1.14	l_iter	62
4.39.1.15	L_Output	62
4.39.1.16	lin_tol_abs	62
4.39.1.17	lin_tol_rel	62
4.39.1.18	linear_solver	62
4.39.1.19	LineSearch	62
4.39.1.20	nl_bestres	62
4.39.1.21	nl_iter	62
4.39.1.22	nl_maxit	62
4.39.1.23	NL_Output	62
4.39.1.24	nl_relres	62
4.39.1.25	nl_res	62
4.39.1.26	nl_res_base	62
4.39.1.27	nl_tol_abs	62
4.39.1.28	nl_tol_rel	62
4.39.1.29	pcg_dat	62
4.39.1.30	precon	62
4.39.1.31	precon_data	62
4.39.1.32	res_data	62
4.39.1.33	v	62
4.39.1.34	x	62

4.40	Precipitation Class Reference	63
4.41	PURE_GAS Struct Reference	63
4.41.1	Member Data Documentation	63
4.41.1.1	density	63
4.41.1.2	dynamic_viscosity	63
4.41.1.3	molecular_diffusion	63
4.41.1.4	molecular_weight	63
4.41.1.5	Schmidt	63
4.41.1.6	specific_heat	63
4.41.1.7	Sutherland_Const	63
4.41.1.8	Sutherland_Temp	64
4.41.1.9	Sutherland_Viscosity	64
4.42	Reaction Class Reference	64
4.42.1	Constructor & Destructor Documentation	65
4.42.1.1	Reaction	65
4.42.1.2	~Reaction	65
4.42.2	Member Function Documentation	65
4.42.2.1	calculateEnergies	65
4.42.2.2	calculateEquilibrium	65
4.42.2.3	checkSpeciesEnergies	65
4.42.2.4	Display_Info	65
4.42.2.5	Eval_Residual	65
4.42.2.6	Get_Energy	65
4.42.2.7	Get_Enthalpy	65
4.42.2.8	Get_Entropy	65
4.42.2.9	Get_Equilibrium	65
4.42.2.10	Get_Stoichiometric	65
4.42.2.11	haveEquilibrium	65
4.42.2.12	Initialize_List	65
4.42.2.13	Set_Energy	65
4.42.2.14	Set_Enthalpy	65
4.42.2.15	Set_EnthalpyANDEntropy	65
4.42.2.16	Set_Entropy	65
4.42.2.17	Set_Equilibrium	65
4.42.2.18	Set_Stoichiometric	65
4.42.3	Member Data Documentation	65
4.42.3.1	CanCalcG	65
4.42.3.2	CanCalcHS	66
4.42.3.3	energy	66
4.42.3.4	enthalpy	66

4.42.3.5	entropy	66
4.42.3.6	Equilibrium	66
4.42.3.7	HaveEquil	66
4.42.3.8	HaveG	66
4.42.3.9	HaveHS	66
4.42.3.10	List	66
4.42.3.11	Stoichiometric	66
4.43	SCOPSOWL_DATA Struct Reference	66
4.43.1	Member Data Documentation	67
4.43.1.1	binder_fraction	67
4.43.1.2	binder_poresize	67
4.43.1.3	binder_porosity	67
4.43.1.4	char_macro	67
4.43.1.5	char_micro	67
4.43.1.6	coord_macro	67
4.43.1.7	coord_micro	67
4.43.1.8	crystal_radius	67
4.43.1.9	DirichletBC	67
4.43.1.10	eval_ads	67
4.43.1.11	eval_diff	67
4.43.1.12	eval_kf	67
4.43.1.13	eval_retard	67
4.43.1.14	eval_surfDiff	67
4.43.1.15	finch_dat	67
4.43.1.16	gas_dat	67
4.43.1.17	gas_temperature	67
4.43.1.18	gas_velocity	68
4.43.1.19	Heterogeneous	68
4.43.1.20	level	68
4.43.1.21	magpie_dat	68
4.43.1.22	NonLinear	68
4.43.1.23	OutputFile	68
4.43.1.24	param_dat	68
4.43.1.25	pellet_density	68
4.43.1.26	pellet_radius	68
4.43.1.27	Print2Console	68
4.43.1.28	Print2File	68
4.43.1.29	sim_time	68
4.43.1.30	skua_dat	68
4.43.1.31	SurfDiff	68

4.43.1.32 t	68
4.43.1.33 t_counter	68
4.43.1.34 t_old	68
4.43.1.35 t_print	68
4.43.1.36 tempy	68
4.43.1.37 total_pressure	68
4.43.1.38 total_steps	68
4.43.1.39 user_data	68
4.43.1.40 y	68
4.44 SCOPSOWL_OPT_DATA Struct Reference	68
4.44.1 Member Data Documentation	69
4.44.1.1 abs_tol_bias	69
4.44.1.2 adsorb_index	69
4.44.1.3 CompareFile	69
4.44.1.4 current_equil	69
4.44.1.5 current_points	69
4.44.1.6 current_press	69
4.44.1.7 current_temp	69
4.44.1.8 diffusion_type	69
4.44.1.9 e_norm	69
4.44.1.10 e_norm_old	70
4.44.1.11 evaluation	70
4.44.1.12 f_bias	70
4.44.1.13 f_bias_old	70
4.44.1.14 max_bias	70
4.44.1.15 max_guess_iter	70
4.44.1.16 min_bias	70
4.44.1.17 num_curves	70
4.44.1.18 num_params	70
4.44.1.19 Optimize	70
4.44.1.20 owl_dat	70
4.44.1.21 param_guess	70
4.44.1.22 param_guess_old	70
4.44.1.23 ParamFile	70
4.44.1.24 q_data	70
4.44.1.25 q_sim	70
4.44.1.26 rel_tol_norm	70
4.44.1.27 Rough	70
4.44.1.28 simulation_equil	70
4.44.1.29 t	70

4.44.1.30 total_eval	70
4.44.1.31 y_base	70
4.45 SCOPSOWL_PARAM_DATA Struct Reference	70
4.45.1 Member Data Documentation	71
4.45.1.1 activation_energy	71
4.45.1.2 Adsorbable	71
4.45.1.3 affinity	71
4.45.1.4 dq_dc	71
4.45.1.5 dq_dco	71
4.45.1.6 film_transfer	71
4.45.1.7 pore_diffusion	71
4.45.1.8 qAvg	71
4.45.1.9 qAvg_old	71
4.45.1.10 qIntegralAvg	71
4.45.1.11 qIntegralAvg_old	71
4.45.1.12 qo	71
4.45.1.13 Qst	71
4.45.1.14 Qst_old	71
4.45.1.15 QstAvg	71
4.45.1.16 QstAvg_old	72
4.45.1.17 Qsto	72
4.45.1.18 ref_diffusion	72
4.45.1.19 ref_pressure	72
4.45.1.20 ref_temperature	72
4.45.1.21 speciesName	72
4.45.1.22 xIC	72
4.46 SHARK_DATA Struct Reference	72
4.46.1 Member Data Documentation	73
4.46.1.1 act_fun	73
4.46.1.2 activity_data	73
4.46.1.3 activity_new	73
4.46.1.4 activity_old	73
4.46.1.5 Conc_new	73
4.46.1.6 Conc_old	73
4.46.1.7 Console_Output	73
4.46.1.8 const_pH	73
4.46.1.9 Contains_pH	73
4.46.1.10 Contains_pOH	73
4.46.1.11 Converged	73
4.46.1.12 dielectric_const	73

4.46.1.13 dt	73
4.46.1.14 dt_min	74
4.46.1.15 EvalActivity	74
4.46.1.16 File_Output	74
4.46.1.17 lin_precon	74
4.46.1.18 MassBalanceList	74
4.46.1.19 MasterList	74
4.46.1.20 Newton_data	74
4.46.1.21 Norm	74
4.46.1.22 num_mbe	74
4.46.1.23 num_other	74
4.46.1.24 num_ssr	74
4.46.1.25 num_usr	74
4.46.1.26 numvar	74
4.46.1.27 other_data	74
4.46.1.28 OtherList	74
4.46.1.29 OutputFile	74
4.46.1.30 pH	74
4.46.1.31 pH_index	74
4.46.1.32 pOH_index	74
4.46.1.33 precon_data	74
4.46.1.34 ReactionList	74
4.46.1.35 Residual	74
4.46.1.36 residual_data	74
4.46.1.37 simulationtime	74
4.46.1.38 SpeciationCurve	74
4.46.1.39 steadystate	74
4.46.1.40 t_count	74
4.46.1.41 t_out	74
4.46.1.42 temperature	75
4.46.1.43 time	75
4.46.1.44 time_old	75
4.46.1.45 TimeAdaptivity	75
4.46.1.46 timesteps	75
4.46.1.47 totalsteps	75
4.46.1.48 UnsteadyList	75
4.46.1.49 X_new	75
4.46.1.50 X_old	75
4.46.1.51 yaml_object	75
4.47 SKUA_DATA Struct Reference	75

4.47.1	Member Data Documentation	76
4.47.1.1	char_measure	76
4.47.1.2	coord	76
4.47.1.3	DirichletBC	76
4.47.1.4	eval_diff	76
4.47.1.5	eval_kf	76
4.47.1.6	finch_dat	76
4.47.1.7	gas_dat	76
4.47.1.8	gas_velocity	76
4.47.1.9	magpie_dat	76
4.47.1.10	NonLinear	76
4.47.1.11	OutputFile	76
4.47.1.12	param_dat	76
4.47.1.13	pellet_radius	76
4.47.1.14	Print2Console	76
4.47.1.15	Print2File	76
4.47.1.16	qTn	76
4.47.1.17	qTnp1	76
4.47.1.18	sim_time	76
4.47.1.19	t	76
4.47.1.20	t_counter	76
4.47.1.21	t_old	76
4.47.1.22	t_print	76
4.47.1.23	total_steps	76
4.47.1.24	user_data	76
4.47.1.25	y	76
4.48	SKUA_OPT_DATA Struct Reference	77
4.48.1	Member Data Documentation	77
4.48.1.1	abs_tol_bias	77
4.48.1.2	adsorb_index	77
4.48.1.3	CompareFile	77
4.48.1.4	current_equil	77
4.48.1.5	current_points	77
4.48.1.6	current_press	77
4.48.1.7	current_temp	77
4.48.1.8	diffusion_type	78
4.48.1.9	e_norm	78
4.48.1.10	e_norm_old	78
4.48.1.11	evaluation	78
4.48.1.12	f_bias	78

4.48.1.13 f_bias_old	78
4.48.1.14 max_bias	78
4.48.1.15 max_guess_iter	78
4.48.1.16 min_bias	78
4.48.1.17 num_curves	78
4.48.1.18 num_params	78
4.48.1.19 Optimize	78
4.48.1.20 param_guess	78
4.48.1.21 param_guess_old	78
4.48.1.22 ParamFile	78
4.48.1.23 q_data	78
4.48.1.24 q_sim	78
4.48.1.25 rel_tol_norm	78
4.48.1.26 Rough	78
4.48.1.27 simulation_equil	78
4.48.1.28 skua_dat	78
4.48.1.29 t	78
4.48.1.30 total_eval	78
4.48.1.31 y_base	78
4.49 SKUA_PARAM Struct Reference	78
4.49.1 Member Data Documentation	79
4.49.1.1 activation_energy	79
4.49.1.2 Adsorbable	79
4.49.1.3 affinity	79
4.49.1.4 film_transfer	79
4.49.1.5 Qstn	79
4.49.1.6 Qstnp1	79
4.49.1.7 ref_diffusion	79
4.49.1.8 ref_pressure	79
4.49.1.9 ref_temperature	79
4.49.1.10 speciesName	79
4.49.1.11 xIC	79
4.49.1.12 xn	79
4.49.1.13 xnp1	79
4.49.1.14 y_eff	79
4.50 Speciation_Test01_Data Struct Reference	79
4.50.1 Member Data Documentation	80
4.50.1.1 C	80
4.50.1.2 CT	80
4.50.1.3 Jacobian	80

4.50.1.4	logC	80
4.50.1.5	logKa1	80
4.50.1.6	logKa2	80
4.50.1.7	logKw	80
4.50.1.8	N	80
4.50.1.9	NaT	80
4.50.1.10	NumJac	80
4.50.1.11	x	80
4.51	SubHeader Class Reference	80
4.51.1	Constructor & Destructor Documentation	81
4.51.1.1	SubHeader	81
4.51.1.2	~SubHeader	81
4.51.1.3	SubHeader	81
4.51.1.4	SubHeader	81
4.51.1.5	SubHeader	81
4.51.1.6	SubHeader	81
4.51.2	Member Function Documentation	81
4.51.2.1	addPair	81
4.51.2.2	addPair	81
4.51.2.3	clear	82
4.51.2.4	DisplayContents	82
4.51.2.5	getAlias	82
4.51.2.6	getMap	82
4.51.2.7	getName	82
4.51.2.8	getState	82
4.51.2.9	isAlias	82
4.51.2.10	isAnchor	82
4.51.2.11	operator=	82
4.51.2.12	operator[]	82
4.51.2.13	operator[]	82
4.51.2.14	setAlias	82
4.51.2.15	setAlias	82
4.51.2.16	setName	82
4.51.2.17	setNameAliasPair	82
4.51.2.18	setState	82
4.51.3	Member Data Documentation	82
4.51.3.1	alias	82
4.51.3.2	Data_Map	82
4.51.3.3	name	82
4.51.3.4	state	82

4.52 SYSTEM_DATA Struct Reference	82
4.52.1 Member Data Documentation	83
4.52.1.1 As	83
4.52.1.2 avg_norm	83
4.52.1.3 Carrier	83
4.52.1.4 I	83
4.52.1.5 Ideal	83
4.52.1.6 J	83
4.52.1.7 K	83
4.52.1.8 max_norm	83
4.52.1.9 N	83
4.52.1.10 Output	83
4.52.1.11 Par	83
4.52.1.12 PI	83
4.52.1.13 pi	83
4.52.1.14 PT	83
4.52.1.15 qT	83
4.52.1.16 Recover	83
4.52.1.17 Sys	83
4.52.1.18 T	84
4.52.1.19 total_eval	84
4.53 TRAJECTORY_DATA Struct Reference	84
4.53.1 Member Data Documentation	85
4.53.1.1 a	85
4.53.1.2 A_separator	85
4.53.1.3 A_wire	85
4.53.1.4 b	85
4.53.1.5 B0	85
4.53.1.6 beta	85
4.53.1.7 Cap	85
4.53.1.8 chi_p	85
4.53.1.9 dt	85
4.53.1.10 dX	85
4.53.1.11 dY	85
4.53.1.12 eta	85
4.53.1.13 H	85
4.53.1.14 H0	85
4.53.1.15 Hamaker	85
4.53.1.16 k	85
4.53.1.17 L	85

4.53.1.18 L_wire	85
4.53.1.19 M	85
4.53.1.20 m_rand	85
4.53.1.21 mp	85
4.53.1.22 Ms	85
4.53.1.23 mu_0	85
4.53.1.24 n_rand	86
4.53.1.25 POL	86
4.53.1.26 porosity	86
4.53.1.27 q_bar	86
4.53.1.28 Q_in	86
4.53.1.29 rho_f	86
4.53.1.30 rho_p	86
4.53.1.31 Rs	86
4.53.1.32 s_rand	86
4.53.1.33 sigma_m	86
4.53.1.34 sigma_n	86
4.53.1.35 sigma_v	86
4.53.1.36 sigma_vz	86
4.53.1.37 sigma_z	86
4.53.1.38 t_rand	86
4.53.1.39 Temp	86
4.53.1.40 V0	86
4.53.1.41 V_separator	86
4.53.1.42 V_wire	86
4.53.1.43 X	86
4.53.1.44 Y	86
4.53.1.45 Y_initial	86
4.54 UI_DATA Struct Reference	86
4.54.1 Member Data Documentation	87
4.54.1.1 argc	87
4.54.1.2 argv	87
4.54.1.3 BasicUI	87
4.54.1.4 count	87
4.54.1.5 Files	87
4.54.1.6 input_files	87
4.54.1.7 max	87
4.54.1.8 MissingArg	87
4.54.1.9 option	87
4.54.1.10 path	87

4.54.1.11 Path	87
4.54.1.12 user_input	87
4.54.1.13 value_type	87
4.55 UnsteadyPrecipitation Class Reference	87
4.56 UnsteadyReaction Class Reference	88
4.56.1 Constructor & Destructor Documentation	89
4.56.1.1 UnsteadyReaction	89
4.56.1.2 ~UnsteadyReaction	89
4.56.2 Member Function Documentation	89
4.56.2.1 calculateEnergies	89
4.56.2.2 calculateEquilibrium	89
4.56.2.3 calculateRate	89
4.56.2.4 checkSpeciesEnergies	90
4.56.2.5 Display_Info	90
4.56.2.6 Eval_IC_Residual	90
4.56.2.7 Eval_ReactionRate	90
4.56.2.8 Eval_Residual	90
4.56.2.9 Eval_Residual	90
4.56.2.10 Explicit_Eval	90
4.56.2.11 Get_ActivationEnergy	90
4.56.2.12 Get_Affinity	90
4.56.2.13 Get_Energy	90
4.56.2.14 Get_Enthalpy	90
4.56.2.15 Get_Entropy	90
4.56.2.16 Get_Equilibrium	90
4.56.2.17 Get_Forward	90
4.56.2.18 Get_ForwardRef	90
4.56.2.19 Get_InitialValue	90
4.56.2.20 Get_MaximumValue	90
4.56.2.21 Get_Reverse	90
4.56.2.22 Get_ReverseRef	90
4.56.2.23 Get_Species_Index	90
4.56.2.24 Get_Stoichiometric	90
4.56.2.25 Get_TimeStep	90
4.56.2.26 haveEquilibrium	90
4.56.2.27 haveRate	90
4.56.2.28 Initialize_List	90
4.56.2.29 Set_ActivationEnergy	90
4.56.2.30 Set_Affinity	90
4.56.2.31 Set_Energy	90

4.56.2.32 Set_Enthalpy	91
4.56.2.33 Set_EnthalpyANDEntropy	91
4.56.2.34 Set_Entropy	91
4.56.2.35 Set_Equilibrium	91
4.56.2.36 Set_Forward	91
4.56.2.37 Set_ForwardRef	91
4.56.2.38 Set_InitialValue	91
4.56.2.39 Set_MaximumValue	91
4.56.2.40 Set_Reverse	91
4.56.2.41 Set_ReverseRef	91
4.56.2.42 Set_Species_Index	91
4.56.2.43 Set_Species_Index	91
4.56.2.44 Set_Stoichiometric	91
4.56.2.45 Set_TimeStep	91
4.56.3 Member Data Documentation	91
4.56.3.1 activation_energy	91
4.56.3.2 forward_rate	91
4.56.3.3 forward_ref_rate	91
4.56.3.4 HaveForRef	91
4.56.3.5 HaveForward	91
4.56.3.6 HaveReverse	91
4.56.3.7 HaveRevRef	91
4.56.3.8 initial_value	91
4.56.3.9 max_value	91
4.56.3.10 reverse_rate	91
4.56.3.11 reverse_ref_rate	91
4.56.3.12 species_index	91
4.56.3.13 temperature_affinity	91
4.56.3.14 time_step	92
4.57 ValuePair Class Reference	92
4.57.1 Constructor & Destructor Documentation	92
4.57.1.1 ValuePair	92
4.57.1.2 ~ValuePair	92
4.57.1.3 ValuePair	92
4.57.1.4 ValuePair	92
4.57.1.5 ValuePair	92
4.57.2 Member Function Documentation	92
4.57.2.1 assertType	93
4.57.2.2 DisplayPair	93
4.57.2.3 editPair	93

4.57.2.4	editValue	93
4.57.2.5	findType	93
4.57.2.6	getBool	93
4.57.2.7	getDouble	93
4.57.2.8	getInt	93
4.57.2.9	getPair	93
4.57.2.10	getString	93
4.57.2.11	getType	93
4.57.2.12	getValue	93
4.57.2.13	operator=	93
4.57.3	Member Data Documentation	93
4.57.3.1	type	93
4.57.3.2	Value_Type	93
4.58	yaml_cpp_class Class Reference	93
4.58.1	Constructor & Destructor Documentation	94
4.58.1.1	yaml_cpp_class	94
4.58.1.2	~yaml_cpp_class	94
4.58.2	Member Function Documentation	94
4.58.2.1	cleanup	94
4.58.2.2	DisplayContents	94
4.58.2.3	executeYamlRead	94
4.58.2.4	getYamlWrapper	94
4.58.2.5	readInputFile	94
4.58.2.6	setInputFile	94
4.58.3	Member Data Documentation	94
4.58.3.1	current_token	94
4.58.3.2	file_name	94
4.58.3.3	input_file	94
4.58.3.4	previous_token	94
4.58.3.5	token_parser	94
4.58.3.6	yaml_wrapper	94
4.59	YamlWrapper Class Reference	94
4.59.1	Constructor & Destructor Documentation	95
4.59.1.1	YamlWrapper	95
4.59.1.2	~YamlWrapper	95
4.59.1.3	YamlWrapper	95
4.59.1.4	YamlWrapper	95
4.59.2	Member Function Documentation	95
4.59.2.1	addDockKey	95
4.59.2.2	begin	95

4.59.2.3	begin	95
4.59.2.4	changeKey	95
4.59.2.5	clear	96
4.59.2.6	copyAnchor2Alias	96
4.59.2.7	DisplayContents	96
4.59.2.8	end	96
4.59.2.9	end	96
4.59.2.10	getAnchoredDoc	96
4.59.2.11	getDocFromHeadAlias	96
4.59.2.12	getDocFromSubAlias	96
4.59.2.13	getDocMap	96
4.59.2.14	getDocument	96
4.59.2.15	operator()	96
4.59.2.16	operator()	96
4.59.2.17	operator=	96
4.59.2.18	resetKeys	96
4.59.2.19	revalidateAllKeys	96
4.59.2.20	size	96
4.59.3	Member Data Documentation	96
4.59.3.1	Doc_Map	96
5	File Documentation	97
5.1	dogfish.cpp File Reference	97
5.1.1	Function Documentation	97
5.1.1.1	default_FilmMTCoeff	97
5.1.1.2	default_IntraDiffusion	97
5.1.1.3	default_Retardation	97
5.1.1.4	default_SurfaceConcentration	97
5.1.1.5	DOGFISH	98
5.1.1.6	DOGFISH_Executioner	98
5.1.1.7	DOGFISH_postprocesses	98
5.1.1.8	DOGFISH_preprocesses	98
5.1.1.9	DOGFISH_reset	98
5.1.1.10	DOGFISH_TESTS	98
5.1.1.11	print2file_DOGFISH_header	98
5.1.1.12	print2file_DOGFISH_result_new	98
5.1.1.13	print2file_DOGFISH_result_old	98
5.1.1.14	print2file_species_header	98
5.1.1.15	set_DOGFISH_ICs	98
5.1.1.16	set_DOGFISH_params	98

5.1.1.17	set_DOGFISH_timestep	98
5.1.1.18	setup_DOGFISH_DATA	98
5.2	dogfish.h File Reference	98
5.2.1	Function Documentation	99
5.2.1.1	default_FilmMTCoeff	99
5.2.1.2	default_IntraDiffusion	99
5.2.1.3	default_Retardation	99
5.2.1.4	default_SurfaceConcentration	99
5.2.1.5	DOGFISH	99
5.2.1.6	DOGFISH_Executioner	99
5.2.1.7	DOGFISH_postprocesses	99
5.2.1.8	DOGFISH_preprocesses	99
5.2.1.9	DOGFISH_reset	99
5.2.1.10	DOGFISH_TESTS	99
5.2.1.11	print2file_DOGFISH_header	99
5.2.1.12	print2file_DOGFISH_result_new	99
5.2.1.13	print2file_DOGFISH_result_old	99
5.2.1.14	print2file_species_header	99
5.2.1.15	set_DOGFISH_ICs	99
5.2.1.16	set_DOGFISH_params	99
5.2.1.17	set_DOGFISH_timestep	99
5.2.1.18	setup_DOGFISH_DATA	99
5.3	eel.cpp File Reference	99
5.3.1	Function Documentation	100
5.3.1.1	EEL_TESTS	100
5.4	eel.h File Reference	100
5.4.1	Function Documentation	100
5.4.1.1	EEL_TESTS	100
5.5	egret.cpp File Reference	100
5.5.1	Function Documentation	100
5.5.1.1	calculate_properties	100
5.5.1.2	EGRET_TESTS	101
5.5.1.3	initialize_data	101
5.5.1.4	set_variables	101
5.6	egret.h File Reference	101
5.6.1	Macro Definition Documentation	101
5.6.1.1	CE3	101
5.6.1.2	Cstd	101
5.6.1.3	D_ii	101
5.6.1.4	D_ij	102

5.6.1.5	Dp_ij	102
5.6.1.6	FilmMTCoeff	102
5.6.1.7	Mu	102
5.6.1.8	Nu	102
5.6.1.9	PE3	102
5.6.1.10	Po	102
5.6.1.11	PSI	102
5.6.1.12	Pstd	102
5.6.1.13	RE3	102
5.6.1.14	ReNum	102
5.6.1.15	Rstd	102
5.6.1.16	ScNum	102
5.6.2	Function Documentation	102
5.6.2.1	calculate_properties	102
5.6.2.2	EGRET_TESTS	102
5.6.2.3	initialize_data	102
5.6.2.4	set_variables	102
5.7	error.cpp File Reference	102
5.7.1	Function Documentation	102
5.7.1.1	error	102
5.8	error.h File Reference	102
5.8.1	Macro Definition Documentation	103
5.8.1.1	mError	103
5.8.2	Enumeration Type Documentation	103
5.8.2.1	error_type	103
5.8.3	Function Documentation	104
5.8.3.1	error	104
5.9	finch.cpp File Reference	104
5.9.1	Function Documentation	105
5.9.1.1	buckley_leverett_ic	105
5.9.1.2	buckley_leverett_params	105
5.9.1.3	burgers_bcs	105
5.9.1.4	burgers_ic	105
5.9.1.5	burgers_params	106
5.9.1.6	check_Mass	106
5.9.1.7	default_bcs	106
5.9.1.8	default_execution	106
5.9.1.9	default_ic	106
5.9.1.10	default_params	106
5.9.1.11	default_postprocess	106

5.9.1.12	default_precon	106
5.9.1.13	default_preprocess	106
5.9.1.14	default_res	106
5.9.1.15	default_reset	106
5.9.1.16	default_solve	106
5.9.1.17	default_timestep	106
5.9.1.18	FINCH_TESTS	106
5.9.1.19	l_direct	106
5.9.1.20	lark_picard_step	106
5.9.1.21	max	106
5.9.1.22	min	106
5.9.1.23	minmod	106
5.9.1.24	minmod_discretization	106
5.9.1.25	nl_picard	106
5.9.1.26	ospre_discretization	106
5.9.1.27	print2file_dim_header	106
5.9.1.28	print2file_newline	106
5.9.1.29	print2file_result_new	106
5.9.1.30	print2file_result_old	106
5.9.1.31	print2file_tab	106
5.9.1.32	print2file_time_header	106
5.9.1.33	setup_FINCH_DATA	107
5.9.1.34	uAverage	107
5.9.1.35	uTotal	107
5.9.1.36	vanAlbada_discretization	107
5.10	finch.h File Reference	107
5.10.1	Macro Definition Documentation	108
5.10.1.1	Cartesian	108
5.10.1.2	Cylindrical	108
5.10.1.3	FINCH_Picard	108
5.10.1.4	LARK_Picard	108
5.10.1.5	LARK_PJFNK	108
5.10.1.6	Spherical	108
5.10.2	Function Documentation	108
5.10.2.1	buckley_leverett_ic	108
5.10.2.2	buckley_leverett_params	108
5.10.2.3	burgers_bcs	108
5.10.2.4	burgers_ic	108
5.10.2.5	burgers_params	108
5.10.2.6	check_Mass	108

5.10.2.7	default_bcs	108
5.10.2.8	default_execution	109
5.10.2.9	default_ic	109
5.10.2.10	default_params	109
5.10.2.11	default_postprocess	109
5.10.2.12	default_precon	109
5.10.2.13	default_preprocess	109
5.10.2.14	default_res	109
5.10.2.15	default_reset	109
5.10.2.16	default_solve	109
5.10.2.17	default_timestep	109
5.10.2.18	FINCH_TESTS	109
5.10.2.19	l_direct	109
5.10.2.20	lark_picard_step	109
5.10.2.21	max	109
5.10.2.22	min	109
5.10.2.23	minmod	109
5.10.2.24	minmod_discretization	109
5.10.2.25	nl_picard	109
5.10.2.26	ospre_discretization	109
5.10.2.27	print2file_dim_header	109
5.10.2.28	print2file_newline	109
5.10.2.29	print2file_result_new	109
5.10.2.30	print2file_result_old	109
5.10.2.31	print2file_tab	109
5.10.2.32	print2file_time_header	109
5.10.2.33	setup_FINCH_DATA	110
5.10.2.34	uAverage	110
5.10.2.35	uTotal	110
5.10.2.36	vanAlbada_discretization	110
5.11	flock.h File Reference	110
5.12	gsta_opt.cpp File Reference	110
5.12.1	Function Documentation	111
5.12.1.1	avgPar	111
5.12.1.2	avgValue	111
5.12.1.3	eduGuess	111
5.12.1.4	eval_GSTA	111
5.12.1.5	gsta_optimize	111
5.12.1.6	gstaFunc	111
5.12.1.7	gstaObjFunc	111

5.12.1.8	isSmooth	111
5.12.1.9	minIndex	111
5.12.1.10	minValue	111
5.12.1.11	orderMag	111
5.12.1.12	orthoLinReg	111
5.12.1.13	roundIt	111
5.12.1.14	rSq	111
5.12.1.15	twoFifths	111
5.12.1.16	weightedAvg	111
5.13	gsta_opt.h File Reference	111
5.13.1	Macro Definition Documentation	112
5.13.1.1	Na	112
5.13.1.2	Po	112
5.13.1.3	R	112
5.13.2	Function Documentation	112
5.13.2.1	avgPar	112
5.13.2.2	avgValue	112
5.13.2.3	eduGuess	112
5.13.2.4	error	112
5.13.2.5	eval_GSTA	112
5.13.2.6	gsta_optimize	112
5.13.2.7	gstaFunc	112
5.13.2.8	gstaObjFunc	112
5.13.2.9	isSmooth	112
5.13.2.10	minIndex	113
5.13.2.11	minValue	113
5.13.2.12	orderMag	113
5.13.2.13	orthoLinReg	113
5.13.2.14	roundIt	113
5.13.2.15	rSq	113
5.13.2.16	twoFifths	113
5.13.2.17	weightedAvg	113
5.14	lark.cpp File Reference	113
5.14.1	Function Documentation	114
5.14.1.1	arnoldi	114
5.14.1.2	backtrackLineSearch	114
5.14.1.3	bicgstab	114
5.14.1.4	cgs	114
5.14.1.5	evalx_ex09	114
5.14.1.6	fom	114

5.14.1.7	funeval_ex09	114
5.14.1.8	funeval_ex10	114
5.14.1.9	gcr	115
5.14.1.10	gmresLeftPreconditioned	115
5.14.1.11	gmresPreconditioner	115
5.14.1.12	gmresr	115
5.14.1.13	gmresRightPreconditioned	115
5.14.1.14	jacvec	115
5.14.1.15	LARK_TESTS	115
5.14.1.16	matvec_ex01	115
5.14.1.17	matvec_ex02	115
5.14.1.18	matvec_ex04	115
5.14.1.19	matvec_ex15	115
5.14.1.20	NumericalJacobian	115
5.14.1.21	operatorTranspose	115
5.14.1.22	pcg	115
5.14.1.23	picard	115
5.14.1.24	pjfnk	115
5.14.1.25	precon_ex01	115
5.14.1.26	precon_ex04	115
5.14.1.27	precon_ex10	115
5.14.1.28	precon_ex15	115
5.14.1.29	update_arnoldi_solution	116
5.15	lark.h File Reference	116
5.15.1	Enumeration Type Documentation	117
5.15.1.1	krylov_method	117
5.15.2	Function Documentation	118
5.15.2.1	arnoldi	118
5.15.2.2	backtrackLineSearch	118
5.15.2.3	bicgstab	118
5.15.2.4	cgs	118
5.15.2.5	evalx_ex09	118
5.15.2.6	fom	118
5.15.2.7	funeval_ex09	118
5.15.2.8	funeval_ex10	118
5.15.2.9	gcr	118
5.15.2.10	gmresLeftPreconditioned	118
5.15.2.11	gmresPreconditioner	118
5.15.2.12	gmresr	118
5.15.2.13	gmresRightPreconditioned	118

5.15.2.14 jacvec	118
5.15.2.15 LARK_TESTS	118
5.15.2.16 matvec_ex01	118
5.15.2.17 matvec_ex02	118
5.15.2.18 matvec_ex04	118
5.15.2.19 matvec_ex15	119
5.15.2.20 NumericalJacobian	119
5.15.2.21 operatorTranspose	119
5.15.2.22 pcg	119
5.15.2.23 picard	119
5.15.2.24 pjfnk	119
5.15.2.25 precon_ex01	119
5.15.2.26 precon_ex04	119
5.15.2.27 precon_ex10	119
5.15.2.28 precon_ex15	119
5.15.2.29 update_arnoldi_solution	119
5.16 macaw.cpp File Reference	119
5.16.1 Function Documentation	119
5.16.1.1 MACAW_TESTS	119
5.17 macaw.h File Reference	119
5.17.1 Macro Definition Documentation	120
5.17.1.1 M_PI	120
5.17.2 Function Documentation	120
5.17.2.1 MACAW_TESTS	120
5.18 magpie.cpp File Reference	120
5.18.1 Function Documentation	121
5.18.1.1 dq_dp	121
5.18.1.2 eMax	121
5.18.1.3 eval_eta	121
5.18.1.4 eval_GPAST	121
5.18.1.5 eval_po	121
5.18.1.6 eval_po_PI	121
5.18.1.7 eval_po_qo	121
5.18.1.8 grad_mSPD	121
5.18.1.9 initialGuess_mSPD	121
5.18.1.10 Inact_mSPD	121
5.18.1.11 MAGPIE	121
5.18.1.12 MAGPIE_SCENARIOS	121
5.18.1.13 PI	121
5.18.1.14 q_p	121

5.18.1.15 qo	121
5.18.1.16 Qst	121
5.18.1.17 qT	121
5.19 magpie.h File Reference	121
5.19.1 Macro Definition Documentation	122
5.19.1.1 A	122
5.19.1.2 DBL_EPSILON	122
5.19.1.3 He	122
5.19.1.4 kB	122
5.19.1.5 lnKo	122
5.19.1.6 Na	123
5.19.1.7 Po	123
5.19.1.8 R	123
5.19.1.9 shapeFactor	123
5.19.1.10 V	123
5.19.1.11 Z	123
5.19.2 Function Documentation	123
5.19.2.1 dq_dp	123
5.19.2.2 eMax	123
5.19.2.3 eval_eta	123
5.19.2.4 eval_GPAST	123
5.19.2.5 eval_po	123
5.19.2.6 eval_po_PI	123
5.19.2.7 eval_po_qo	123
5.19.2.8 grad_mSPD	123
5.19.2.9 initialGuess_mSPD	123
5.19.2.10 lnact_mSPD	123
5.19.2.11 MAGPIE	123
5.19.2.12 MAGPIE_SCENARIOS	123
5.19.2.13 PI	123
5.19.2.14 q_p	123
5.19.2.15 qo	123
5.19.2.16 Qst	123
5.19.2.17 qT	123
5.20 main.cpp File Reference	123
5.20.1 Function Documentation	124
5.20.1.1 main	124
5.21 mola.cpp File Reference	124
5.21.1 Function Documentation	124
5.21.1.1 MOLA_TESTS	124

5.22 mola.h File Reference	124
5.22.1 Function Documentation	124
5.22.1.1 MOLA_TESTS	124
5.23 monkfish.cpp File Reference	124
5.23.1 Function Documentation	125
5.23.1.1 default_density	125
5.23.1.2 default_exterior_concentration	125
5.23.1.3 default_film_transfer	125
5.23.1.4 default_interparticle_diffusion	125
5.23.1.5 default_monk_adsorption	125
5.23.1.6 default_monk_equilibrium	125
5.23.1.7 default_monkfish_retardation	125
5.23.1.8 default_porosity	125
5.23.1.9 MONKFISH_TESTS	125
5.24 monkfish.h File Reference	125
5.24.1 Function Documentation	126
5.24.1.1 default_density	126
5.24.1.2 default_exterior_concentration	126
5.24.1.3 default_film_transfer	126
5.24.1.4 default_interparticle_diffusion	126
5.24.1.5 default_monk_adsorption	126
5.24.1.6 default_monk_equilibrium	126
5.24.1.7 default_monkfish_retardation	126
5.24.1.8 default_porosity	126
5.24.1.9 MONKFISH_TESTS	126
5.24.1.10 setup_MONKFISH_DATA	126
5.25 sandbox.cpp File Reference	126
5.25.1 Function Documentation	126
5.25.1.1 RUN_SANDBOX	126
5.25.1.2 Speciation_Test01_Function	126
5.25.1.3 Speciation_Test01_Guess	126
5.25.1.4 Speciation_Test01_Jacobian	126
5.25.1.5 Speciation_Test01_MatVec	126
5.26 sandbox.h File Reference	127
5.26.1 Function Documentation	127
5.26.1.1 RUN_SANDBOX	127
5.26.1.2 Speciation_Test01_Function	127
5.26.1.3 Speciation_Test01_Guess	127
5.26.1.4 Speciation_Test01_Jacobian	127
5.26.1.5 Speciation_Test01_MatVec	127

5.27	school.h File Reference	127
5.28	scopsowl.cpp File Reference	127
5.28.1	Function Documentation	128
5.28.1.1	const_filmMassTransfer	128
5.28.1.2	const_pore_diffusion	128
5.28.1.3	CURVE_TEST03	128
5.28.1.4	CURVE_TEST04	128
5.28.1.5	CURVE_TEST05	128
5.28.1.6	default_adsorption	128
5.28.1.7	default_effective_diffusion	128
5.28.1.8	default_filmMassTransfer	128
5.28.1.9	default_pore_diffusion	128
5.28.1.10	default_retardation	128
5.28.1.11	default_surf_diffusion	128
5.28.1.12	LARGE_CYCLE_TEST01	129
5.28.1.13	print2file_SCOPSOWL_header	129
5.28.1.14	print2file_SCOPSOWL_result_new	129
5.28.1.15	print2file_SCOPSOWL_result_old	129
5.28.1.16	print2file_SCOPSOWL_time_header	129
5.28.1.17	print2file_species_header	129
5.28.1.18	SCOPSOWL	129
5.28.1.19	SCOPSOWL_Executioner	129
5.28.1.20	SCOPSOWL_postprocesses	129
5.28.1.21	SCOPSOWL_preprocesses	129
5.28.1.22	SCOPSOWL_reset	129
5.28.1.23	SCOPSOWL_SCENARIOS	129
5.28.1.24	SCOPSOWL_TESTS	129
5.28.1.25	set_SCOPSOWL_ICs	129
5.28.1.26	set_SCOPSOWL_params	129
5.28.1.27	set_SCOPSOWL_timestep	129
5.28.1.28	setup_SCOPSOWL_DATA	129
5.28.1.29	SMALL_CYCLE_TEST02	129
5.29	scopsowl.h File Reference	129
5.29.1	Macro Definition Documentation	130
5.29.1.1	avgDp	130
5.29.1.2	Dk	130
5.29.1.3	Dp	130
5.29.1.4	SCOPSOWL_HPP_	130
5.29.2	Function Documentation	130
5.29.2.1	const_filmMassTransfer	130

5.29.2.2	const_pore_diffusion	131
5.29.2.3	CURVE_TEST03	131
5.29.2.4	CURVE_TEST04	131
5.29.2.5	CURVE_TEST05	131
5.29.2.6	default_adsorption	131
5.29.2.7	default_effective_diffusion	131
5.29.2.8	default_filmMassTransfer	131
5.29.2.9	default_pore_diffusion	131
5.29.2.10	default_retardation	131
5.29.2.11	default_surf_diffusion	131
5.29.2.12	LARGE_CYCLE_TEST01	131
5.29.2.13	print2file_SCOPSOWL_header	131
5.29.2.14	print2file_SCOPSOWL_result_new	131
5.29.2.15	print2file_SCOPSOWL_result_old	131
5.29.2.16	print2file_SCOPSOWL_time_header	131
5.29.2.17	print2file_species_header	131
5.29.2.18	SCOPSOWL	131
5.29.2.19	SCOPSOWL_Executioner	131
5.29.2.20	SCOPSOWL_postprocesses	131
5.29.2.21	SCOPSOWL_preprocesses	131
5.29.2.22	SCOPSOWL_reset	131
5.29.2.23	SCOPSOWL_SCENARIOS	131
5.29.2.24	SCOPSOWL_TESTS	131
5.29.2.25	set_SCOPSOWL_ICs	131
5.29.2.26	set_SCOPSOWL_params	131
5.29.2.27	set_SCOPSOWL_timestep	131
5.29.2.28	setup_SCOPSOWL_DATA	131
5.29.2.29	SMALL_CYCLE_TEST02	132
5.30	scopsowl_opt.cpp File Reference	132
5.30.1	Function Documentation	132
5.30.1.1	eval_SCOPSOWL_Uptake	132
5.30.1.2	initial_guess_SCOPSOWL	132
5.30.1.3	SCOPSOWL_OPT_set_y	132
5.30.1.4	SCOPSOWL_OPTIMIZE	132
5.31	scopsowl_opt.h File Reference	132
5.31.1	Function Documentation	132
5.31.1.1	eval_SCOPSOWL_Uptake	132
5.31.1.2	initial_guess_SCOPSOWL	132
5.31.1.3	SCOPSOWL_OPT_set_y	132
5.31.1.4	SCOPSOWL_OPTIMIZE	133

5.32 shark.cpp File Reference	133
5.32.1 Function Documentation	134
5.32.1.1 act_choice	134
5.32.1.2 Convert2Concentration	134
5.32.1.3 Convert2LogConcentration	134
5.32.1.4 Davies_equation	134
5.32.1.5 DaviesLadshaw_equation	134
5.32.1.6 DebyeHuckel_equation	134
5.32.1.7 ideal_solution	134
5.32.1.8 linearsolve_choice	134
5.32.1.9 linesearch_choice	134
5.32.1.10 print2file_shark_header	134
5.32.1.11 print2file_shark_info	134
5.32.1.12 print2file_shark_results_new	134
5.32.1.13 print2file_shark_results_old	134
5.32.1.14 read_equilrxn	134
5.32.1.15 read_massbalance	134
5.32.1.16 read_options	134
5.32.1.17 read_scenario	134
5.32.1.18 read_species	134
5.32.1.19 read_unsteadyrxn	134
5.32.1.20 setup_SHARK_DATA	134
5.32.1.21 SHARK	134
5.32.1.22 shark_add_customResidual	134
5.32.1.23 shark_energy_calculations	134
5.32.1.24 shark_executioner	134
5.32.1.25 shark_guess	134
5.32.1.26 shark_initial_conditions	135
5.32.1.27 shark_parameter_check	135
5.32.1.28 shark_pH_finder	135
5.32.1.29 shark_postprocesses	135
5.32.1.30 shark_preprocesses	135
5.32.1.31 shark_reset	135
5.32.1.32 shark_residual	135
5.32.1.33 SHARK_SCENARIO	135
5.32.1.34 shark_solver	135
5.32.1.35 shark_temperature_calculations	135
5.32.1.36 SHARK_TESTS	135
5.32.1.37 shark_timestep_adapt	135
5.32.1.38 shark_timestep_const	135

5.33 shark.h File Reference	135
5.33.1 Macro Definition Documentation	137
5.33.1.1 Rstd	137
5.33.2 Typedef Documentation	137
5.33.2.1 SHARK_DATA	137
5.33.3 Enumeration Type Documentation	137
5.33.3.1 valid_act	137
5.33.4 Function Documentation	137
5.33.4.1 act_choice	137
5.33.4.2 Convert2Concentration	137
5.33.4.3 Convert2LogConcentration	137
5.33.4.4 Davies_equation	137
5.33.4.5 DaviesLadshaw_equation	137
5.33.4.6 DebyeHuckel_equation	137
5.33.4.7 ideal_solution	137
5.33.4.8 linearsolve_choice	137
5.33.4.9 linesearch_choice	137
5.33.4.10 print2file_shark_header	137
5.33.4.11 print2file_shark_info	137
5.33.4.12 print2file_shark_results_new	137
5.33.4.13 print2file_shark_results_old	137
5.33.4.14 read_equilrxn	137
5.33.4.15 read_massbalance	137
5.33.4.16 read_options	137
5.33.4.17 read_scenario	138
5.33.4.18 read_species	138
5.33.4.19 read_unsteadyrxn	138
5.33.4.20 setup_SHARK_DATA	138
5.33.4.21 SHARK	138
5.33.4.22 shark_add_customResidual	138
5.33.4.23 shark_energy_calculations	138
5.33.4.24 shark_executioner	138
5.33.4.25 shark_guess	138
5.33.4.26 shark_initial_conditions	138
5.33.4.27 shark_parameter_check	138
5.33.4.28 shark_pH_finder	138
5.33.4.29 shark_postprocesses	138
5.33.4.30 shark_preprocesses	138
5.33.4.31 shark_reset	138
5.33.4.32 shark_residual	138

5.33.4.33 SHARK_SCENARIO	138
5.33.4.34 shark_solver	138
5.33.4.35 shark_temperature_calculations	138
5.33.4.36 SHARK_TESTS	138
5.33.4.37 shark_timestep_adapt	138
5.33.4.38 shark_timestep_const	138
5.34 skua.cpp File Reference	138
5.34.1 Function Documentation	139
5.34.1.1 const_Dc	139
5.34.1.2 const_kf	139
5.34.1.3 default_Dc	139
5.34.1.4 default_kf	139
5.34.1.5 empirical_kf	139
5.34.1.6 molefractionCheck	139
5.34.1.7 print2file_SKUA_header	139
5.34.1.8 print2file_SKUA_results_new	139
5.34.1.9 print2file_SKUA_results_old	139
5.34.1.10 print2file_SKUA_time_header	139
5.34.1.11 print2file_species_header	140
5.34.1.12 set_SKUA_ICs	140
5.34.1.13 set_SKUA_params	140
5.34.1.14 set_SKUA_timestep	140
5.34.1.15 setup_SKUA_DATA	140
5.34.1.16 simple_darken_Dc	140
5.34.1.17 SKUA	140
5.34.1.18 SKUA_CYCLE_TEST01	140
5.34.1.19 SKUA_CYCLE_TEST02	140
5.34.1.20 SKUA_Executioner	140
5.34.1.21 SKUA_LOW_TEST03	140
5.34.1.22 SKUA_MID_TEST04	140
5.34.1.23 SKUA_postprocesses	140
5.34.1.24 SKUA_preprocesses	140
5.34.1.25 SKUA_reset	140
5.34.1.26 SKUA_SCENARIOS	140
5.34.1.27 SKUA_TESTS	140
5.34.1.28 theoretical_darken_Dc	140
5.35 skua.h File Reference	140
5.35.1 Macro Definition Documentation	141
5.35.1.1 D_c	141
5.35.1.2 D_inf	141

5.35.1.3	D_o	141
5.35.1.4	SKUA_HPP_	141
5.35.2	Function Documentation	141
5.35.2.1	const_Dc	141
5.35.2.2	const_kf	141
5.35.2.3	default_Dc	141
5.35.2.4	default_kf	141
5.35.2.5	empirical_kf	141
5.35.2.6	molefractionCheck	142
5.35.2.7	print2file_SKUA_header	142
5.35.2.8	print2file_SKUA_results_new	142
5.35.2.9	print2file_SKUA_results_old	142
5.35.2.10	print2file_SKUA_time_header	142
5.35.2.11	print2file_species_header	142
5.35.2.12	set_SKUA_ICs	142
5.35.2.13	set_SKUA_params	142
5.35.2.14	set_SKUA_timestep	142
5.35.2.15	setup_SKUA_DATA	142
5.35.2.16	simple_darken_Dc	142
5.35.2.17	SKUA	142
5.35.2.18	SKUA_CYCLE_TEST01	142
5.35.2.19	SKUA_CYCLE_TEST02	142
5.35.2.20	SKUA_Executioner	142
5.35.2.21	SKUA_LOW_TEST03	142
5.35.2.22	SKUA_MID_TEST04	142
5.35.2.23	SKUA_postprocesses	142
5.35.2.24	SKUA_preprocesses	142
5.35.2.25	SKUA_reset	142
5.35.2.26	SKUA_SCENARIOS	142
5.35.2.27	SKUA_TESTS	142
5.35.2.28	theoretical_darken_Dc	142
5.36	skua_opt.cpp File Reference	142
5.36.1	Function Documentation	143
5.36.1.1	eval_SKUA_Uptake	143
5.36.1.2	initial_guess_SKUA	143
5.36.1.3	SKUA_OPT_set_y	143
5.36.1.4	SKUA_OPTIMIZE	143
5.37	skua_opt.h File Reference	143
5.37.1	Function Documentation	143
5.37.1.1	eval_SKUA_Uptake	143

5.37.1.2	initial_guess_SKUA	143
5.37.1.3	SKUA_OPT_set_y	143
5.37.1.4	SKUA_OPTIMIZE	143
5.38	Trajectory.cpp File Reference	143
5.38.1	Function Documentation	144
5.38.1.1	Brown_RAD	144
5.38.1.2	Brown_THETA	144
5.38.1.3	CARTESIAN	144
5.38.1.4	DISPLACEMENT	144
5.38.1.5	Grav_R	144
5.38.1.6	Grav_T	144
5.38.1.7	LOCATION	144
5.38.1.8	Magnetic_R	144
5.38.1.9	Magnetic_T	144
5.38.1.10	Number_Generator	145
5.38.1.11	POLAR	145
5.38.1.12	RADIAL_FORCE	145
5.38.1.13	Removal_Efficiency	145
5.38.1.14	Run_Trajectory	145
5.38.1.15	TANGENTIAL_FORCE	145
5.38.1.16	Trajectory_SetupConstants	145
5.38.1.17	V_RAD	145
5.38.1.18	V_THETA	145
5.38.1.19	Van_R	145
5.39	Trajectory.h File Reference	145
5.39.1	Function Documentation	146
5.39.1.1	Brown_RAD	146
5.39.1.2	Brown_THETA	146
5.39.1.3	CARTESIAN	146
5.39.1.4	DISPLACEMENT	146
5.39.1.5	Grav_R	146
5.39.1.6	Grav_T	146
5.39.1.7	LOCATION	146
5.39.1.8	Magnetic_R	146
5.39.1.9	Magnetic_T	146
5.39.1.10	Number_Generator	146
5.39.1.11	POLAR	146
5.39.1.12	RADIAL_FORCE	146
5.39.1.13	Removal_Efficiency	146
5.39.1.14	Run_Trajectory	146

5.39.1.15 TANGENTIAL_FORCE	146
5.39.1.16 Trajectory_SetupConstants	146
5.39.1.17 V_RAD	147
5.39.1.18 V_THETA	147
5.39.1.19 Van_R	147
5.40 ui.cpp File Reference	147
5.40.1 Function Documentation	147
5.40.1.1 allLower	147
5.40.1.2 aui_help	147
5.40.1.3 bui_help	147
5.40.1.4 display_help	147
5.40.1.5 display_version	147
5.40.1.6 exec	148
5.40.1.7 exec_loop	148
5.40.1.8 exit	148
5.40.1.9 help	148
5.40.1.10 input	148
5.40.1.11 invalid_input	148
5.40.1.12 number_files	148
5.40.1.13 path	148
5.40.1.14 run_exec	148
5.40.1.15 run_executable	148
5.40.1.16 run_test	148
5.40.1.17 test	148
5.40.1.18 test_loop	148
5.40.1.19 valid_addon_options	148
5.40.1.20 valid_exec_string	148
5.40.1.21 valid_input_execute	148
5.40.1.22 valid_input_main	148
5.40.1.23 valid_input_tests	148
5.40.1.24 valid_test_string	148
5.40.1.25 version	148
5.41 ui.h File Reference	148
5.41.1 Macro Definition Documentation	150
5.41.1.1 ECO_EXECUTABLE	150
5.41.1.2 ECO_VERSION	150
5.41.1.3 UI_HPP_	150
5.41.2 Enumeration Type Documentation	150
5.41.2.1 valid_options	150
5.41.3 Function Documentation	150

5.41.3.1	allLower	150
5.41.3.2	aui_help	150
5.41.3.3	bui_help	150
5.41.3.4	display_help	150
5.41.3.5	display_version	150
5.41.3.6	exec	150
5.41.3.7	exec_loop	151
5.41.3.8	exec_option	151
5.41.3.9	exit	151
5.41.3.10	help	151
5.41.3.11	input	151
5.41.3.12	invalid_input	151
5.41.3.13	number_files	151
5.41.3.14	path	151
5.41.3.15	run_exec	151
5.41.3.16	run_executable	151
5.41.3.17	run_test	151
5.41.3.18	test	151
5.41.3.19	test_loop	151
5.41.3.20	valid_addon_options	151
5.41.3.21	valid_exec_string	151
5.41.3.22	valid_input_execute	151
5.41.3.23	valid_input_main	151
5.41.3.24	valid_input_tests	151
5.41.3.25	valid_test_string	151
5.41.3.26	version	151
5.42	yaml_wrapper.cpp File Reference	151
5.42.1	Function Documentation	151
5.42.1.1	YAML_CPP_TEST	151
5.42.1.2	YAML_WRAPPER_TESTS	152
5.43	yaml_wrapper.h File Reference	152
5.43.1	Typedef Documentation	152
5.43.1.1	data_type	152
5.43.1.2	header_state	152
5.43.2	Enumeration Type Documentation	152
5.43.2.1	data_type	152
5.43.2.2	header_state	153
5.43.3	Function Documentation	153
5.43.3.1	YAML_CPP_TEST	153
5.43.3.2	YAML_WRAPPER_TESTS	153

Index**153**

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

ARNOLDI_DATA	7
Atom	8
Molecule	50
MasterSpeciesList	42
PeriodicTable	58
BACKTRACK_DATA	10
BiCGSTAB_DATA	11
CGS_DATA	13
DOGFISH_DATA	17
DOGFISH_PARAM	19
EX01_DATA	19
EX02_DATA	20
EX04_DATA	20
EX09_DATA	21
EX15_DATA	21
FINCH_DATA	22
GCR_DATA	27
GMRESLP_DATA	29
GMRESR_DATA	30
GMRESRP_DATA	31
GPAST_DATA	33
GSTA_DATA	34
GSTA_OPT_DATA	34
KeyValueMap	38
MAGPIE_DATA	41
MassBalance	41
Matrix< T >	44
Matrix< double >	44
Matrix< int >	44
Mechanism	48
MIXED_GAS	48
MONKFISH_DATA	53
MONKFISH_PARAM	55
mSPD_DATA	56
NUM_JAC_DATA	56
OPTRANS_DATA	57
PCG_DATA	57

PICARD_DATA	60
PJFNK_DATA	60
PURE_GAS	63
Reaction	64
Precipitation	63
UnsteadyPrecipitation	87
UnsteadyReaction	88
SCOPSOWL_DATA	66
SCOPSOWL_OPT_DATA	68
SCOPSOWL_PARAM_DATA	70
SHARK_DATA	72
SKUA_DATA	75
SKUA_OPT_DATA	77
SKUA_PARAM	78
Speciation_Test01_Data	79
SubHeader	80
Document	14
Header	35
SYSTEM_DATA	82
TRAJECTORY_DATA	84
UI_DATA	86
ValueTypePair	92
yaml_cpp_class	93
YamlWrapper	94

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

ARNOLDI_DATA	7
Atom	8
BACKTRACK_DATA	10
BiCGSTAB_DATA	11
CGS_DATA	13
Document	14
DOGFISH_DATA	17
DOGFISH_PARAM	19
EX01_DATA	19
EX02_DATA	20
EX04_DATA	20
EX09_DATA	21
EX15_DATA	21
FINCH_DATA	22
GCR_DATA	27
GMRESLP_DATA	29
GMRESR_DATA	30
GMRESRP_DATA	31
GPAST_DATA	33
GSTA_DATA	34
GSTA_OPT_DATA	34
Header	35
KeyValueMap	38
MAGPIE_DATA	41
MassBalance	41
MasterSpeciesList	42
Matrix< T >	44
Mechanism	48
MIXED_GAS	48
Molecule	50
MONKFISH_DATA	53
MONKFISH_PARAM	55
mSPD_DATA	56
NUM_JAC_DATA	56
OPTRANS_DATA	57
PCG_DATA	57
PeriodicTable	58
PICARD_DATA	60

PJFNK_DATA	60
Precipitation	63
PURE_GAS	63
Reaction	64
SCOPSOWL_DATA	66
SCOPSOWL_OPT_DATA	68
SCOPSOWL_PARAM_DATA	70
SHARK_DATA	72
SKUA_DATA	75
SKUA_OPT_DATA	77
SKUA_PARAM	78
Speciation_Test01_Data	79
SubHeader	80
SYSTEM_DATA	82
TRAJECTORY_DATA	84
UI_DATA	86
UnsteadyPrecipitation	87
UnsteadyReaction	88
ValueTypePair	92
yaml_cpp_class	93
YamlWrapper	94

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

dogfish.cpp	97
dogfish.h	98
eel.cpp	99
eel.h	100
egret.cpp	100
egret.h	101
error.cpp	102
error.h	102
finch.cpp	104
finch.h	107
flock.h	110
gsta_opt.cpp	110
gsta_opt.h	111
lark.cpp	113
lark.h	116
macaw.cpp	119
macaw.h	119
magpie.cpp	120
magpie.h	121
main.cpp	123
mola.cpp	124
mola.h	124
monkfish.cpp	124
monkfish.h	125
sandbox.cpp	126
sandbox.h	127
school.h	127
scopsowl.cpp	127
scopsowl.h	129
scopsowl_opt.cpp	132
scopsowl_opt.h	132
shark.cpp	133
shark.h	135
skua.cpp	138
skua.h	140
skua_opt.cpp	142
skua_opt.h	143
Trajectory.cpp	143

Trajectory.h	145
ui.cpp	147
ui.h	148
yaml_wrapper.cpp	151
yaml_wrapper.h	152

Chapter 4

Class Documentation

4.1 ARNOLDI_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- int [k](#)
- int [iter](#)
- double [beta](#)
- double [hp1](#)
- bool [Output](#) = true
- std::vector< [Matrix](#)< double > > [Vk](#)
- [Matrix](#)< double > [Hkp1](#)
- [Matrix](#)< double > [yk](#)
- [Matrix](#)< double > [e1](#)
- [Matrix](#)< double > [w](#)
- [Matrix](#)< double > [v](#)
- [Matrix](#)< double > [sum](#)

4.1.1 Member Data Documentation

4.1.1.1 double ARNOLDI_DATA::beta

4.1.1.2 [Matrix](#)<double> ARNOLDI_DATA::e1

4.1.1.3 [Matrix](#)<double> ARNOLDI_DATA::Hkp1

4.1.1.4 double ARNOLDI_DATA::hp1

4.1.1.5 int ARNOLDI_DATA::iter

4.1.1.6 int ARNOLDI_DATA::k

4.1.1.7 bool ARNOLDI_DATA::Output = true

4.1.1.8 [Matrix](#)<double> ARNOLDI_DATA::sum

4.1.1.9 [Matrix](#)<double> ARNOLDI_DATA::v

4.1.1.10 `std::vector< Matrix<double> > ARNOLDI_DATA::Vk`

4.1.1.11 `Matrix<double> ARNOLDI_DATA::w`

4.1.1.12 `Matrix<double> ARNOLDI_DATA::yk`

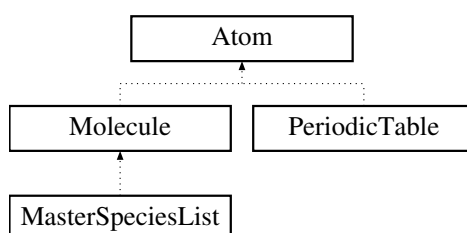
The documentation for this struct was generated from the following file:

- [lark.h](#)

4.2 Atom Class Reference

```
#include <eel.h>
```

Inheritance diagram for Atom:



Public Member Functions

- [Atom](#) ()
- [~Atom](#) ()
- [Atom](#) (std::string [Name](#))
- [Atom](#) (int number)
- void [Register](#) (std::string [Symbol](#))
- void [Register](#) (int number)
- void [editAtomicWeight](#) (double AW)
- void [editOxidationState](#) (int state)
- void [editProtons](#) (int proton)
- void [editNeutrons](#) (int neutron)
- void [editElectrons](#) (int electron)
- void [editValence](#) (int val)
- void [removeProton](#) ()
- void [removeNeutron](#) ()
- void [removeElectron](#) ()
- double [AtomicWeight](#) ()
- int [OxidationState](#) ()
- int [Protons](#) ()
- int [Neutrons](#) ()
- int [Electrons](#) ()
- int [BondingElectrons](#) ()
- std::string [AtomName](#) ()
- std::string [AtomSymbol](#) ()
- std::string [AtomCategory](#) ()
- std::string [AtomState](#) ()
- int [AtomicNumber](#) ()
- void [DisplayInfo](#) ()

Protected Attributes

- double [atomic_weight](#)
- int [oxidation_state](#)
- int [protons](#)
- int [neutrons](#)
- int [electrons](#)
- int [valence_e](#)

Private Attributes

- std::string [Name](#)
- std::string [Symbol](#)
- std::string [Category](#)
- std::string [NaturalState](#)
- int [atomic_number](#)

4.2.1 Constructor & Destructor Documentation

4.2.1.1 `Atom::Atom ()`

4.2.1.2 `Atom::~~Atom ()`

4.2.1.3 `Atom::Atom (std::string Name)`

4.2.1.4 `Atom::Atom (int number)`

4.2.2 Member Function Documentation

4.2.2.1 `std::string Atom::AtomCategory ()`

4.2.2.2 `int Atom::AtomicNumber ()`

4.2.2.3 `double Atom::AtomicWeight ()`

4.2.2.4 `std::string Atom::AtomName ()`

4.2.2.5 `std::string Atom::AtomState ()`

4.2.2.6 `std::string Atom::AtomSymbol ()`

4.2.2.7 `int Atom::BondingElectrons ()`

4.2.2.8 `void Atom::DisplayInfo ()`

4.2.2.9 `void Atom::editAtomicWeight (double AW)`

4.2.2.10 `void Atom::editElectrons (int electron)`

4.2.2.11 `void Atom::editNeutrons (int neutron)`

4.2.2.12 `void Atom::editOxidationState (int state)`

4.2.2.13 `void Atom::editProtons (int proton)`

4.2.2.14 void Atom::editValence (int *val*)

4.2.2.15 int Atom::Electrons ()

4.2.2.16 int Atom::Neutrons ()

4.2.2.17 int Atom::OxidationState ()

4.2.2.18 int Atom::Protons ()

4.2.2.19 void Atom::Register (std::string *Symbol*)

4.2.2.20 void Atom::Register (int *number*)

4.2.2.21 void Atom::removeElectron ()

4.2.2.22 void Atom::removeNeutron ()

4.2.2.23 void Atom::removeProton ()

4.2.3 Member Data Documentation

4.2.3.1 int Atom::atomic_number [private]

4.2.3.2 double Atom::atomic_weight [protected]

4.2.3.3 std::string Atom::Category [private]

4.2.3.4 int Atom::electrons [protected]

4.2.3.5 std::string Atom::Name [private]

4.2.3.6 std::string Atom::NaturalState [private]

4.2.3.7 int Atom::neutrons [protected]

4.2.3.8 int Atom::oxidation_state [protected]

4.2.3.9 int Atom::protons [protected]

4.2.3.10 std::string Atom::Symbol [private]

4.2.3.11 int Atom::valence_e [protected]

The documentation for this class was generated from the following files:

- [eel.h](#)
- [eel.cpp](#)

4.3 BACKTRACK_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- double [alpha](#) = 1e-4
- double [rho](#) = 0.1
- double [lambdaMin](#) = DBL_EPSILON
- double [normFkp1](#)
- bool [constRho](#) = false
- [Matrix](#)< double > [Fk](#)
- [Matrix](#)< double > [xk](#)

4.3.1 Member Data Documentation

4.3.1.1 double BACKTRACK_DATA::alpha = 1e-4

4.3.1.2 bool BACKTRACK_DATA::constRho = false

4.3.1.3 [Matrix](#)<double> BACKTRACK_DATA::Fk

4.3.1.4 double BACKTRACK_DATA::lambdaMin = DBL_EPSILON

4.3.1.5 double BACKTRACK_DATA::normFkp1

4.3.1.6 double BACKTRACK_DATA::rho = 0.1

4.3.1.7 [Matrix](#)<double> BACKTRACK_DATA::xk

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.4 BiCGSTAB_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- int [maxit](#) = 0
- int [iter](#) = 0
- bool [breakdown](#)
- double [alpha](#)
- double [beta](#)
- double [rho](#)
- double [rho_old](#)
- double [omega](#)
- double [omega_old](#)
- double [tol_rel](#) = 1e-6
- double [tol_abs](#) = 1e-6
- double [res](#)
- double [relres](#)
- double [relres_base](#)
- double [bestres](#)
- bool [Output](#) = true
- [Matrix](#)< double > [x](#)

- [Matrix< double > bestx](#)
- [Matrix< double > r](#)
- [Matrix< double > r0](#)
- [Matrix< double > v](#)
- [Matrix< double > p](#)
- [Matrix< double > y](#)
- [Matrix< double > s](#)
- [Matrix< double > z](#)
- [Matrix< double > t](#)

4.4.1 Member Data Documentation

4.4.1.1 `double BiCGSTAB_DATA::alpha`

4.4.1.2 `double BiCGSTAB_DATA::bestres`

4.4.1.3 `Matrix<double> BiCGSTAB_DATA::bestx`

4.4.1.4 `double BiCGSTAB_DATA::beta`

4.4.1.5 `bool BiCGSTAB_DATA::breakdown`

4.4.1.6 `int BiCGSTAB_DATA::iter = 0`

4.4.1.7 `int BiCGSTAB_DATA::maxit = 0`

4.4.1.8 `double BiCGSTAB_DATA::omega`

4.4.1.9 `double BiCGSTAB_DATA::omega_old`

4.4.1.10 `bool BiCGSTAB_DATA::Output = true`

4.4.1.11 `Matrix<double> BiCGSTAB_DATA::p`

4.4.1.12 `Matrix<double> BiCGSTAB_DATA::r`

4.4.1.13 `Matrix<double> BiCGSTAB_DATA::r0`

4.4.1.14 `double BiCGSTAB_DATA::relres`

4.4.1.15 `double BiCGSTAB_DATA::relres_base`

4.4.1.16 `double BiCGSTAB_DATA::res`

4.4.1.17 `double BiCGSTAB_DATA::rho`

4.4.1.18 `double BiCGSTAB_DATA::rho_old`

4.4.1.19 `Matrix<double> BiCGSTAB_DATA::s`

4.4.1.20 `Matrix<double> BiCGSTAB_DATA::t`

4.4.1.21 `double BiCGSTAB_DATA::tol_abs = 1e-6`

4.4.1.22 `double BiCGSTAB_DATA::tol_rel = 1e-6`

4.4.1.23 **Matrix**<double> BiCGSTAB_DATA::v

4.4.1.24 **Matrix**<double> BiCGSTAB_DATA::x

4.4.1.25 **Matrix**<double> BiCGSTAB_DATA::y

4.4.1.26 **Matrix**<double> BiCGSTAB_DATA::z

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.5 CGS_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- int [maxit](#) = 0
- int [iter](#) = 0
- bool [breakdown](#)
- double [alpha](#)
- double [beta](#)
- double [rho](#)
- double [sigma](#)
- double [tol_rel](#) = 1e-6
- double [tol_abs](#) = 1e-6
- double [res](#)
- double [relres](#)
- double [relres_base](#)
- double [bestres](#)
- bool [Output](#) = true
- **Matrix**< double > [x](#)
- **Matrix**< double > [bestx](#)
- **Matrix**< double > [r](#)
- **Matrix**< double > [r0](#)
- **Matrix**< double > [u](#)
- **Matrix**< double > [w](#)
- **Matrix**< double > [v](#)
- **Matrix**< double > [p](#)
- **Matrix**< double > [c](#)
- **Matrix**< double > [z](#)

4.5.1 Member Data Documentation

4.5.1.1 double CGS_DATA::alpha

4.5.1.2 double CGS_DATA::bestres

4.5.1.3 **Matrix**<double> CGS_DATA::bestx

4.5.1.4 double CGS_DATA::beta

4.5.1.5 `bool CGS_DATA::breakdown`

4.5.1.6 `Matrix<double> CGS_DATA::c`

4.5.1.7 `int CGS_DATA::iter = 0`

4.5.1.8 `int CGS_DATA::maxit = 0`

4.5.1.9 `bool CGS_DATA::Output = true`

4.5.1.10 `Matrix<double> CGS_DATA::p`

4.5.1.11 `Matrix<double> CGS_DATA::r`

4.5.1.12 `Matrix<double> CGS_DATA::r0`

4.5.1.13 `double CGS_DATA::relres`

4.5.1.14 `double CGS_DATA::relres_base`

4.5.1.15 `double CGS_DATA::res`

4.5.1.16 `double CGS_DATA::rho`

4.5.1.17 `double CGS_DATA::sigma`

4.5.1.18 `double CGS_DATA::tol_abs = 1e-6`

4.5.1.19 `double CGS_DATA::tol_rel = 1e-6`

4.5.1.20 `Matrix<double> CGS_DATA::u`

4.5.1.21 `Matrix<double> CGS_DATA::v`

4.5.1.22 `Matrix<double> CGS_DATA::w`

4.5.1.23 `Matrix<double> CGS_DATA::x`

4.5.1.24 `Matrix<double> CGS_DATA::z`

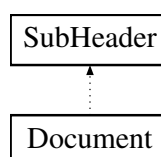
The documentation for this struct was generated from the following file:

- [lark.h](#)

4.6 Document Class Reference

```
#include <yaml_wrapper.h>
```

Inheritance diagram for Document:



Public Member Functions

- [Document](#) ()
- [~Document](#) ()
- [Document](#) (const [Document](#) &doc)
- [Document](#) (std::string [name](#))
- [Document](#) (const [KeyValueMap](#) &map)
- [Document](#) (std::string [name](#), const [KeyValueMap](#) &map)
- [Document](#) (std::string key, const [Header](#) &head)
- [Document](#) & [operator=](#) (const [Document](#) &doc)
- [ValueTypePair](#) & [operator\[\]](#) (const std::string key)
- [ValueTypePair](#) [operator\[\]](#) (const std::string key) const
- [Header](#) & [operator\(\)](#) (const std::string key)
- [Header](#) [operator\(\)](#) (const std::string key) const
- std::map< std::string, [Header](#) > & [getHeadMap](#) ()
- [KeyValueMap](#) & [getDataMap](#) ()
- [Header](#) & [getHeader](#) (std::string key)
- std::map< std::string, [Header](#) > ::const_iterator [end](#) () const
- std::map< std::string, [Header](#) > ::iterator [end](#) ()
- std::map< std::string, [Header](#) > ::const_iterator [begin](#) () const
- std::map< std::string, [Header](#) > ::iterator [begin](#) ()
- void [clear](#) ()
- void [resetKeys](#) ()
- void [changeKey](#) (std::string oldKey, std::string newKey)
- void [revalidateAllKeys](#) ()
- void [addPair](#) (std::string key, std::string val)
- void [addPair](#) (std::string key, std::string val, int t)
- void [setName](#) (std::string [name](#))
- void [setAlias](#) (std::string [alias](#))
- void [setNameAliasPair](#) (std::string n, std::string a, int s)
- void [setState](#) (int [state](#))
- void [DisplayContents](#) ()
- void [addHeadKey](#) (std::string key)
- void [copyAnchor2Alias](#) (std::string [alias](#), [Header](#) &ref)
- int [size](#) ()
- std::string [getName](#) ()
- std::string [getAlias](#) ()
- int [getState](#) ()
- bool [isAlias](#) ()
- bool [isAnchor](#) ()
- [Header](#) & [getAnchoredHeader](#) (std::string [alias](#))
- [Header](#) & [getHeadFromSubAlias](#) (std::string [alias](#))

Private Attributes

- std::map< std::string, [Header](#) > [Head_Map](#)

Additional Inherited Members

4.6.1 Constructor & Destructor Documentation

4.6.1.1 `Document::Document ()`

4.6.1.2 `Document::~~Document ()`

4.6.1.3 `Document::Document (const Document & doc)`

4.6.1.4 `Document::Document (std::string name)`

4.6.1.5 `Document::Document (const KeyValueType & map)`

4.6.1.6 `Document::Document (std::string name, const KeyValueType & map)`

4.6.1.7 `Document::Document (std::string key, const Header & head)`

4.6.2 Member Function Documentation

4.6.2.1 `void Document::addHeadKey (std::string key)`

4.6.2.2 `void Document::addPair (std::string key, std::string val)`

4.6.2.3 `void Document::addPair (std::string key, std::string val, int t)`

4.6.2.4 `std::map< std::string, Header >::const_iterator Document::begin () const`

4.6.2.5 `std::map< std::string, Header >::iterator Document::begin ()`

4.6.2.6 `void Document::changeKey (std::string oldKey, std::string newKey)`

4.6.2.7 `void Document::clear ()`

4.6.2.8 `void Document::copyAnchor2Alias (std::string alias, Header & ref)`

4.6.2.9 `void Document::DisplayContents ()`

4.6.2.10 `std::map< std::string, Header >::const_iterator Document::end () const`

4.6.2.11 `std::map< std::string, Header >::iterator Document::end ()`

4.6.2.12 `std::string Document::getAlias ()`

4.6.2.13 `Header & Document::getAnchoredHeader (std::string alias)`

4.6.2.14 `KeyValueType & Document::getDataMap ()`

4.6.2.15 `Header & Document::getHeader (std::string key)`

4.6.2.16 `Header & Document::getHeadFromSubAlias (std::string alias)`

4.6.2.17 `std::map< std::string, Header > & Document::getHeadMap ()`

4.6.2.18 `std::string Document::getName ()`

- 4.6.2.19 `int Document::getState ()`
- 4.6.2.20 `bool Document::isAlias ()`
- 4.6.2.21 `bool Document::isAnchor ()`
- 4.6.2.22 `Header & Document::operator() (const std::string key)`
- 4.6.2.23 `Header Document::operator() (const std::string key) const`
- 4.6.2.24 `Document & Document::operator= (const Document & doc)`
- 4.6.2.25 `ValueTypePair & Document::operator[] (const std::string key)`
- 4.6.2.26 `ValueTypePair Document::operator[] (const std::string key) const`
- 4.6.2.27 `void Document::resetKeys ()`
- 4.6.2.28 `void Document::revalidateAllKeys ()`
- 4.6.2.29 `void Document::setAlias (std::string alias)`
- 4.6.2.30 `void Document::setName (std::string name)`
- 4.6.2.31 `void Document::setNameAliasPair (std::string n, std::string a, int s)`
- 4.6.2.32 `void Document::setState (int state)`
- 4.6.2.33 `int Document::size ()`

4.6.3 Member Data Documentation

- 4.6.3.1 `std::map<std::string, Header> Document::Head_Map [private]`

The documentation for this class was generated from the following files:

- [yaml_wrapper.h](#)
- [yaml_wrapper.cpp](#)

4.7 DOGFISH_DATA Struct Reference

```
#include <dogfish.h>
```

Public Attributes

- unsigned long int `total_steps` = 0
- double `time_old` = 0.0
- double `time` = 0.0
- bool `Print2File` = true
- bool `Print2Console` = true
- bool `DirichletBC` = false
- bool `NonLinear` = false
- double `t_counter` = 0.0
- double `t_print`

- int [NumComp](#)
- double [end_time](#)
- double [total_sorption_old](#)
- double [total_sorption](#)
- double [fiber_length](#)
- double [fiber_diameter](#)
- FILE * [OutputFile](#)
- double(* [eval_R](#))(int i, int l, const void *data)
- double(* [eval_DI](#))(int i, int l, const void *data)
- double(* [eval_kf](#))(int i, const void *data)
- double(* [eval_qs](#))(int i, const void *data)
- const void * [user_data](#)
- std::vector< [FINCH_DATA](#) > [finch_dat](#)
- std::vector< [DOGFISH_PARAM](#) > [param_dat](#)

4.7.1 Member Data Documentation

4.7.1.1 bool DOGFISH_DATA::DirichletBC = false

4.7.1.2 double DOGFISH_DATA::end_time

4.7.1.3 double(* DOGFISH_DATA::eval_DI)(int i, int l, const void *data)

4.7.1.4 double(* DOGFISH_DATA::eval_kf)(int i, const void *data)

4.7.1.5 double(* DOGFISH_DATA::eval_qs)(int i, const void *data)

4.7.1.6 double(* DOGFISH_DATA::eval_R)(int i, int l, const void *data)

4.7.1.7 double DOGFISH_DATA::fiber_diameter

4.7.1.8 double DOGFISH_DATA::fiber_length

4.7.1.9 std::vector<FINCH_DATA> DOGFISH_DATA::finch_dat

4.7.1.10 bool DOGFISH_DATA::NonLinear = false

4.7.1.11 int DOGFISH_DATA::NumComp

4.7.1.12 FILE* DOGFISH_DATA::OutputFile

4.7.1.13 std::vector<DOGFISH_PARAM> DOGFISH_DATA::param_dat

4.7.1.14 bool DOGFISH_DATA::Print2Console = true

4.7.1.15 bool DOGFISH_DATA::Print2File = true

4.7.1.16 double DOGFISH_DATA::t_counter = 0.0

4.7.1.17 double DOGFISH_DATA::t_print

4.7.1.18 double DOGFISH_DATA::time = 0.0

4.7.1.19 double DOGFISH_DATA::time_old = 0.0

4.7.1.20 double DOGFISH_DATA::total_sorption

4.7.1.21 double DOGFISH_DATA::total_sorption_old

4.7.1.22 unsigned long int DOGFISH_DATA::total_steps = 0

4.7.1.23 const void* DOGFISH_DATA::user_data

The documentation for this struct was generated from the following file:

- [dogfish.h](#)

4.8 DOGFISH_PARAM Struct Reference

```
#include <dogfish.h>
```

Public Attributes

- double [intraparticle_diffusion](#)
- double [film_transfer_coeff](#)
- double [surface_concentration](#)
- double [initial_sorption](#)
- double [sorbed_molefraction](#)
- [Molecule species](#)

4.8.1 Member Data Documentation

4.8.1.1 double DOGFISH_PARAM::film_transfer_coeff

4.8.1.2 double DOGFISH_PARAM::initial_sorption

4.8.1.3 double DOGFISH_PARAM::intraparticle_diffusion

4.8.1.4 double DOGFISH_PARAM::sorbed_molefraction

4.8.1.5 Molecule DOGFISH_PARAM::species

4.8.1.6 double DOGFISH_PARAM::surface_concentration

The documentation for this struct was generated from the following file:

- [dogfish.h](#)

4.9 EX01_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- [Matrix](#)< double > [M](#)
- [Matrix](#)< double > [b](#)

4.9.1 Member Data Documentation

4.9.1.1 `Matrix<double> EX01_DATA::b`

4.9.1.2 `Matrix<double> EX01_DATA::M`

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.10 EX02_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- `Matrix< double > M`
- `Matrix< double > b`

4.10.1 Member Data Documentation

4.10.1.1 `Matrix<double> EX02_DATA::b`

4.10.1.2 `Matrix<double> EX02_DATA::M`

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.11 EX04_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- `Matrix< double > M`
- `Matrix< double > b`

4.11.1 Member Data Documentation

4.11.1.1 `Matrix<double> EX04_DATA::b`

4.11.1.2 `Matrix<double> EX04_DATA::M`

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.12 EX09_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- double [h](#)
- double [k](#)
- int [N](#)
- [Matrix](#)< double > [x](#)
- [Matrix](#)< double > [M](#)
- [Matrix](#)< double > [s](#)
- [Matrix](#)< double > [p](#)

4.12.1 Member Data Documentation

4.12.1.1 double EX09_DATA::h

4.12.1.2 double EX09_DATA::k

4.12.1.3 [Matrix](#)<double> EX09_DATA::M

4.12.1.4 int EX09_DATA::N

4.12.1.5 [Matrix](#)<double> EX09_DATA::p

4.12.1.6 [Matrix](#)<double> EX09_DATA::s

4.12.1.7 [Matrix](#)<double> EX09_DATA::x

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.13 EX15_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- int [N](#)
- int [m](#)
- [Matrix](#)< double > [b](#)

4.13.1 Member Data Documentation

4.13.1.1 [Matrix](#)<double> EX15_DATA::b

4.13.1.2 int EX15_DATA::m

4.13.1.3 int EX15_DATA::N

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.14 FINCH_DATA Struct Reference

```
#include <finch.h>
```

Public Attributes

- int [d](#) = 0
- double [dt](#) = 0.0125
- double [dt_old](#) = 0.0125
- double [T](#) = 1.0
- double [dz](#) = 0.1
- double [L](#) = 1.0
- double [s](#) = 1.0
- double [t](#) = 0.0
- double [t_old](#) = 0.0
- double [uT](#) = 0.0
- double [uT_old](#) = 0.0
- double [uAvg](#) = 0.0
- double [uAvg_old](#) = 0.0
- double [uIC](#) = 0.0
- double [vIC](#) = 1.0
- double [DIC](#) = 1.0
- double [kIC](#) = 1.0
- double [RIC](#) = 1.0
- double [uo](#) = 1.0
- double [vo](#) = 1.0
- double [Do](#) = 1.0
- double [ko](#) = 1.0
- double [Ro](#) = 1.0
- double [kfn](#) = 1.0
- double [kfnp1](#) = 1.0
- double [lambda_I](#)
- double [lambda_E](#)
- int [LN](#) = 10
- bool [CN](#) = true
- bool [Update](#) = false
- bool [Dirichlet](#) = false
- bool [CheckMass](#) = false
- bool [ExplicitFlux](#) = false
- bool [Iterative](#) = true
- bool [SteadyState](#) = false
- bool [NormTrack](#) = true
- double [beta](#) = 0.5
- double [tol_rel](#) = 1e-6
- double [tol_abs](#) = 1e-6
- int [max_iter](#) = 20
- int [total_iter](#) = 0

- int `nl_method` = `FINCH_Picard`
- `std::vector< double >` `CL_I`
- `std::vector< double >` `CL_E`
- `std::vector< double >` `CC_I`
- `std::vector< double >` `CC_E`
- `std::vector< double >` `CR_I`
- `std::vector< double >` `CR_E`
- `std::vector< double >` `fL_I`
- `std::vector< double >` `fL_E`
- `std::vector< double >` `fC_I`
- `std::vector< double >` `fC_E`
- `std::vector< double >` `fR_I`
- `std::vector< double >` `fR_E`
- `std::vector< double >` `OI`
- `std::vector< double >` `OE`
- `std::vector< double >` `NI`
- `std::vector< double >` `NE`
- `std::vector< double >` `MI`
- `std::vector< double >` `ME`
- `std::vector< double >` `uz_I_I`
- `std::vector< double >` `uz_lm1_I`
- `std::vector< double >` `uz_lp1_I`
- `std::vector< double >` `uz_I_E`
- `std::vector< double >` `uz_lm1_E`
- `std::vector< double >` `uz_lp1_E`
- `Matrix< double >` `unm1`
- `Matrix< double >` `un`
- `Matrix< double >` `unp1`
- `Matrix< double >` `u_star`
- `Matrix< double >` `ubest`
- `Matrix< double >` `vn`
- `Matrix< double >` `vnp1`
- `Matrix< double >` `Dn`
- `Matrix< double >` `Dnp1`
- `Matrix< double >` `kn`
- `Matrix< double >` `knp1`
- `Matrix< double >` `Sn`
- `Matrix< double >` `Snp1`
- `Matrix< double >` `Rn`
- `Matrix< double >` `Rnp1`
- `Matrix< double >` `Fn`
- `Matrix< double >` `Fnp1`
- `Matrix< double >` `gl`
- `Matrix< double >` `gE`
- `Matrix< double >` `res`
- `Matrix< double >` `pres`
- `int(* callroutine)(const void *user_data)`
- `int(* setic)(const void *user_data)`
- `int(* settime)(const void *user_data)`
- `int(* setpreprocess)(const void *user_data)`
- `int(* solve)(const void *user_data)`
- `int(* setparams)(const void *user_data)`
- `int(* discretize)(const void *user_data)`
- `int(* setbcs)(const void *user_data)`
- `int(* evalres)(const Matrix< double > &x, Matrix< double > &res, const void *user_data)`

- `int(* evalprecon)(const Matrix< double > &b, Matrix< double > &p, const void *user_data)`
- `int(* setpostprocess)(const void *user_data)`
- `int(* resettime)(const void *user_data)`
- `PICARD_DATA picard_dat`
- `PJFNK_DATA pjfnk_dat`
- `const void * param_data`

4.14.1 Member Data Documentation

4.14.1.1 `double FINCH_DATA::beta = 0.5`

4.14.1.2 `int(* FINCH_DATA::callroutine)(const void *user_data)`

4.14.1.3 `std::vector<double> FINCH_DATA::CC_E`

4.14.1.4 `std::vector<double> FINCH_DATA::CC_I`

4.14.1.5 `bool FINCH_DATA::CheckMass = false`

4.14.1.6 `std::vector<double> FINCH_DATA::CL_E`

4.14.1.7 `std::vector<double> FINCH_DATA::CL_I`

4.14.1.8 `bool FINCH_DATA::CN = true`

4.14.1.9 `std::vector<double> FINCH_DATA::CR_E`

4.14.1.10 `std::vector<double> FINCH_DATA::CR_I`

4.14.1.11 `int FINCH_DATA::d = 0`

4.14.1.12 `double FINCH_DATA::DIC = 1.0`

4.14.1.13 `bool FINCH_DATA::Dirichlet = false`

4.14.1.14 `int(* FINCH_DATA::discretize)(const void *user_data)`

4.14.1.15 `Matrix<double> FINCH_DATA::Dn`

4.14.1.16 `Matrix<double> FINCH_DATA::Dnp1`

4.14.1.17 `double FINCH_DATA::Do = 1.0`

4.14.1.18 `double FINCH_DATA::dt = 0.0125`

4.14.1.19 `double FINCH_DATA::dt_old = 0.0125`

4.14.1.20 `double FINCH_DATA::dz = 0.1`

4.14.1.21 `int(* FINCH_DATA::evalprecon)(const Matrix< double > &b, Matrix< double > &p, const void *user_data)`

4.14.1.22 `int(* FINCH_DATA::evalres)(const Matrix< double > &x, Matrix< double > &res, const void *user_data)`

4.14.1.23 `bool FINCH_DATA::ExplicitFlux = false`

- 4.14.1.24 `std::vector<double> FINCH_DATA::fC_E`
- 4.14.1.25 `std::vector<double> FINCH_DATA::fC_I`
- 4.14.1.26 `std::vector<double> FINCH_DATA::fL_E`
- 4.14.1.27 `std::vector<double> FINCH_DATA::fL_I`
- 4.14.1.28 `Matrix<double> FINCH_DATA::Fn`
- 4.14.1.29 `Matrix<double> FINCH_DATA::Fnp1`
- 4.14.1.30 `std::vector<double> FINCH_DATA::fR_E`
- 4.14.1.31 `std::vector<double> FINCH_DATA::fR_I`
- 4.14.1.32 `Matrix<double> FINCH_DATA::gE`
- 4.14.1.33 `Matrix<double> FINCH_DATA::gI`
- 4.14.1.34 `bool FINCH_DATA::iterative = true`
- 4.14.1.35 `double FINCH_DATA::kfn = 1.0`
- 4.14.1.36 `double FINCH_DATA::kfnp1 = 1.0`
- 4.14.1.37 `double FINCH_DATA::klC = 1.0`
- 4.14.1.38 `Matrix<double> FINCH_DATA::kn`
- 4.14.1.39 `Matrix<double> FINCH_DATA::knp1`
- 4.14.1.40 `double FINCH_DATA::ko = 1.0`
- 4.14.1.41 `double FINCH_DATA::L = 1.0`
- 4.14.1.42 `double FINCH_DATA::lambda_E`
- 4.14.1.43 `double FINCH_DATA::lambda_I`
- 4.14.1.44 `int FINCH_DATA::LN = 10`
- 4.14.1.45 `int FINCH_DATA::max_iter = 20`
- 4.14.1.46 `std::vector<double> FINCH_DATA::ME`
- 4.14.1.47 `std::vector<double> FINCH_DATA::MI`
- 4.14.1.48 `std::vector<double> FINCH_DATA::NE`
- 4.14.1.49 `std::vector<double> FINCH_DATA::NI`
- 4.14.1.50 `int FINCH_DATA::nl_method = FINCH_Picard`
- 4.14.1.51 `bool FINCH_DATA::NormTrack = true`

- 4.14.1.52 `std::vector<double> FINCH_DATA::OE`
- 4.14.1.53 `std::vector<double> FINCH_DATA::OI`
- 4.14.1.54 `const void* FINCH_DATA::param_data`
- 4.14.1.55 `PICARD_DATA FINCH_DATA::picard_dat`
- 4.14.1.56 `PJFNK_DATA FINCH_DATA::pjfnk_dat`
- 4.14.1.57 `Matrix<double> FINCH_DATA::pres`
- 4.14.1.58 `Matrix<double> FINCH_DATA::res`
- 4.14.1.59 `int(* FINCH_DATA::resetime)(const void *user_data)`
- 4.14.1.60 `double FINCH_DATA::RIC = 1.0`
- 4.14.1.61 `Matrix<double> FINCH_DATA::Rn`
- 4.14.1.62 `Matrix<double> FINCH_DATA::Rnp1`
- 4.14.1.63 `double FINCH_DATA::Ro = 1.0`
- 4.14.1.64 `double FINCH_DATA::s = 1.0`
- 4.14.1.65 `int(* FINCH_DATA::setbcs)(const void *user_data)`
- 4.14.1.66 `int(* FINCH_DATA::setic)(const void *user_data)`
- 4.14.1.67 `int(* FINCH_DATA::setparams)(const void *user_data)`
- 4.14.1.68 `int(* FINCH_DATA::setpostprocess)(const void *user_data)`
- 4.14.1.69 `int(* FINCH_DATA::setpreprocess)(const void *user_data)`
- 4.14.1.70 `int(* FINCH_DATA::settime)(const void *user_data)`
- 4.14.1.71 `Matrix<double> FINCH_DATA::Sn`
- 4.14.1.72 `Matrix<double> FINCH_DATA::Snp1`
- 4.14.1.73 `int(* FINCH_DATA::solve)(const void *user_data)`
- 4.14.1.74 `bool FINCH_DATA::SteadyState = false`
- 4.14.1.75 `double FINCH_DATA::T = 1.0`
- 4.14.1.76 `double FINCH_DATA::t = 0.0`
- 4.14.1.77 `double FINCH_DATA::t_old = 0.0`
- 4.14.1.78 `double FINCH_DATA::tol_abs = 1e-6`
- 4.14.1.79 `double FINCH_DATA::tol_rel = 1e-6`

- 4.14.1.80 `int FINCH_DATA::total_iter = 0`
- 4.14.1.81 `Matrix<double> FINCH_DATA::u_star`
- 4.14.1.82 `double FINCH_DATA::uAvg = 0.0`
- 4.14.1.83 `double FINCH_DATA::uAvg_old = 0.0`
- 4.14.1.84 `Matrix<double> FINCH_DATA::ubest`
- 4.14.1.85 `double FINCH_DATA::uIC = 0.0`
- 4.14.1.86 `Matrix<double> FINCH_DATA::un`
- 4.14.1.87 `Matrix<double> FINCH_DATA::unm1`
- 4.14.1.88 `Matrix<double> FINCH_DATA::unp1`
- 4.14.1.89 `double FINCH_DATA::uo = 1.0`
- 4.14.1.90 `bool FINCH_DATA::Update = false`
- 4.14.1.91 `double FINCH_DATA::uT = 0.0`
- 4.14.1.92 `double FINCH_DATA::uT_old = 0.0`
- 4.14.1.93 `std::vector<double> FINCH_DATA::uz_I_E`
- 4.14.1.94 `std::vector<double> FINCH_DATA::uz_I_I`
- 4.14.1.95 `std::vector<double> FINCH_DATA::uz_lm1_E`
- 4.14.1.96 `std::vector<double> FINCH_DATA::uz_lm1_I`
- 4.14.1.97 `std::vector<double> FINCH_DATA::uz_lp1_E`
- 4.14.1.98 `std::vector<double> FINCH_DATA::uz_lp1_I`
- 4.14.1.99 `double FINCH_DATA::vIC = 1.0`
- 4.14.1.100 `Matrix<double> FINCH_DATA::vn`
- 4.14.1.101 `Matrix<double> FINCH_DATA::vnp1`
- 4.14.1.102 `double FINCH_DATA::vo = 1.0`

The documentation for this struct was generated from the following file:

- [finch.h](#)

4.15 GCR_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- int `restart` = -1
- int `maxit` = 0
- int `iter_outer` = 0
- int `iter_inner` = 0
- int `total_iter` = 0
- bool `breakdown` = false
- double `alpha`
- double `beta`
- double `tol_rel` = 1e-6
- double `tol_abs` = 1e-6
- double `res`
- double `relres`
- double `relres_base`
- double `bestres`
- bool `Output` = true
- `Matrix< double >` `x`
- `Matrix< double >` `bestx`
- `Matrix< double >` `r`
- `Matrix< double >` `c_temp`
- `Matrix< double >` `u_temp`
- `std::vector< Matrix< double > >` `u`
- `std::vector< Matrix< double > >` `c`
- `OPTRANS_DATA` `transpose_dat`

4.15.1 Member Data Documentation

4.15.1.1 double `GCR_DATA::alpha`

4.15.1.2 double `GCR_DATA::bestres`

4.15.1.3 `Matrix<double>` `GCR_DATA::bestx`

4.15.1.4 double `GCR_DATA::beta`

4.15.1.5 bool `GCR_DATA::breakdown` = false

4.15.1.6 `std::vector<Matrix<double> >` `GCR_DATA::c`

4.15.1.7 `Matrix<double>` `GCR_DATA::c_temp`

4.15.1.8 int `GCR_DATA::iter_inner` = 0

4.15.1.9 int `GCR_DATA::iter_outer` = 0

4.15.1.10 int `GCR_DATA::maxit` = 0

4.15.1.11 bool `GCR_DATA::Output` = true

4.15.1.12 `Matrix<double>` `GCR_DATA::r`

4.15.1.13 double `GCR_DATA::relres`

4.15.1.14 double `GCR_DATA::relres_base`

- 4.15.1.15 double GCR_DATA::res
- 4.15.1.16 int GCR_DATA::restart = -1
- 4.15.1.17 double GCR_DATA::tol_abs = 1e-6
- 4.15.1.18 double GCR_DATA::tol_rel = 1e-6
- 4.15.1.19 int GCR_DATA::total_iter = 0
- 4.15.1.20 OPTRANS_DATA GCR_DATA::transpose_dat
- 4.15.1.21 std::vector<Matrix<double> > GCR_DATA::u
- 4.15.1.22 Matrix<double> GCR_DATA::u_temp
- 4.15.1.23 Matrix<double> GCR_DATA::x

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.16 GMRESLP_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- int [restart](#) = -1
- int [maxit](#) = 0
- int [iter](#) = 0
- int [steps](#) = 0
- double [tol_rel](#) = 1e-6
- double [tol_abs](#) = 1e-6
- double [res](#)
- double [relres](#)
- double [relres_base](#)
- double [bestres](#)
- bool [Output](#) = true
- Matrix< double > [x](#)
- Matrix< double > [bestx](#)
- Matrix< double > [r](#)
- ARNOLDI_DATA [arnoldi_dat](#)

4.16.1 Member Data Documentation

- 4.16.1.1 ARNOLDI_DATA GMRESLP_DATA::arnoldi_dat
- 4.16.1.2 double GMRESLP_DATA::bestres
- 4.16.1.3 Matrix<double> GMRESLP_DATA::bestx
- 4.16.1.4 int GMRESLP_DATA::iter = 0

- 4.16.1.5 int GMRESLP_DATA::maxit = 0
- 4.16.1.6 bool GMRESLP_DATA::Output = true
- 4.16.1.7 Matrix<double> GMRESLP_DATA::r
- 4.16.1.8 double GMRESLP_DATA::relres
- 4.16.1.9 double GMRESLP_DATA::relres_base
- 4.16.1.10 double GMRESLP_DATA::res
- 4.16.1.11 int GMRESLP_DATA::restart = -1
- 4.16.1.12 int GMRESLP_DATA::steps = 0
- 4.16.1.13 double GMRESLP_DATA::tol_abs = 1e-6
- 4.16.1.14 double GMRESLP_DATA::tol_rel = 1e-6
- 4.16.1.15 Matrix<double> GMRESLP_DATA::x

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.17 GMRESR_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- int [gcr_restart](#) = -1
- int [gcr_maxit](#) = 0
- int [gmres_restart](#) = -1
- int [gmres_maxit](#) = 1
- int [N](#)
- int [total_iter](#)
- int [iter_outer](#)
- int [iter_inner](#)
- bool [GCR_Output](#) = true
- bool [GMRES_Output](#) = false
- double [gmres_tol](#) = 0.1
- double [gcr_rel_tol](#) = 1e-6
- double [gcr_abs_tol](#) = 1e-6
- Matrix< double > [arg](#)
- GCR_DATA [gcr_dat](#)
- GMRESR_DATA [gmres_dat](#)
- int(* [matvec](#))(const Matrix< double > &x, Matrix< double > &Ax, const void *[matvec_data](#))
- int(* [terminal_precon](#))(const Matrix< double > &r, Matrix< double > &p, const void *[precon_data](#))
- const void * [matvec_data](#)
- const void * [term_precon](#)

4.17.1 Member Data Documentation

4.17.1.1 **Matrix<double>** GMRESR_DATA::arg

4.17.1.2 **double** GMRESR_DATA::gcr_abs_tol = 1e-6

4.17.1.3 **GCR_DATA** GMRESR_DATA::gcr_dat

4.17.1.4 **int** GMRESR_DATA::gcr_maxit = 0

4.17.1.5 **bool** GMRESR_DATA::GCR_Output = true

4.17.1.6 **double** GMRESR_DATA::gcr_rel_tol = 1e-6

4.17.1.7 **int** GMRESR_DATA::gcr_restart = -1

4.17.1.8 **GMRESRP_DATA** GMRESR_DATA::gmres_dat

4.17.1.9 **int** GMRESR_DATA::gmres_maxit = 1

4.17.1.10 **bool** GMRESR_DATA::GMRES_Output = false

4.17.1.11 **int** GMRESR_DATA::gmres_restart = -1

4.17.1.12 **double** GMRESR_DATA::gmres_tol = 0.1

4.17.1.13 **int** GMRESR_DATA::iter_inner

4.17.1.14 **int** GMRESR_DATA::iter_outer

4.17.1.15 **int**(* GMRESR_DATA::matvec)(const **Matrix< double >** &x, **Matrix< double >** &Ax, const void *matvec_data)

4.17.1.16 **const void*** GMRESR_DATA::matvec_data

4.17.1.17 **int** GMRESR_DATA::N

4.17.1.18 **const void*** GMRESR_DATA::term_precon

4.17.1.19 **int**(* GMRESR_DATA::terminal_precon)(const **Matrix< double >** &r, **Matrix< double >** &p, const void *precon_data)

4.17.1.20 **int** GMRESR_DATA::total_iter

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.18 GMRESRP_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- **int** [restart](#) = -1

- int `maxit` = 0
- int `iter_outer` = 0
- int `iter_inner` = 0
- int `iter_total` = 0
- double `tol_rel` = 1e-6
- double `tol_abs` = 1e-6
- double `res`
- double `relres`
- double `relres_base`
- double `bestres`
- bool `Output` = true
- `Matrix< double >` `x`
- `Matrix< double >` `bestx`
- `Matrix< double >` `r`
- `std::vector< Matrix< double > >` `Vk`
- `std::vector< std::vector< double > >` `H`
- `std::vector< std::vector< double > >` `H_bar`
- `std::vector< double >` `y`
- `std::vector< double >` `e0`
- `std::vector< double >` `e0_bar`
- `Matrix< double >` `w`
- `Matrix< double >` `v`
- `Matrix< double >` `sum`

4.18.1 Member Data Documentation

4.18.1.1 double `GMRESRP_DATA::bestres`

4.18.1.2 `Matrix<double>` `GMRESRP_DATA::bestx`

4.18.1.3 `std::vector< double >` `GMRESRP_DATA::e0`

4.18.1.4 `std::vector< double >` `GMRESRP_DATA::e0_bar`

4.18.1.5 `std::vector< std::vector< double > >` `GMRESRP_DATA::H`

4.18.1.6 `std::vector< std::vector< double > >` `GMRESRP_DATA::H_bar`

4.18.1.7 int `GMRESRP_DATA::iter_inner` = 0

4.18.1.8 int `GMRESRP_DATA::iter_outer` = 0

4.18.1.9 int `GMRESRP_DATA::iter_total` = 0

4.18.1.10 int `GMRESRP_DATA::maxit` = 0

4.18.1.11 bool `GMRESRP_DATA::Output` = true

4.18.1.12 `Matrix<double>` `GMRESRP_DATA::r`

4.18.1.13 double `GMRESRP_DATA::relres`

4.18.1.14 double `GMRESRP_DATA::relres_base`

- 4.18.1.15 double GMRESRP_DATA::res
- 4.18.1.16 int GMRESRP_DATA::restart = -1
- 4.18.1.17 Matrix<double> GMRESRP_DATA::sum
- 4.18.1.18 double GMRESRP_DATA::tol_abs = 1e-6
- 4.18.1.19 double GMRESRP_DATA::tol_rel = 1e-6
- 4.18.1.20 Matrix<double> GMRESRP_DATA::v
- 4.18.1.21 std::vector< Matrix<double> > GMRESRP_DATA::Vk
- 4.18.1.22 Matrix<double> GMRESRP_DATA::w
- 4.18.1.23 Matrix<double> GMRESRP_DATA::x
- 4.18.1.24 std::vector< double > GMRESRP_DATA::y

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.19 GPAST_DATA Struct Reference

```
#include <magpie.h>
```

Public Attributes

- double [x](#)
- double [y](#)
- double [He](#)
- double [q](#)
- std::vector< double > [gama_inf](#)
- double [qo](#)
- double [Plo](#)
- std::vector< double > [po](#)
- double [poi](#)
- bool [present](#)

4.19.1 Member Data Documentation

- 4.19.1.1 std::vector<double> GPAST_DATA::gama_inf
- 4.19.1.2 double GPAST_DATA::He
- 4.19.1.3 double GPAST_DATA::Plo
- 4.19.1.4 std::vector<double> GPAST_DATA::po
- 4.19.1.5 double GPAST_DATA::poi

4.19.1.6 `bool GPAST_DATA::present`

4.19.1.7 `double GPAST_DATA::q`

4.19.1.8 `double GPAST_DATA::qo`

4.19.1.9 `double GPAST_DATA::x`

4.19.1.10 `double GPAST_DATA::y`

The documentation for this struct was generated from the following file:

- [magpie.h](#)

4.20 GSTA_DATA Struct Reference

```
#include <magpie.h>
```

Public Attributes

- `double qmax`
- `int m`
- `std::vector< double > dHo`
- `std::vector< double > dSo`

4.20.1 Member Data Documentation

4.20.1.1 `std::vector<double> GSTA_DATA::dHo`

4.20.1.2 `std::vector<double> GSTA_DATA::dSo`

4.20.1.3 `int GSTA_DATA::m`

4.20.1.4 `double GSTA_DATA::qmax`

The documentation for this struct was generated from the following file:

- [magpie.h](#)

4.21 GSTA_OPT_DATA Struct Reference

```
#include <gsta_opt.h>
```

Public Attributes

- `int total_eval`
- `int n_par`
- `double qmax`
- `int iso`
- `std::vector< std::vector
< double > > Fobj`

- `std::vector< std::vector< double > > q`
- `std::vector< std::vector< double > > P`
- `std::vector< std::vector< double > > best_par`
- `std::vector< std::vector< double > > Kno`
- `std::vector< std::vector< std::vector< double > > > all_pars`
- `std::vector< std::vector< double > > norms`
- `std::vector< double > opt_qmax`

4.21.1 Member Data Documentation

4.21.1.1 `std::vector<std::vector<std::vector<double> > > GSTA_OPT_DATA::all_pars`

4.21.1.2 `std::vector<std::vector<double> > GSTA_OPT_DATA::best_par`

4.21.1.3 `std::vector<std::vector<double> > GSTA_OPT_DATA::Fobj`

4.21.1.4 `int GSTA_OPT_DATA::iso`

4.21.1.5 `std::vector<std::vector<double> > GSTA_OPT_DATA::Kno`

4.21.1.6 `int GSTA_OPT_DATA::n_par`

4.21.1.7 `std::vector<std::vector<double> > GSTA_OPT_DATA::norms`

4.21.1.8 `std::vector<double> GSTA_OPT_DATA::opt_qmax`

4.21.1.9 `std::vector<std::vector<double> > GSTA_OPT_DATA::P`

4.21.1.10 `std::vector<std::vector<double> > GSTA_OPT_DATA::q`

4.21.1.11 `double GSTA_OPT_DATA::qmax`

4.21.1.12 `int GSTA_OPT_DATA::total_eval`

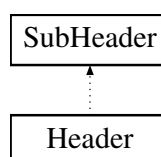
The documentation for this struct was generated from the following file:

- [gsta_opt.h](#)

4.22 Header Class Reference

```
#include <yaml_wrapper.h>
```

Inheritance diagram for Header:



Public Member Functions

- [Header](#) ()
- [~Header](#) ()
- [Header](#) (const [Header](#) &head)
- [Header](#) (std::string [name](#))
- [Header](#) (const [KeyValueType](#) &map)
- [Header](#) (std::string [name](#), const [KeyValueType](#) &map)
- [Header](#) (std::string key, const [SubHeader](#) &sub)
- [Header](#) & [operator=](#) (const [Header](#) &head)
- [ValueTypePair](#) & [operator\[\]](#) (const std::string key)
- [ValueTypePair](#) [operator\[\]](#) (const std::string key) const
- [SubHeader](#) & [operator\(\)](#) (const std::string key)
- [SubHeader](#) [operator\(\)](#) (const std::string key) const
- std::map< std::string, [SubHeader](#) > & [getSubMap](#) ()
- [KeyValueType](#) & [getDataMap](#) ()
- [SubHeader](#) & [getSubHeader](#) (std::string key)
- std::map< std::string, [SubHeader](#) >::const_iterator [end](#) () const
- std::map< std::string, [SubHeader](#) >::iterator [end](#) ()
- std::map< std::string, [SubHeader](#) >::const_iterator [begin](#) () const
- std::map< std::string, [SubHeader](#) >::iterator [begin](#) ()
- void [clear](#) ()
- void [resetKeys](#) ()
- void [changeKey](#) (std::string oldKey, std::string newKey)
- void [addPair](#) (std::string key, std::string val)
- void [addPair](#) (std::string key, std::string val, int t)
- void [setName](#) (std::string [name](#))
- void [setAlias](#) (std::string [alias](#))
- void [setNameAliasPair](#) (std::string n, std::string a, int s)
- void [setState](#) (int [state](#))
- void [DisplayContents](#) ()
- void [addSubKey](#) (std::string key)
- void [copyAnchor2Alias](#) (std::string [alias](#), [SubHeader](#) &ref)
- int [size](#) ()
- std::string [getName](#) ()
- std::string [getAlias](#) ()
- int [getState](#) ()
- bool [isAlias](#) ()
- bool [isAnchor](#) ()
- [SubHeader](#) & [getAnchoredSub](#) (std::string [alias](#))

Private Attributes

- std::map< std::string, [SubHeader](#) > [Sub_Map](#)

Additional Inherited Members

4.22.1 Constructor & Destructor Documentation

- 4.22.1.1 `Header::Header ()`
- 4.22.1.2 `Header::~Header ()`
- 4.22.1.3 `Header::Header (const Header & head)`
- 4.22.1.4 `Header::Header (std::string name)`
- 4.22.1.5 `Header::Header (const KeyValueType & map)`
- 4.22.1.6 `Header::Header (std::string name, const KeyValueType & map)`
- 4.22.1.7 `Header::Header (std::string key, const SubHeader & sub)`

4.22.2 Member Function Documentation

- 4.22.2.1 `void Header::addPair (std::string key, std::string val)`
- 4.22.2.2 `void Header::addPair (std::string key, std::string val, int t)`
- 4.22.2.3 `void Header::addSubKey (std::string key)`
- 4.22.2.4 `std::map< std::string, SubHeader >::const_iterator Header::begin () const`
- 4.22.2.5 `std::map< std::string, SubHeader >::iterator Header::begin ()`
- 4.22.2.6 `void Header::changeKey (std::string oldKey, std::string newKey)`
- 4.22.2.7 `void Header::clear ()`
- 4.22.2.8 `void Header::copyAnchor2Alias (std::string alias, SubHeader & ref)`
- 4.22.2.9 `void Header::DisplayContents ()`
- 4.22.2.10 `std::map< std::string, SubHeader >::const_iterator Header::end () const`
- 4.22.2.11 `std::map< std::string, SubHeader >::iterator Header::end ()`
- 4.22.2.12 `std::string Header::getAlias ()`
- 4.22.2.13 `SubHeader & Header::getAnchoredSub (std::string alias)`
- 4.22.2.14 `KeyValueType & Header::getDataMap ()`
- 4.22.2.15 `std::string Header::getName ()`
- 4.22.2.16 `int Header::getState ()`
- 4.22.2.17 `SubHeader & Header::getSubHeader (std::string key)`
- 4.22.2.18 `std::map< std::string, SubHeader > & Header::getSubMap ()`

- 4.22.2.19 `bool Header::isAlias ()`
- 4.22.2.20 `bool Header::isAnchor ()`
- 4.22.2.21 `SubHeader & Header::operator() (const std::string key)`
- 4.22.2.22 `SubHeader Header::operator() (const std::string key) const`
- 4.22.2.23 `Header & Header::operator= (const Header & head)`
- 4.22.2.24 `ValueTypePair & Header::operator[] (const std::string key)`
- 4.22.2.25 `ValueTypePair Header::operator[] (const std::string key) const`
- 4.22.2.26 `void Header::resetKeys ()`
- 4.22.2.27 `void Header::setAlias (std::string alias)`
- 4.22.2.28 `void Header::setName (std::string name)`
- 4.22.2.29 `void Header::setNameAliasPair (std::string n, std::string a, int s)`
- 4.22.2.30 `void Header::setState (int state)`
- 4.22.2.31 `int Header::size ()`

4.22.3 Member Data Documentation

- 4.22.3.1 `std::map<std::string, SubHeader> Header::Sub_Map` `[private]`

The documentation for this class was generated from the following files:

- [yaml_wrapper.h](#)
- [yaml_wrapper.cpp](#)

4.23 KeyValueType Class Reference

```
#include <yaml_wrapper.h>
```

Public Member Functions

- [KeyValueType](#) ()
- [~KeyValueType](#) ()
- [KeyValueType](#) (const std::map< std::string, std::string > &map)
- [KeyValueType](#) (std::string key, std::string value)
- [KeyValueType](#) (const [KeyValueType](#) &map)
- [KeyValueType](#) & [operator=](#) (const [KeyValueType](#) &map)
- [ValueTypePair](#) & [operator\[\]](#) (const std::string key)
- [ValueTypePair](#) [operator\[\]](#) (const std::string key) const
- std::map< std::string, [ValueTypePair](#) > & [getMap](#) ()
- std::map< std::string, [ValueTypePair](#) > ::const_iterator [end](#) () const

- `std::map< std::string, ValuePair >::iterator end ()`
- `std::map< std::string, ValuePair >::const_iterator begin () const`
- `std::map< std::string, ValuePair >::iterator begin ()`
- `void clear ()`
- `void addKey (std::string key)`
- `void editValue4Key (std::string val, std::string key)`
- `void editValue4Key (std::string val, int type, std::string key)`
- `void addPair (std::string key, ValuePair val)`
- `void addPair (std::string key, std::string val)`
- `void addPair (std::string key, std::string val, int type)`
- `void findType (std::string key)`
- `void assertType (std::string key, int type)`
- `void findAllTypes ()`
- `void DisplayMap ()`
- `int size ()`
- `std::string getString (std::string key)`
- `bool getBool (std::string key)`
- `double getDouble (std::string key)`
- `int getInt (std::string key)`
- `std::string getValue (std::string key)`
- `int getType (std::string key)`
- `ValuePair & getPair (std::string key)`

Private Attributes

- `std::map< std::string, ValuePair > Key_Value`

4.23.1 Constructor & Destructor Documentation

4.23.1.1 `KeyValueType::KeyValueType ()`

4.23.1.2 `KeyValueType::~~KeyValueType ()`

4.23.1.3 `KeyValueType::KeyValueType (const std::map< std::string, std::string > & map)`

4.23.1.4 `KeyValueType::KeyValueType (std::string key, std::string value)`

4.23.1.5 `KeyValueType::KeyValueType (const KeyValueType & map)`

4.23.2 Member Function Documentation

4.23.2.1 `void KeyValueType::addKey (std::string key)`

4.23.2.2 `void KeyValueType::addPair (std::string key, ValuePair val)`

4.23.2.3 `void KeyValueType::addPair (std::string key, std::string val)`

4.23.2.4 `void KeyValueType::addPair (std::string key, std::string val, int type)`

- 4.23.2.5 `void KeyValueType::assertType (std::string key, int type)`
- 4.23.2.6 `std::map< std::string, ValueTypePair >::const_iterator KeyValueType::begin () const`
- 4.23.2.7 `std::map< std::string, ValueTypePair >::iterator KeyValueType::begin ()`
- 4.23.2.8 `void KeyValueType::clear ()`
- 4.23.2.9 `void KeyValueType::DisplayMap ()`
- 4.23.2.10 `void KeyValueType::editValue4Key (std::string val, std::string key)`
- 4.23.2.11 `void KeyValueType::editValue4Key (std::string val, int type, std::string key)`
- 4.23.2.12 `std::map< std::string, ValueTypePair >::const_iterator KeyValueType::end () const`
- 4.23.2.13 `std::map< std::string, ValueTypePair >::iterator KeyValueType::end ()`
- 4.23.2.14 `void KeyValueType::findAllTypes ()`
- 4.23.2.15 `void KeyValueType::findType (std::string key)`
- 4.23.2.16 `bool KeyValueType::getBool (std::string key)`
- 4.23.2.17 `double KeyValueType::getDouble (std::string key)`
- 4.23.2.18 `int KeyValueType::getInt (std::string key)`
- 4.23.2.19 `std::map< std::string, ValueTypePair > & KeyValueType::getMap ()`
- 4.23.2.20 `ValueTypePair & KeyValueType::getPair (std::string key)`
- 4.23.2.21 `std::string KeyValueType::getString (std::string key)`
- 4.23.2.22 `int KeyValueType::getType (std::string key)`
- 4.23.2.23 `std::string KeyValueType::getValue (std::string key)`
- 4.23.2.24 `KeyValueType & KeyValueType::operator= (const KeyValueType & map)`
- 4.23.2.25 `ValueTypePair & KeyValueType::operator[] (const std::string key)`
- 4.23.2.26 `ValueTypePair KeyValueType::operator[] (const std::string key) const`
- 4.23.2.27 `int KeyValueType::size ()`

4.23.3 Member Data Documentation

- 4.23.3.1 `std::map<std::string, ValueTypePair > KeyValueType::Key_Value [private]`

The documentation for this class was generated from the following files:

- [yaml_wrapper.h](#)
- [yaml_wrapper.cpp](#)

4.24 MAGPIE_DATA Struct Reference

```
#include <magpie.h>
```

Public Attributes

- `std::vector< GSTA_DATA > gsta_dat`
- `std::vector< mSPD_DATA > mspd_dat`
- `std::vector< GPAST_DATA > gpast_dat`
- `SYSTEM_DATA sys_dat`

4.24.1 Member Data Documentation

4.24.1.1 `std::vector<GPAST_DATA> MAGPIE_DATA::gpast_dat`

4.24.1.2 `std::vector<GSTA_DATA> MAGPIE_DATA::gsta_dat`

4.24.1.3 `std::vector<mSPD_DATA> MAGPIE_DATA::mspd_dat`

4.24.1.4 `SYSTEM_DATA MAGPIE_DATA::sys_dat`

The documentation for this struct was generated from the following file:

- [magpie.h](#)

4.25 MassBalance Class Reference

```
#include <shark.h>
```

Public Member Functions

- [MassBalance](#) ()
- [~MassBalance](#) ()
- void [Initialize_List](#) ([MasterSpeciesList](#) &[List](#))
- void [Display_Info](#) ()
- void [Set_Delta](#) (int i, double v)
- void [Set_TotalConcentration](#) (double v)
- void [Set_Name](#) (std::string name)
- double [Get_Delta](#) (int i)
- double [Sum_Delta](#) ()
- double [Get_TotalConcentration](#) ()
- std::string [Get_Name](#) ()
- double [Eval_Residual](#) (const [Matrix](#)< double > &x)

Protected Attributes

- [MasterSpeciesList](#) * [List](#)
- std::vector< double > [Delta](#)
- double [TotalConcentration](#)

Private Attributes

- `std::string` [Name](#)

4.25.1 Constructor & Destructor Documentation

4.25.1.1 `MassBalance::MassBalance ()`

4.25.1.2 `MassBalance::~~MassBalance ()`

4.25.2 Member Function Documentation

4.25.2.1 `void MassBalance::Display_Info ()`

4.25.2.2 `double MassBalance::Eval_Residual (const Matrix< double > & x)`

4.25.2.3 `double MassBalance::Get_Delta (int i)`

4.25.2.4 `std::string MassBalance::Get_Name ()`

4.25.2.5 `double MassBalance::Get_TotalConcentration ()`

4.25.2.6 `void MassBalance::Initialize_List (MasterSpeciesList & List)`

4.25.2.7 `void MassBalance::Set_Delta (int i, double v)`

4.25.2.8 `void MassBalance::Set_Name (std::string name)`

4.25.2.9 `void MassBalance::Set_TotalConcentration (double v)`

4.25.2.10 `double MassBalance::Sum_Delta ()`

4.25.3 Member Data Documentation

4.25.3.1 `std::vector<double> MassBalance::Delta` [protected]

4.25.3.2 `MasterSpeciesList* MassBalance::List` [protected]

4.25.3.3 `std::string MassBalance::Name` [private]

4.25.3.4 `double MassBalance::TotalConcentration` [protected]

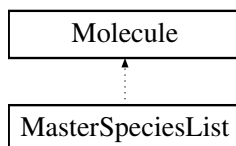
The documentation for this class was generated from the following files:

- [shark.h](#)
- [shark.cpp](#)

4.26 MasterSpeciesList Class Reference

```
#include <shark.h>
```

Inheritance diagram for MasterSpeciesList:



Public Member Functions

- `MasterSpeciesList ()`
- `~MasterSpeciesList ()`
- `MasterSpeciesList (const MasterSpeciesList &msl)`
- `MasterSpeciesList & operator= (const MasterSpeciesList &msl)`
- `void set_list_size (int i)`
- `void set_species (int i, std::string formula)`
- `void set_species (int i, int charge, double enthalpy, double entropy, double energy, bool HS, bool G, std::string Phase, std::string Name, std::string Formula, std::string lin_formula)`
- `void DisplayInfo (int i)`
- `void DisplayAll ()`
- `void DisplayConcentrations (Matrix< double > &C)`
- `void set_alkalinity (double alk)`
- `int list_size ()`
- `Molecule & get_species (int i)`
- `int get_index (std::string name)`
- `double charge (int i)`
- `double alkalinity ()`
- `std::string speciesName (int i)`
- `double Eval_ChargeResidual (const Matrix< double > &x)`

Protected Attributes

- `int size`
- `std::vector< Molecule > species`
- `double residual_alkalinity`

Additional Inherited Members

4.26.1 Constructor & Destructor Documentation

4.26.1.1 `MasterSpeciesList::MasterSpeciesList ()`

4.26.1.2 `MasterSpeciesList::~~MasterSpeciesList ()`

4.26.1.3 `MasterSpeciesList::MasterSpeciesList (const MasterSpeciesList & msl)`

4.26.2 Member Function Documentation

4.26.2.1 `double MasterSpeciesList::alkalinity ()`

4.26.2.2 `double MasterSpeciesList::charge (int i)`

4.26.2.3 `void MasterSpeciesList::DisplayAll ()`

- 4.26.2.4 void MasterSpeciesList::DisplayConcentrations (Matrix< double > & C)
- 4.26.2.5 void MasterSpeciesList::DisplayInfo (int i)
- 4.26.2.6 double MasterSpeciesList::Eval_ChargeResidual (const Matrix< double > & x)
- 4.26.2.7 int MasterSpeciesList::get_index (std::string name)
- 4.26.2.8 Molecule & MasterSpeciesList::get_species (int i)
- 4.26.2.9 int MasterSpeciesList::list_size ()
- 4.26.2.10 MasterSpeciesList & MasterSpeciesList::operator= (const MasterSpeciesList & msl)
- 4.26.2.11 void MasterSpeciesList::set_alkalinity (double alk)
- 4.26.2.12 void MasterSpeciesList::set_list_size (int i)
- 4.26.2.13 void MasterSpeciesList::set_species (int i, std::string formula)
- 4.26.2.14 void MasterSpeciesList::set_species (int i, int charge, double enthalpy, double entropy, double energy, bool HS, bool G, std::string Phase, std::string Name, std::string Formula, std::string lin_formula)
- 4.26.2.15 std::string MasterSpeciesList::speciesName (int i)

4.26.3 Member Data Documentation

- 4.26.3.1 double MasterSpeciesList::residual_alkalinity [protected]
- 4.26.3.2 int MasterSpeciesList::size [protected]
- 4.26.3.3 std::vector<Molecule> MasterSpeciesList::species [protected]

The documentation for this class was generated from the following files:

- [shark.h](#)
- [shark.cpp](#)

4.27 Matrix< T > Class Template Reference

```
#include <macaw.h>
```

Public Member Functions

- Matrix (int rows, int columns)
- T & operator() (int i, int j)
- T operator() (int i, int j) const
- Matrix (const Matrix &M)
- Matrix & operator= (const Matrix &M)
- Matrix ()
- ~Matrix ()
- void set_size (int i, int j)
- void zeros ()
- void edit (int i, int j, T value)

- int [rows](#) ()
- int [columns](#) ()
- T [determinate](#) ()
- T [norm](#) ()
- T [sum](#) ()
- T [inner_product](#) (const [Matrix](#) &x)
- [Matrix](#) & [cofactor](#) (const [Matrix](#) &M)
- [Matrix](#) [operator+](#) (const [Matrix](#) &M)
- [Matrix](#) [operator-](#) (const [Matrix](#) &M)
- [Matrix](#) [operator*](#) (const T)
- [Matrix](#) [operator/](#) (const T)
- [Matrix](#) [operator*](#) (const [Matrix](#) &M)
- [Matrix](#) & [transpose](#) (const [Matrix](#) &M)
- [Matrix](#) & [transpose_multiply](#) (const [Matrix](#) &MT, const [Matrix](#) &v)
- [Matrix](#) & [adjoint](#) (const [Matrix](#) &M)
- [Matrix](#) & [inverse](#) (const [Matrix](#) &M)
- void [Display](#) (const std::string Name)
- [Matrix](#) & [tridiagonalSolve](#) (const [Matrix](#) &A, const [Matrix](#) &b)
- [Matrix](#) & [ladshawSolve](#) (const [Matrix](#) &A, const [Matrix](#) &d)
- [Matrix](#) & [tridiagonalFill](#) (const T A, const T B, const T C, bool [Spherical](#))
- [Matrix](#) & [naturalLaplacian3D](#) (int m)
- [Matrix](#) & [sphericalBCFill](#) (int node, const T coeff, T variable)
- [Matrix](#) & [ConstantICFill](#) (const T IC)
- [Matrix](#) & [SolnTransform](#) (const [Matrix](#) &A, bool Forward)
- T [sphericalAvg](#) (double radius, double dr, double bound, bool Dirichlet)
- T [IntegralAvg](#) (double radius, double dr, double bound, bool Dirichlet)
- T [IntegralTotal](#) (double dr, double bound, bool Dirichlet)
- [Matrix](#) & [tridiagonalVectorFill](#) (const std::vector< T > &A, const std::vector< T > &B, const std::vector< T > &C)
- [Matrix](#) & [columnVectorFill](#) (const std::vector< T > &A)
- [Matrix](#) & [columnProjection](#) (const [Matrix](#) &b, const [Matrix](#) &b_old, const double dt, const double dt_old)
- [Matrix](#) & [dirichletBCFill](#) (int node, const T coeff, T variable)
- [Matrix](#) & [diagonalSolve](#) (const [Matrix](#) &D, const [Matrix](#) &v)
- [Matrix](#) & [upperTriangularSolve](#) (const [Matrix](#) &U, const [Matrix](#) &v)
- [Matrix](#) & [lowerTriangularSolve](#) (const [Matrix](#) &L, const [Matrix](#) &v)
- [Matrix](#) & [upperHessenberg2Triangular](#) ([Matrix](#) &b)
- [Matrix](#) & [lowerHessenberg2Triangular](#) ([Matrix](#) &b)
- [Matrix](#) & [upperHessenbergSolve](#) (const [Matrix](#) &H, const [Matrix](#) &v)
- [Matrix](#) & [lowerHessenbergSolve](#) (const [Matrix](#) &H, const [Matrix](#) &v)
- [Matrix](#) & [columnExtract](#) (int j, const [Matrix](#) &M)
- [Matrix](#) & [rowExtract](#) (int i, const [Matrix](#) &M)
- [Matrix](#) & [columnReplace](#) (int j, const [Matrix](#) &v)
- [Matrix](#) & [rowReplace](#) (int i, const [Matrix](#) &v)
- void [rowShrink](#) ()
- void [columnShrink](#) ()
- void [rowExtend](#) (const [Matrix](#) &v)
- void [columnExtend](#) (const [Matrix](#) &v)

Protected Attributes

- int [num_rows](#)
- int [num_cols](#)
- std::vector< T > [Data](#)

4.27.1 Constructor & Destructor Documentation

4.27.1.1 `template<class T> Matrix< T>::Matrix (int rows, int columns)`

4.27.1.2 `template<class T> Matrix< T>::Matrix (const Matrix< T> & M)`

4.27.1.3 `template<class T> Matrix< T>::Matrix ()`

4.27.1.4 `template<class T> Matrix< T>::~~Matrix ()`

4.27.2 Member Function Documentation

4.27.2.1 `template<class T> Matrix< T> & Matrix< T>::adjoint (const Matrix< T> & M)`

4.27.2.2 `template<class T> Matrix< T> & Matrix< T>::cofactor (const Matrix< T> & M)`

4.27.2.3 `template<class T> void Matrix< T>::columnExtend (const Matrix< T> & v)`

4.27.2.4 `template<class T> Matrix< T> & Matrix< T>::columnExtract (int j, const Matrix< T> & M)`

4.27.2.5 `template<class T> Matrix< T> & Matrix< T>::columnProjection (const Matrix< T> & b, const Matrix< T> & b_old, const double dt, const double dt_old)`

4.27.2.6 `template<class T> Matrix< T> & Matrix< T>::columnReplace (int j, const Matrix< T> & v)`

4.27.2.7 `template<class T> int Matrix< T>::columns ()`

4.27.2.8 `template<class T> void Matrix< T>::columnShrink ()`

4.27.2.9 `template<class T> Matrix< T> & Matrix< T>::columnVectorFill (const std::vector< T> & A)`

4.27.2.10 `template<class T> Matrix< T> & Matrix< T>::ConstantICFill (const T IC)`

4.27.2.11 `template<class T> T Matrix< T>::determinate ()`

4.27.2.12 `template<class T> Matrix< T> & Matrix< T>::diagonalSolve (const Matrix< T> & D, const Matrix< T> & v)`

4.27.2.13 `template<class T> Matrix< T> & Matrix< T>::dirichletBCFill (int node, const T coeff, T variable)`

4.27.2.14 `template<class T> void Matrix< T>::Display (const std::string Name)`

4.27.2.15 `template<class T> void Matrix< T>::edit (int i, int j, T value)`

4.27.2.16 `template<class T> T Matrix< T>::inner_product (const Matrix< T> & x)`

4.27.2.17 `template<class T> T Matrix< T>::IntegralAvg (double radius, double dr, double bound, bool Dirichlet)`

4.27.2.18 `template<class T> T Matrix< T>::IntegralTotal (double dr, double bound, bool Dirichlet)`

4.27.2.19 `template<class T> Matrix< T> & Matrix< T>::inverse (const Matrix< T> & M)`

4.27.2.20 `template<class T> Matrix< T> & Matrix< T>::ladshawSolve (const Matrix< T> & A, const Matrix< T> & d)`

4.27.2.21 `template<class T> Matrix< T> & Matrix< T>::lowerHessenberg2Triangular (Matrix< T> & b)`

- 4.27.2.22 `template<class T> Matrix< T > & Matrix< T >::lowerHessenbergSolve (const Matrix< T > & H, const Matrix< T > & v)`
- 4.27.2.23 `template<class T> Matrix< T > & Matrix< T >::lowerTriangularSolve (const Matrix< T > & L, const Matrix< T > & v)`
- 4.27.2.24 `template<class T> Matrix< T > & Matrix< T >::naturalLaplacian3D (int m)`
- 4.27.2.25 `template<class T> T Matrix< T >::norm ()`
- 4.27.2.26 `template<class T> T & Matrix< T >::operator() (int i, int j)`
- 4.27.2.27 `template<class T> T Matrix< T >::operator() (int i, int j) const`
- 4.27.2.28 `template<class T> Matrix< T > Matrix< T >::operator* (const T a)`
- 4.27.2.29 `template<class T> Matrix< T > Matrix< T >::operator* (const Matrix< T > & M)`
- 4.27.2.30 `template<class T> Matrix< T > Matrix< T >::operator+ (const Matrix< T > & M)`
- 4.27.2.31 `template<class T> Matrix< T > Matrix< T >::operator- (const Matrix< T > & M)`
- 4.27.2.32 `template<class T> Matrix< T > Matrix< T >::operator/ (const T a)`
- 4.27.2.33 `template<class T> Matrix< T > & Matrix< T >::operator= (const Matrix< T > & M)`
- 4.27.2.34 `template<class T> void Matrix< T >::rowExtend (const Matrix< T > & v)`
- 4.27.2.35 `template<class T> Matrix< T > & Matrix< T >::rowExtract (int i, const Matrix< T > & M)`
- 4.27.2.36 `template<class T> Matrix< T > & Matrix< T >::rowReplace (int i, const Matrix< T > & v)`
- 4.27.2.37 `template<class T> int Matrix< T >::rows ()`
- 4.27.2.38 `template<class T> void Matrix< T >::rowShrink ()`
- 4.27.2.39 `template<class T> void Matrix< T >::set_size (int i, int j)`
- 4.27.2.40 `template<class T> Matrix< T > & Matrix< T >::SolnTransform (const Matrix< T > & A, bool Forward)`
- 4.27.2.41 `template<class T> T Matrix< T >::sphericalAvg (double radius, double dr, double bound, bool Dirichlet)`
- 4.27.2.42 `template<class T> Matrix< T > & Matrix< T >::sphericalBCFill (int node, const T coeff, T variable)`
- 4.27.2.43 `template<class T> T Matrix< T >::sum ()`
- 4.27.2.44 `template<class T> Matrix< T > & Matrix< T >::transpose (const Matrix< T > & M)`
- 4.27.2.45 `template<class T> Matrix< T > & Matrix< T >::transpose_multiply (const Matrix< T > & MT, const Matrix< T > & v)`
- 4.27.2.46 `template<class T> Matrix< T > & Matrix< T >::tridiagonalFill (const T A, const T B, const T C, bool Spherical)`
- 4.27.2.47 `template<class T> Matrix< T > & Matrix< T >::tridiagonalSolve (const Matrix< T > & A, const Matrix< T > & b)`

- 4.27.2.48 `template<class T> Matrix< T > & Matrix< T >::tridiagonalVectorFill (const std::vector< T > & A, const std::vector< T > & B, const std::vector< T > & C)`
- 4.27.2.49 `template<class T> Matrix< T > & Matrix< T >::upperHessenberg2Triangular (Matrix< T > & b)`
- 4.27.2.50 `template<class T> Matrix< T > & Matrix< T >::upperHessenbergSolve (const Matrix< T > & H, const Matrix< T > & v)`
- 4.27.2.51 `template<class T> Matrix< T > & Matrix< T >::upperTriangularSolve (const Matrix< T > & U, const Matrix< T > & v)`
- 4.27.2.52 `template<class T> void Matrix< T >::zeros ()`

4.27.3 Member Data Documentation

- 4.27.3.1 `template<class T> std::vector<T> Matrix< T >::Data [protected]`
- 4.27.3.2 `template<class T> int Matrix< T >::num_cols [protected]`
- 4.27.3.3 `template<class T> int Matrix< T >::num_rows [protected]`

The documentation for this class was generated from the following file:

- [macaw.h](#)

4.28 Mechanism Class Reference

```
#include <shark.h>
```

Protected Attributes

- [MasterSpeciesList * List](#)
- `std::vector< UnsteadyReaction > reactions`
- `std::vector< double > weight`
- `int species_index`

4.28.1 Member Data Documentation

- 4.28.1.1 `MasterSpeciesList* Mechanism::List [protected]`
- 4.28.1.2 `std::vector<UnsteadyReaction> Mechanism::reactions [protected]`
- 4.28.1.3 `int Mechanism::species_index [protected]`
- 4.28.1.4 `std::vector<double> Mechanism::weight [protected]`

The documentation for this class was generated from the following file:

- [shark.h](#)

4.29 MIXED_GAS Struct Reference

```
#include <egret.h>
```

Public Attributes

- int [N](#)
- bool [CheckMolefractions](#) = true
- double [total_pressure](#)
- double [gas_temperature](#)
- double [velocity](#)
- double [char_length](#)
- std::vector< double > [molefraction](#)
- double [total_density](#)
- double [total_dyn_vis](#)
- double [kinematic_viscosity](#)
- double [total_molecular_weight](#)
- double [total_specific_heat](#)
- double [Reynolds](#)
- [Matrix](#)< double > [binary_diffusion](#)
- std::vector< [PURE_GAS](#) > [species_dat](#)

4.29.1 Member Data Documentation

- 4.29.1.1 [Matrix](#)<double> [MIXED_GAS::binary_diffusion](#)
- 4.29.1.2 double [MIXED_GAS::char_length](#)
- 4.29.1.3 bool [MIXED_GAS::CheckMolefractions](#) = true
- 4.29.1.4 double [MIXED_GAS::gas_temperature](#)
- 4.29.1.5 double [MIXED_GAS::kinematic_viscosity](#)
- 4.29.1.6 std::vector<double> [MIXED_GAS::molefraction](#)
- 4.29.1.7 int [MIXED_GAS::N](#)
- 4.29.1.8 double [MIXED_GAS::Reynolds](#)
- 4.29.1.9 std::vector<[PURE_GAS](#)> [MIXED_GAS::species_dat](#)
- 4.29.1.10 double [MIXED_GAS::total_density](#)
- 4.29.1.11 double [MIXED_GAS::total_dyn_vis](#)
- 4.29.1.12 double [MIXED_GAS::total_molecular_weight](#)
- 4.29.1.13 double [MIXED_GAS::total_pressure](#)
- 4.29.1.14 double [MIXED_GAS::total_specific_heat](#)
- 4.29.1.15 double [MIXED_GAS::velocity](#)

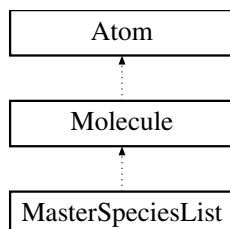
The documentation for this struct was generated from the following file:

- [egret.h](#)

4.30 Molecule Class Reference

```
#include <mola.h>
```

Inheritance diagram for Molecule:



Public Member Functions

- [Molecule](#) ()
- [~Molecule](#) ()
- [Molecule](#) (int [charge](#), double enthalpy, double entropy, double energy, bool HS, bool G, std::string [Phase](#), std::string [Name](#), std::string [Formula](#), std::string lin_formula)
- void [Register](#) (int [charge](#), double enthalpy, double entropy, double energy, bool HS, bool G, std::string [Phase](#), std::string [Name](#), std::string [Formula](#), std::string lin_formula)
- void [Register](#) (std::string formula)
- void [setFormula](#) (std::string form)
- void [recalculateMolarWeight](#) ()
- void [setMolarWeigth](#) (double mw)
- void [editCharge](#) (int c)
- void [editOneOxidationState](#) (int state, std::string [Symbol](#))
- void [editAllOxidationStates](#) (int state, std::string [Symbol](#))
- void [calculateAvgOxiState](#) (std::string [Symbol](#))
- void [editEnthalpy](#) (double enthalpy)
- void [editEntropy](#) (double entropy)
- void [editHS](#) (double H, double S)
- void [editEnergy](#) (double energy)
- void [removeOneAtom](#) (std::string [Symbol](#))
- void [removeAllAtoms](#) (std::string [Symbol](#))
- int [Charge](#) ()
- double [MolarWeight](#) ()
- bool [HaveHS](#) ()
- bool [HaveEnergy](#) ()
- bool [isRegistered](#) ()
- double [Enthalpy](#) ()
- double [Entropy](#) ()
- double [Energy](#) ()
- std::string [MoleculeName](#) ()
- std::string [MolecularFormula](#) ()
- std::string [MoleculePhase](#) ()
- void [DisplayInfo](#) ()

Protected Attributes

- int [charge](#)
- double [molar_weight](#)
- double [formation_enthalpy](#)
- double [formation_entropy](#)
- double [formation_energy](#)
- std::string [Phase](#)
- std::vector< [Atom](#) > [atoms](#)

Private Attributes

- std::string [Name](#)
- std::string [Formula](#)
- bool [haveG](#)
- bool [haveHS](#)
- bool [registered](#)

Additional Inherited Members

4.30.1 Constructor & Destructor Documentation

4.30.1.1 `Molecule::Molecule ()`

4.30.1.2 `Molecule::~~Molecule ()`

4.30.1.3 `Molecule::Molecule (int charge, double enthalpy, double entropy, double energy, bool HS, bool G, std::string Phase, std::string Name, std::string Formula, std::string lin_formula)`

4.30.2 Member Function Documentation

4.30.2.1 `void Molecule::calculateAvgOxiState (std::string Symbol)`

4.30.2.2 `int Molecule::Charge ()`

4.30.2.3 `void Molecule::DisplayInfo ()`

4.30.2.4 `void Molecule::editAllOxidationStates (int state, std::string Symbol)`

4.30.2.5 `void Molecule::editCharge (int c)`

4.30.2.6 `void Molecule::editEnergy (double energy)`

4.30.2.7 `void Molecule::editEnthalpy (double enthalpy)`

4.30.2.8 `void Molecule::editEntropy (double entropy)`

4.30.2.9 `void Molecule::editHS (double H, double S)`

4.30.2.10 `void Molecule::editOneOxidationState (int state, std::string Symbol)`

4.30.2.11 `double Molecule::Energy ()`

4.30.2.12 `double Molecule::Enthalpy ()`

- 4.30.2.13 `double Molecule::Entropy ()`
- 4.30.2.14 `bool Molecule::HaveEnergy ()`
- 4.30.2.15 `bool Molecule::HaveHS ()`
- 4.30.2.16 `bool Molecule::isRegistered ()`
- 4.30.2.17 `double Molecule::MolarWeight ()`
- 4.30.2.18 `std::string Molecule::MolecularFormula ()`
- 4.30.2.19 `std::string Molecule::MoleculeName ()`
- 4.30.2.20 `std::string Molecule::MoleculePhase ()`
- 4.30.2.21 `void Molecule::recalculateMolarWeight ()`
- 4.30.2.22 `void Molecule::Register (int charge, double enthalpy, double entropy, double energy, bool HS, bool G, std::string Phase, std::string Name, std::string Formula, std::string lin_formula)`
- 4.30.2.23 `void Molecule::Register (std::string formula)`
- 4.30.2.24 `void Molecule::removeAllAtoms (std::string Symbol)`
- 4.30.2.25 `void Molecule::removeOneAtom (std::string Symbol)`
- 4.30.2.26 `void Molecule::setFormula (std::string form)`
- 4.30.2.27 `void Molecule::setMolarWeigth (double mw)`

4.30.3 Member Data Documentation

- 4.30.3.1 `std::vector<Atom> Molecule::atoms` [protected]
- 4.30.3.2 `int Molecule::charge` [protected]
- 4.30.3.3 `double Molecule::formation_energy` [protected]
- 4.30.3.4 `double Molecule::formation_enthalpy` [protected]
- 4.30.3.5 `double Molecule::formation_entropy` [protected]
- 4.30.3.6 `std::string Molecule::Formula` [private]
- 4.30.3.7 `bool Molecule::haveG` [private]
- 4.30.3.8 `bool Molecule::haveHS` [private]
- 4.30.3.9 `double Molecule::molar_weight` [protected]
- 4.30.3.10 `std::string Molecule::Name` [private]
- 4.30.3.11 `std::string Molecule::Phase` [protected]

4.30.3.12 `bool Molecule::registered` `[private]`

The documentation for this class was generated from the following files:

- [mola.h](#)
- [mola.cpp](#)

4.31 MONKFISH_DATA Struct Reference

```
#include <monkfish.h>
```

Public Attributes

- unsigned long int `total_steps` = 0
- double `time_old` = 0.0
- double `time` = 0.0
- bool `Print2File` = true
- bool `Print2Console` = true
- bool `DirichletBC` = true
- bool `NonLinear` = false
- bool `haveMinMax` = false
- bool `MultiScale` = true
- int `level` = 2
- double `t_counter` = 0.0
- double `t_print`
- int `NumComp`
- double `end_time`
- double `total_sorption_old`
- double `total_sorption`
- double `single_fiber_density`
- double `avg_fiber_density`
- double `max_fiber_density`
- double `min_fiber_density`
- double `max_porosity`
- double `min_porosity`
- double `domain_diameter`
- FILE * `Output`
- double(* `eval_eps`)(int i, int l, const void *`user_data`)
- double(* `eval_rho`)(int i, int l, const void *`user_data`)
- double(* `eval_Dex`)(int i, int l, const void *`user_data`)
- double(* `eval_ads`)(int i, int l, const void *`user_data`)
- double(* `eval_Ret`)(int i, int l, const void *`user_data`)
- double(* `eval_Cex`)(int i, const void *`user_data`)
- double(* `eval_kf`)(int i, const void *`user_data`)
- const void * `user_data`
- std::vector< `FINCH_DATA` > `finch_dat`
- std::vector< `MONKFISH_PARAM` > `param_dat`
- std::vector< `DOGFISH_DATA` > `dog_dat`

4.31.1 Member Data Documentation

- 4.31.1.1 double MONKFISH_DATA::avg_fiber_density
- 4.31.1.2 bool MONKFISH_DATA::DirichletBC = true
- 4.31.1.3 std::vector<DOGFISH_DATA> MONKFISH_DATA::dog_dat
- 4.31.1.4 double MONKFISH_DATA::domain_diameter
- 4.31.1.5 double MONKFISH_DATA::end_time
- 4.31.1.6 double(* MONKFISH_DATA::eval_ads)(int i, int l, const void *user_data)
- 4.31.1.7 double(* MONKFISH_DATA::eval_Cex)(int i, const void *user_data)
- 4.31.1.8 double(* MONKFISH_DATA::eval_Dex)(int i, int l, const void *user_data)
- 4.31.1.9 double(* MONKFISH_DATA::eval_eps)(int i, int l, const void *user_data)
- 4.31.1.10 double(* MONKFISH_DATA::eval_kf)(int i, const void *user_data)
- 4.31.1.11 double(* MONKFISH_DATA::eval_Ret)(int i, int l, const void *user_data)
- 4.31.1.12 double(* MONKFISH_DATA::eval_rho)(int i, int l, const void *user_data)
- 4.31.1.13 std::vector<FINCH_DATA> MONKFISH_DATA::finch_dat
- 4.31.1.14 bool MONKFISH_DATA::haveMinMax = false
- 4.31.1.15 int MONKFISH_DATA::level = 2
- 4.31.1.16 double MONKFISH_DATA::max_fiber_density
- 4.31.1.17 double MONKFISH_DATA::max_porosity
- 4.31.1.18 double MONKFISH_DATA::min_fiber_density
- 4.31.1.19 double MONKFISH_DATA::min_porosity
- 4.31.1.20 bool MONKFISH_DATA::MultiScale = true
- 4.31.1.21 bool MONKFISH_DATA::NonLinear = false
- 4.31.1.22 int MONKFISH_DATA::NumComp
- 4.31.1.23 FILE* MONKFISH_DATA::Output
- 4.31.1.24 std::vector<MONKFISH_PARAM> MONKFISH_DATA::param_dat
- 4.31.1.25 bool MONKFISH_DATA::Print2Console = true
- 4.31.1.26 bool MONKFISH_DATA::Print2File = true
- 4.31.1.27 double MONKFISH_DATA::single_fiber_density

- 4.31.1.28 double MONKFISH_DATA::t_counter = 0.0
- 4.31.1.29 double MONKFISH_DATA::t_print
- 4.31.1.30 double MONKFISH_DATA::time = 0.0
- 4.31.1.31 double MONKFISH_DATA::time_old = 0.0
- 4.31.1.32 double MONKFISH_DATA::total_sorption
- 4.31.1.33 double MONKFISH_DATA::total_sorption_old
- 4.31.1.34 unsigned long int MONKFISH_DATA::total_steps = 0
- 4.31.1.35 const void* MONKFISH_DATA::user_data

The documentation for this struct was generated from the following file:

- [monkfish.h](#)

4.32 MONKFISH_PARAM Struct Reference

```
#include <monkfish.h>
```

Public Attributes

- double [interparticle_diffusion](#)
- double [exterior_concentration](#)
- double [exterior_transfer_coeff](#)
- double [sorbed_molefraction](#)
- double [initial_sorption](#)
- double [sorption_bc](#)
- double [intraparticle_diffusion](#)
- double [film_transfer_coeff](#)
- [Matrix](#)< double > [avg_sorption](#)
- [Matrix](#)< double > [avg_sorption_old](#)
- [Molecule species](#)

4.32.1 Member Data Documentation

- 4.32.1.1 [Matrix](#)<double> MONKFISH_PARAM::avg_sorption
- 4.32.1.2 [Matrix](#)<double> MONKFISH_PARAM::avg_sorption_old
- 4.32.1.3 double MONKFISH_PARAM::exterior_concentration
- 4.32.1.4 double MONKFISH_PARAM::exterior_transfer_coeff
- 4.32.1.5 double MONKFISH_PARAM::film_transfer_coeff
- 4.32.1.6 double MONKFISH_PARAM::initial_sorption
- 4.32.1.7 double MONKFISH_PARAM::interparticle_diffusion

4.32.1.8 double MONKFISH_PARAM::intraparticle_diffusion

4.32.1.9 double MONKFISH_PARAM::sorbed_molefraction

4.32.1.10 double MONKFISH_PARAM::sorption_bc

4.32.1.11 Molecule MONKFISH_PARAM::species

The documentation for this struct was generated from the following file:

- [monkfish.h](#)

4.33 mSPD_DATA Struct Reference

```
#include <magpie.h>
```

Public Attributes

- double [s](#)
- double [v](#)
- double [eMax](#)
- std::vector< double > [eta](#)
- double [gama](#)

4.33.1 Member Data Documentation

4.33.1.1 double mSPD_DATA::eMax

4.33.1.2 std::vector<double> mSPD_DATA::eta

4.33.1.3 double mSPD_DATA::gama

4.33.1.4 double mSPD_DATA::s

4.33.1.5 double mSPD_DATA::v

The documentation for this struct was generated from the following file:

- [magpie.h](#)

4.34 NUM_JAC_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- double [eps](#) = sqrt([DBL_EPSILON](#))
- [Matrix](#)< double > [Fx](#)
- [Matrix](#)< double > [Fxp](#)
- [Matrix](#)< double > [dxj](#)

4.34.1 Member Data Documentation

4.34.1.1 **Matrix**<double> NUM_JAC_DATA::dxj

4.34.1.2 double NUM_JAC_DATA::eps = sqrt(DBL_EPSILON)

4.34.1.3 **Matrix**<double> NUM_JAC_DATA::Fx

4.34.1.4 **Matrix**<double> NUM_JAC_DATA::Fxp

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.35 OPTRANS_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- **Matrix**< double > [li](#)
- **Matrix**< double > [Ai](#)

4.35.1 Member Data Documentation

4.35.1.1 **Matrix**<double> OPTRANS_DATA::Ai

4.35.1.2 **Matrix**<double> OPTRANS_DATA::li

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.36 PCG_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- int [maxit](#) = 0
- int [iter](#) = 0
- double [alpha](#)
- double [beta](#)
- double [tol_rel](#) = 1e-6
- double [tol_abs](#) = 1e-6
- double [res](#)
- double [relres](#)
- double [relres_base](#)
- double [bestres](#)
- bool [Output](#) = true
- **Matrix**< double > [x](#)

- [Matrix< double > bestx](#)
- [Matrix< double > r](#)
- [Matrix< double > r_old](#)
- [Matrix< double > z](#)
- [Matrix< double > z_old](#)
- [Matrix< double > p](#)
- [Matrix< double > Ap](#)

4.36.1 Member Data Documentation

4.36.1.1 `double PCG_DATA::alpha`

4.36.1.2 `Matrix<double> PCG_DATA::Ap`

4.36.1.3 `double PCG_DATA::bestres`

4.36.1.4 `Matrix<double> PCG_DATA::bestx`

4.36.1.5 `double PCG_DATA::beta`

4.36.1.6 `int PCG_DATA::iter = 0`

4.36.1.7 `int PCG_DATA::maxit = 0`

4.36.1.8 `bool PCG_DATA::Output = true`

4.36.1.9 `Matrix<double> PCG_DATA::p`

4.36.1.10 `Matrix<double> PCG_DATA::r`

4.36.1.11 `Matrix<double> PCG_DATA::r_old`

4.36.1.12 `double PCG_DATA::relres`

4.36.1.13 `double PCG_DATA::relres_base`

4.36.1.14 `double PCG_DATA::res`

4.36.1.15 `double PCG_DATA::tol_abs = 1e-6`

4.36.1.16 `double PCG_DATA::tol_rel = 1e-6`

4.36.1.17 `Matrix<double> PCG_DATA::x`

4.36.1.18 `Matrix<double> PCG_DATA::z`

4.36.1.19 `Matrix<double> PCG_DATA::z_old`

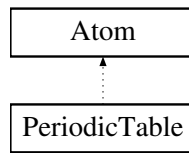
The documentation for this struct was generated from the following file:

- [lark.h](#)

4.37 PeriodicTable Class Reference

```
#include <eel.h>
```

Inheritance diagram for PeriodicTable:



Public Member Functions

- [PeriodicTable](#) ()
- [~PeriodicTable](#) ()
- [PeriodicTable](#) (int *n, int N)
- [PeriodicTable](#) (std::vector< std::string > &[Symbol](#))
- [PeriodicTable](#) (std::vector< int > &n)
- void [DisplayTable](#) ()

Protected Attributes

- std::vector< [Atom](#) > [Table](#)

Private Attributes

- int [number_elements](#)

Additional Inherited Members

4.37.1 Constructor & Destructor Documentation

4.37.1.1 [PeriodicTable::PeriodicTable](#) ()

4.37.1.2 [PeriodicTable::~PeriodicTable](#) ()

4.37.1.3 [PeriodicTable::PeriodicTable](#) (int * *n*, int *N*)

4.37.1.4 [PeriodicTable::PeriodicTable](#) (std::vector< std::string > & *Symbol*)

4.37.1.5 [PeriodicTable::PeriodicTable](#) (std::vector< int > & *n*)

4.37.2 Member Function Documentation

4.37.2.1 void [PeriodicTable::DisplayTable](#) ()

4.37.3 Member Data Documentation

4.37.3.1 int [PeriodicTable::number_elements](#) [private]

4.37.3.2 std::vector<[Atom](#)> [PeriodicTable::Table](#) [protected]

The documentation for this class was generated from the following files:

- [eel.h](#)
- [eel.cpp](#)

4.38 PICARD_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- int [maxit](#) = 0
- int [iter](#) = 0
- double [tol_rel](#) = 1e-6
- double [tol_abs](#) = 1e-6
- double [res](#)
- double [relres](#)
- double [relres_base](#)
- double [bestres](#)
- bool [Output](#) = true
- [Matrix](#)< double > [x0](#)
- [Matrix](#)< double > [bestx](#)
- [Matrix](#)< double > [r](#)

4.38.1 Member Data Documentation

4.38.1.1 double PICARD_DATA::bestres

4.38.1.2 [Matrix](#)<double> PICARD_DATA::bestx

4.38.1.3 int PICARD_DATA::iter = 0

4.38.1.4 int PICARD_DATA::maxit = 0

4.38.1.5 bool PICARD_DATA::Output = true

4.38.1.6 [Matrix](#)<double> PICARD_DATA::r

4.38.1.7 double PICARD_DATA::relres

4.38.1.8 double PICARD_DATA::relres_base

4.38.1.9 double PICARD_DATA::res

4.38.1.10 double PICARD_DATA::tol_abs = 1e-6

4.38.1.11 double PICARD_DATA::tol_rel = 1e-6

4.38.1.12 [Matrix](#)<double> PICARD_DATA::x0

The documentation for this struct was generated from the following file:

- [lark.h](#)

4.39 PJFNK_DATA Struct Reference

```
#include <lark.h>
```

Public Attributes

- int [nl_iter](#) = 0
- int [l_iter](#) = 0
- int [nl_maxit](#) = 0
- int [linear_solver](#) = -1
- double [nl_tol_abs](#) = 1e-6
- double [nl_tol_rel](#) = 1e-6
- double [lin_tol_rel](#) = 1e-6
- double [lin_tol_abs](#) = 1e-6
- double [nl_res](#)
- double [nl_relres](#)
- double [nl_res_base](#)
- double [nl_bestres](#)
- double [eps](#) = sqrt(DBL_EPSILON)
- bool [NL_Output](#) = true
- bool [L_Output](#) = false
- bool [LineSearch](#) = false
- bool [Bounce](#) = false
- [Matrix](#)< double > [F](#)
- [Matrix](#)< double > [Fv](#)
- [Matrix](#)< double > [v](#)
- [Matrix](#)< double > [x](#)
- [Matrix](#)< double > [bestx](#)
- [GMRESLP_DATA](#) [gmreslp_dat](#)
- [PCG_DATA](#) [pcg_dat](#)
- [BiCGSTAB_DATA](#) [bicgstab_dat](#)
- [CGS_DATA](#) [cgs_dat](#)
- [GMRESRP_DATA](#) [gmresrp_dat](#)
- [GCR_DATA](#) [gcr_dat](#)
- [GMRESR_DATA](#) [gmresr_dat](#)
- [BACKTRACK_DATA](#) [backtrack_dat](#)
- const void * [res_data](#)
- const void * [precon_data](#)
- int(* [funeval](#))(const [Matrix](#)< double > &[x](#), [Matrix](#)< double > &[F](#), const void *[res_data](#))
- int(* [precon](#))(const [Matrix](#)< double > &[r](#), [Matrix](#)< double > &[p](#), const void *[precon_data](#))

4.39.1 Member Data Documentation

4.39.1.1 [BACKTRACK_DATA](#) PJFNK_DATA::backtrack_dat4.39.1.2 [Matrix](#)<double> PJFNK_DATA::bestx4.39.1.3 [BiCGSTAB_DATA](#) PJFNK_DATA::bicgstab_dat

4.39.1.4 bool PJFNK_DATA::Bounce = false

4.39.1.5 [CGS_DATA](#) PJFNK_DATA::cgs_dat

4.39.1.6 double PJFNK_DATA::eps = sqrt(DBL_EPSILON)

4.39.1.7 [Matrix](#)<double> PJFNK_DATA::F4.39.1.8 int(* PJFNK_DATA::funeval)(const [Matrix](#)< double > &[x](#), [Matrix](#)< double > &[F](#), const void *[res_data](#))

- 4.39.1.9 **Matrix<double>** PJFNK_DATA::Fv
- 4.39.1.10 **GCR_DATA** PJFNK_DATA::gcr_dat
- 4.39.1.11 **GMRESLP_DATA** PJFNK_DATA::gmreslp_dat
- 4.39.1.12 **GMRESR_DATA** PJFNK_DATA::gmresr_dat
- 4.39.1.13 **GMRESRP_DATA** PJFNK_DATA::gmresrp_dat
- 4.39.1.14 **int** PJFNK_DATA::l_iter = 0
- 4.39.1.15 **bool** PJFNK_DATA::L_Output = false
- 4.39.1.16 **double** PJFNK_DATA::lin_tol_abs = 1e-6
- 4.39.1.17 **double** PJFNK_DATA::lin_tol_rel = 1e-6
- 4.39.1.18 **int** PJFNK_DATA::linear_solver = -1
- 4.39.1.19 **bool** PJFNK_DATA::LineSearch = false
- 4.39.1.20 **double** PJFNK_DATA::nl_bestres
- 4.39.1.21 **int** PJFNK_DATA::nl_iter = 0
- 4.39.1.22 **int** PJFNK_DATA::nl_maxit = 0
- 4.39.1.23 **bool** PJFNK_DATA::NL_Output = true
- 4.39.1.24 **double** PJFNK_DATA::nl_relres
- 4.39.1.25 **double** PJFNK_DATA::nl_res
- 4.39.1.26 **double** PJFNK_DATA::nl_res_base
- 4.39.1.27 **double** PJFNK_DATA::nl_tol_abs = 1e-6
- 4.39.1.28 **double** PJFNK_DATA::nl_tol_rel = 1e-6
- 4.39.1.29 **PCG_DATA** PJFNK_DATA::pcg_dat
- 4.39.1.30 **int**(* PJFNK_DATA::precon)(const **Matrix< double >** &r, **Matrix< double >** &p, const void *precon_data)
- 4.39.1.31 **const void*** PJFNK_DATA::precon_data
- 4.39.1.32 **const void*** PJFNK_DATA::res_data
- 4.39.1.33 **Matrix<double>** PJFNK_DATA::v
- 4.39.1.34 **Matrix<double>** PJFNK_DATA::x

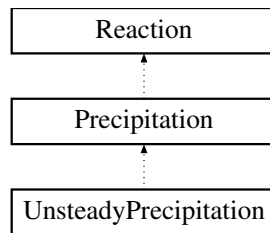
The documentation for this struct was generated from the following file:

- [lark.h](#)

4.40 Precipitation Class Reference

```
#include <shark.h>
```

Inheritance diagram for Precipitation:



Additional Inherited Members

The documentation for this class was generated from the following file:

- [shark.h](#)

4.41 PURE_GAS Struct Reference

```
#include <egret.h>
```

Public Attributes

- double [molecular_weight](#)
- double [Sutherland_Temp](#)
- double [Sutherland_Const](#)
- double [Sutherland_Viscosity](#)
- double [specific_heat](#)
- double [molecular_diffusion](#)
- double [dynamic_viscosity](#)
- double [density](#)
- double [Schmidt](#)

4.41.1 Member Data Documentation

4.41.1.1 double PURE_GAS::density

4.41.1.2 double PURE_GAS::dynamic_viscosity

4.41.1.3 double PURE_GAS::molecular_diffusion

4.41.1.4 double PURE_GAS::molecular_weight

4.41.1.5 double PURE_GAS::Schmidt

4.41.1.6 double PURE_GAS::specific_heat

4.41.1.7 double PURE_GAS::Sutherland_Const

4.41.1.8 double PURE_GAS::Sutherland_Temp

4.41.1.9 double PURE_GAS::Sutherland_Viscosity

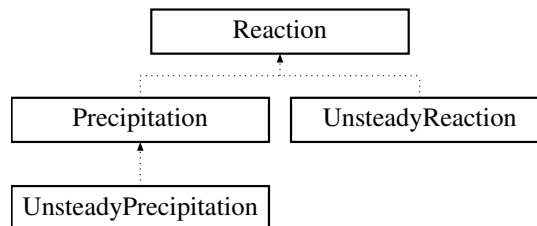
The documentation for this struct was generated from the following file:

- [egret.h](#)

4.42 Reaction Class Reference

```
#include <shark.h>
```

Inheritance diagram for Reaction:



Public Member Functions

- [Reaction](#) ()
- [~Reaction](#) ()
- void [Initialize_List](#) (MasterSpeciesList &List)
- void [Display_Info](#) ()
- void [Set_Stoichiometric](#) (int i, double v)
- void [Set_Equilibrium](#) (double v)
- void [Set_Enthalpy](#) (double H)
- void [Set_Entropy](#) (double S)
- void [Set_EnthalpyANDEntropy](#) (double H, double S)
- void [Set_Energy](#) (double G)
- void [checkSpeciesEnergies](#) ()
- void [calculateEnergies](#) ()
- void [calculateEquilibrium](#) (double T)
- bool [haveEquilibrium](#) ()
- double [Get_Stoichiometric](#) (int i)
- double [Get_Equilibrium](#) ()
- double [Get_Enthalpy](#) ()
- double [Get_Entropy](#) ()
- double [Get_Energy](#) ()
- double [Eval_Residual](#) (const [Matrix](#)< double > &x, const [Matrix](#)< double > &gama)

Protected Attributes

- [MasterSpeciesList](#) * List
- std::vector< double > [Stoichiometric](#)
- double [Equilibrium](#)
- double [enthalpy](#)
- double [entropy](#)

- double [energy](#)
- bool [CanCalcHS](#)
- bool [CanCalcG](#)
- bool [HaveHS](#)
- bool [HaveG](#)
- bool [HaveEquil](#)

4.42.1 Constructor & Destructor Documentation

4.42.1.1 `Reaction::Reaction ()`

4.42.1.2 `Reaction::~~Reaction ()`

4.42.2 Member Function Documentation

4.42.2.1 `void Reaction::calculateEnergies ()`

4.42.2.2 `void Reaction::calculateEquilibrium (double T)`

4.42.2.3 `void Reaction::checkSpeciesEnergies ()`

4.42.2.4 `void Reaction::Display_Info ()`

4.42.2.5 `double Reaction::Eval_Residual (const Matrix< double > & x, const Matrix< double > & gama)`

4.42.2.6 `double Reaction::Get_Energy ()`

4.42.2.7 `double Reaction::Get_Enthalpy ()`

4.42.2.8 `double Reaction::Get_Entropy ()`

4.42.2.9 `double Reaction::Get_Equilibrium ()`

4.42.2.10 `double Reaction::Get_Stoichiometric (int i)`

4.42.2.11 `bool Reaction::haveEquilibrium ()`

4.42.2.12 `void Reaction::Initialize_List (MasterSpeciesList & List)`

4.42.2.13 `void Reaction::Set_Energy (double G)`

4.42.2.14 `void Reaction::Set_Enthalpy (double H)`

4.42.2.15 `void Reaction::Set_EnthalpyANDEntropy (double H, double S)`

4.42.2.16 `void Reaction::Set_Entropy (double S)`

4.42.2.17 `void Reaction::Set_Equilibrium (double v)`

4.42.2.18 `void Reaction::Set_Stoichiometric (int i, double v)`

4.42.3 Member Data Documentation

4.42.3.1 `bool Reaction::CanCalcG` `[protected]`

- 4.42.3.2 `bool Reaction::CanCalcHS` [protected]
- 4.42.3.3 `double Reaction::energy` [protected]
- 4.42.3.4 `double Reaction::enthalpy` [protected]
- 4.42.3.5 `double Reaction::entropy` [protected]
- 4.42.3.6 `double Reaction::Equilibrium` [protected]
- 4.42.3.7 `bool Reaction::HaveEquil` [protected]
- 4.42.3.8 `bool Reaction::HaveG` [protected]
- 4.42.3.9 `bool Reaction::HaveHS` [protected]
- 4.42.3.10 `MasterSpeciesList* Reaction::List` [protected]
- 4.42.3.11 `std::vector<double> Reaction::Stoichiometric` [protected]

The documentation for this class was generated from the following files:

- [shark.h](#)
- [shark.cpp](#)

4.43 SCOPSOWL_DATA Struct Reference

```
#include <scopsowl.h>
```

Public Attributes

- unsigned long int [total_steps](#)
- int [coord_macro](#)
- int [coord_micro](#)
- int [level](#) = 2
- double [sim_time](#)
- double [t_old](#)
- double [t](#)
- double [t_counter](#) = 0.0
- double [t_print](#)
- bool [Print2File](#) = true
- bool [Print2Console](#) = true
- bool [SurfDiff](#) = true
- bool [Heterogeneous](#) = true
- double [gas_velocity](#)
- double [total_pressure](#)
- double [gas_temperature](#)
- double [pellet_radius](#)
- double [crystal_radius](#)
- double [char_macro](#)
- double [char_micro](#)
- double [binder_fraction](#)
- double [binder_porosity](#)

- double [binder_poresize](#)
- double [pellet_density](#)
- bool [DirichletBC](#) = false
- bool [NonLinear](#) = true
- std::vector< double > [y](#)
- std::vector< double > [tempy](#)
- FILE * [OutputFile](#)
- double(* [eval_ads](#))(int i, int l, const void *[user_data](#))
- double(* [eval_retard](#))(int i, int l, const void *[user_data](#))
- double(* [eval_diff](#))(int i, int l, const void *[user_data](#))
- double(* [eval_surfDiff](#))(int i, int l, const void *[user_data](#))
- double(* [eval_kf](#))(int i, const void *[user_data](#))
- const void * [user_data](#)
- [MIXED_GAS](#) * [gas_dat](#)
- [MAGPIE_DATA](#) [magpie_dat](#)
- std::vector< [FINCH_DATA](#) > [finch_dat](#)
- std::vector< [SCOPSOWL_PARAM_DATA](#) > [param_dat](#)
- std::vector< [SKUA_DATA](#) > [skua_dat](#)

4.43.1 Member Data Documentation

- 4.43.1.1 double SCOPSOWL_DATA::binder_fraction
- 4.43.1.2 double SCOPSOWL_DATA::binder_poresize
- 4.43.1.3 double SCOPSOWL_DATA::binder_porosity
- 4.43.1.4 double SCOPSOWL_DATA::char_macro
- 4.43.1.5 double SCOPSOWL_DATA::char_micro
- 4.43.1.6 int SCOPSOWL_DATA::coord_macro
- 4.43.1.7 int SCOPSOWL_DATA::coord_micro
- 4.43.1.8 double SCOPSOWL_DATA::crystal_radius
- 4.43.1.9 bool SCOPSOWL_DATA::DirichletBC = false
- 4.43.1.10 double(* SCOPSOWL_DATA::eval_ads)(int i, int l, const void *[user_data](#))
- 4.43.1.11 double(* SCOPSOWL_DATA::eval_diff)(int i, int l, const void *[user_data](#))
- 4.43.1.12 double(* SCOPSOWL_DATA::eval_kf)(int i, const void *[user_data](#))
- 4.43.1.13 double(* SCOPSOWL_DATA::eval_retard)(int i, int l, const void *[user_data](#))
- 4.43.1.14 double(* SCOPSOWL_DATA::eval_surfDiff)(int i, int l, const void *[user_data](#))
- 4.43.1.15 std::vector<[FINCH_DATA](#)> SCOPSOWL_DATA::finch_dat
- 4.43.1.16 [MIXED_GAS](#)* SCOPSOWL_DATA::gas_dat
- 4.43.1.17 double SCOPSOWL_DATA::gas_temperature

4.43.1.18 double SCOPSOWL_DATA::gas_velocity

4.43.1.19 bool SCOPSOWL_DATA::Heterogeneous = true

4.43.1.20 int SCOPSOWL_DATA::level = 2

4.43.1.21 MAGPIE_DATA SCOPSOWL_DATA::magpie_dat

4.43.1.22 bool SCOPSOWL_DATA::NonLinear = true

4.43.1.23 FILE* SCOPSOWL_DATA::OutputFile

4.43.1.24 std::vector<SCOPSOWL_PARAM_DATA> SCOPSOWL_DATA::param_dat

4.43.1.25 double SCOPSOWL_DATA::pellet_density

4.43.1.26 double SCOPSOWL_DATA::pellet_radius

4.43.1.27 bool SCOPSOWL_DATA::Print2Console = true

4.43.1.28 bool SCOPSOWL_DATA::Print2File = true

4.43.1.29 double SCOPSOWL_DATA::sim_time

4.43.1.30 std::vector<SKUA_DATA> SCOPSOWL_DATA::skua_dat

4.43.1.31 bool SCOPSOWL_DATA::SurfDiff = true

4.43.1.32 double SCOPSOWL_DATA::t

4.43.1.33 double SCOPSOWL_DATA::t_counter = 0.0

4.43.1.34 double SCOPSOWL_DATA::t_old

4.43.1.35 double SCOPSOWL_DATA::t_print

4.43.1.36 std::vector<double> SCOPSOWL_DATA::tempy

4.43.1.37 double SCOPSOWL_DATA::total_pressure

4.43.1.38 unsigned long int SCOPSOWL_DATA::total_steps

4.43.1.39 const void* SCOPSOWL_DATA::user_data

4.43.1.40 std::vector<double> SCOPSOWL_DATA::y

The documentation for this struct was generated from the following file:

- [scopsowl.h](#)

4.44 SCOPSOWL_OPT_DATA Struct Reference

```
#include <scopsowl_opt.h>
```

Public Attributes

- int [num_curves](#)
- int [evaluation](#)
- unsigned long int [total_eval](#)
- int [current_points](#)
- int [num_params](#) = 1
- int [diffusion_type](#)
- int [adsorb_index](#)
- int [max_guess_iter](#) = 20
- bool [Optimize](#)
- bool [Rough](#)
- double [current_temp](#)
- double [current_press](#)
- double [current_equil](#)
- double [simulation_equil](#)
- double [max_bias](#)
- double [min_bias](#)
- double [e_norm](#)
- double [f_bias](#)
- double [e_norm_old](#)
- double [f_bias_old](#)
- double [param_guess](#)
- double [param_guess_old](#)
- double [rel_tol_norm](#) = 0.01
- double [abs_tol_bias](#) = 1.0
- std::vector< double > [y_base](#)
- std::vector< double > [q_data](#)
- std::vector< double > [q_sim](#)
- std::vector< double > [t](#)
- FILE * [ParamFile](#)
- FILE * [CompareFile](#)
- SCOPSOWL_DATA owl_dat

4.44.1 Member Data Documentation

4.44.1.1 double SCOPSOWL_OPT_DATA::abs_tol_bias = 1.0

4.44.1.2 int SCOPSOWL_OPT_DATA::adsorb_index

4.44.1.3 FILE* SCOPSOWL_OPT_DATA::CompareFile

4.44.1.4 double SCOPSOWL_OPT_DATA::current_equil

4.44.1.5 int SCOPSOWL_OPT_DATA::current_points

4.44.1.6 double SCOPSOWL_OPT_DATA::current_press

4.44.1.7 double SCOPSOWL_OPT_DATA::current_temp

4.44.1.8 int SCOPSOWL_OPT_DATA::diffusion_type

4.44.1.9 double SCOPSOWL_OPT_DATA::e_norm

4.44.1.10 double SCOPSOWL_OPT_DATA::e_norm_old

4.44.1.11 int SCOPSOWL_OPT_DATA::evaluation

4.44.1.12 double SCOPSOWL_OPT_DATA::f_bias

4.44.1.13 double SCOPSOWL_OPT_DATA::f_bias_old

4.44.1.14 double SCOPSOWL_OPT_DATA::max_bias

4.44.1.15 int SCOPSOWL_OPT_DATA::max_guess_iter = 20

4.44.1.16 double SCOPSOWL_OPT_DATA::min_bias

4.44.1.17 int SCOPSOWL_OPT_DATA::num_curves

4.44.1.18 int SCOPSOWL_OPT_DATA::num_params = 1

4.44.1.19 bool SCOPSOWL_OPT_DATA::Optimize

4.44.1.20 SCOPSOWL_DATA SCOPSOWL_OPT_DATA::owl_dat

4.44.1.21 double SCOPSOWL_OPT_DATA::param_guess

4.44.1.22 double SCOPSOWL_OPT_DATA::param_guess_old

4.44.1.23 FILE* SCOPSOWL_OPT_DATA::ParamFile

4.44.1.24 std::vector<double> SCOPSOWL_OPT_DATA::q_data

4.44.1.25 std::vector<double> SCOPSOWL_OPT_DATA::q_sim

4.44.1.26 double SCOPSOWL_OPT_DATA::rel_tol_norm = 0.01

4.44.1.27 bool SCOPSOWL_OPT_DATA::Rough

4.44.1.28 double SCOPSOWL_OPT_DATA::simulation_equil

4.44.1.29 std::vector<double> SCOPSOWL_OPT_DATA::t

4.44.1.30 unsigned long int SCOPSOWL_OPT_DATA::total_eval

4.44.1.31 std::vector<double> SCOPSOWL_OPT_DATA::y_base

The documentation for this struct was generated from the following file:

- [scopsowl_opt.h](#)

4.45 SCOPSOWL_PARAM_DATA Struct Reference

```
#include <scopsowl.h>
```

Public Attributes

- [Matrix](#)< double > [qAvg](#)

- [Matrix](#)< double > [qAvg_old](#)
- [Matrix](#)< double > [Qst](#)
- [Matrix](#)< double > [Qst_old](#)
- [Matrix](#)< double > [dq_dc](#)
- double [xlC](#)
- double [qIntegralAvg](#)
- double [qIntegralAvg_old](#)
- double [QstAvg](#)
- double [QstAvg_old](#)
- double [qo](#)
- double [Qsto](#)
- double [dq_dco](#)
- double [pore_diffusion](#)
- double [film_transfer](#)
- double [activation_energy](#)
- double [ref_diffusion](#)
- double [ref_temperature](#)
- double [affinity](#)
- double [ref_pressure](#)
- bool [Adsorbable](#)
- std::string [speciesName](#)

4.45.1 Member Data Documentation

- 4.45.1.1 double SCOPSOWL_PARAM_DATA::activation_energy
- 4.45.1.2 bool SCOPSOWL_PARAM_DATA::Adsorbable
- 4.45.1.3 double SCOPSOWL_PARAM_DATA::affinity
- 4.45.1.4 [Matrix](#)<double> SCOPSOWL_PARAM_DATA::dq_dc
- 4.45.1.5 double SCOPSOWL_PARAM_DATA::dq_dco
- 4.45.1.6 double SCOPSOWL_PARAM_DATA::film_transfer
- 4.45.1.7 double SCOPSOWL_PARAM_DATA::pore_diffusion
- 4.45.1.8 [Matrix](#)<double> SCOPSOWL_PARAM_DATA::qAvg
- 4.45.1.9 [Matrix](#)<double> SCOPSOWL_PARAM_DATA::qAvg_old
- 4.45.1.10 double SCOPSOWL_PARAM_DATA::qIntegralAvg
- 4.45.1.11 double SCOPSOWL_PARAM_DATA::qIntegralAvg_old
- 4.45.1.12 double SCOPSOWL_PARAM_DATA::qo
- 4.45.1.13 [Matrix](#)<double> SCOPSOWL_PARAM_DATA::Qst
- 4.45.1.14 [Matrix](#)<double> SCOPSOWL_PARAM_DATA::Qst_old
- 4.45.1.15 double SCOPSOWL_PARAM_DATA::QstAvg

- 4.45.1.16 double SCOPSOWL_PARAM_DATA::QstAvg_old
- 4.45.1.17 double SCOPSOWL_PARAM_DATA::Qsto
- 4.45.1.18 double SCOPSOWL_PARAM_DATA::ref_diffusion
- 4.45.1.19 double SCOPSOWL_PARAM_DATA::ref_pressure
- 4.45.1.20 double SCOPSOWL_PARAM_DATA::ref_temperature
- 4.45.1.21 std::string SCOPSOWL_PARAM_DATA::speciesName
- 4.45.1.22 double SCOPSOWL_PARAM_DATA::xIC

The documentation for this struct was generated from the following file:

- [scopsowl.h](#)

4.46 SHARK_DATA Struct Reference

```
#include <shark.h>
```

Public Attributes

- [MasterSpeciesList](#) MasterList
- std::vector< [Reaction](#) > ReactionList
- std::vector< [MassBalance](#) > MassBalanceList
- std::vector< [UnsteadyReaction](#) > UnsteadyList
- std::vector< double(*) (const [Matrix](#)< double > &x, [SHARK_DATA](#) *shark_dat, const void *data) > OtherList
- int numvar
- int num_ssr
- int num_mbe
- int num_usr
- int num_other = 0
- int act_fun = IDEAL
- int totalsteps = 0
- int timesteps = 0
- int pH_index = -1
- int pOH_index = -1
- double simulationtime = 0.0
- double dt = 0.1
- double dt_min = sqrt(DBL_EPSILON)
- double t_out = 0.0
- double t_count = 0.0
- double time = 0.0
- double time_old = 0.0
- double pH = 7.0
- double Norm = 0.0
- double dielectric_const = 78.325
- double temperature = 298.15

- bool [steadystate](#) = true
- bool [TimeAdaptivity](#) = false
- bool [const_pH](#) = false
- bool [SpeciationCurve](#) = false
- bool [Console_Output](#) = true
- bool [File_Output](#) = false
- bool [Contains_pH](#) = false
- bool [Contains_pOH](#) = false
- bool [Converged](#) = false
- [Matrix](#)< double > [X_old](#)
- [Matrix](#)< double > [X_new](#)
- [Matrix](#)< double > [Conc_old](#)
- [Matrix](#)< double > [Conc_new](#)
- [Matrix](#)< double > [activity_new](#)
- [Matrix](#)< double > [activity_old](#)
- [int](#)(* [EvalActivity](#))(const [Matrix](#)< double > &x, [Matrix](#)< double > &F, const void *data)
- [int](#)(* [Residual](#))(const [Matrix](#)< double > &x, [Matrix](#)< double > &F, const void *data)
- [int](#)(* [lin_precon](#))(const [Matrix](#)< double > &r, [Matrix](#)< double > &p, const void *data)
- [PJFNK_DATA](#) [Newton_data](#)
- const void * [activity_data](#)
- const void * [residual_data](#)
- const void * [precon_data](#)
- const void * [other_data](#)
- FILE * [OutputFile](#)
- [yaml_cpp_class](#) [yaml_object](#)

4.46.1 Member Data Documentation

4.46.1.1 `int SHARK_DATA::act_fun = IDEAL`

4.46.1.2 `const void* SHARK_DATA::activity_data`

4.46.1.3 `Matrix<double> SHARK_DATA::activity_new`

4.46.1.4 `Matrix<double> SHARK_DATA::activity_old`

4.46.1.5 `Matrix<double> SHARK_DATA::Conc_new`

4.46.1.6 `Matrix<double> SHARK_DATA::Conc_old`

4.46.1.7 `bool SHARK_DATA::Console_Output = true`

4.46.1.8 `bool SHARK_DATA::const_pH = false`

4.46.1.9 `bool SHARK_DATA::Contains_pH = false`

4.46.1.10 `bool SHARK_DATA::Contains_pOH = false`

4.46.1.11 `bool SHARK_DATA::Converged = false`

4.46.1.12 `double SHARK_DATA::dielectric_const = 78.325`

4.46.1.13 `double SHARK_DATA::dt = 0.1`

- 4.46.1.14 `double SHARK_DATA::dt_min = sqrt(DBL_EPSILON)`
- 4.46.1.15 `int(* SHARK_DATA::EvalActivity)(const Matrix< double > &x, Matrix< double > &F, const void *data)`
- 4.46.1.16 `bool SHARK_DATA::File_Output = false`
- 4.46.1.17 `int(* SHARK_DATA::lin_precon)(const Matrix< double > &r, Matrix< double > &p, const void *data)`
- 4.46.1.18 `std::vector<MassBalance> SHARK_DATA::MassBalanceList`
- 4.46.1.19 `MasterSpeciesList SHARK_DATA::MasterList`
- 4.46.1.20 `PJFNK_DATA SHARK_DATA::Newton_data`
- 4.46.1.21 `double SHARK_DATA::Norm = 0.0`
- 4.46.1.22 `int SHARK_DATA::num_mbe`
- 4.46.1.23 `int SHARK_DATA::num_other = 0`
- 4.46.1.24 `int SHARK_DATA::num_ssr`
- 4.46.1.25 `int SHARK_DATA::num_usr`
- 4.46.1.26 `int SHARK_DATA::numvar`
- 4.46.1.27 `const void* SHARK_DATA::other_data`
- 4.46.1.28 `std::vector< double (*) (const Matrix<double> &x, SHARK_DATA *shark_dat, const void *data) > SHARK_DATA::OtherList`
- 4.46.1.29 `FILE* SHARK_DATA::OutputFile`
- 4.46.1.30 `double SHARK_DATA::pH = 7.0`
- 4.46.1.31 `int SHARK_DATA::pH_index = -1`
- 4.46.1.32 `int SHARK_DATA::pOH_index = -1`
- 4.46.1.33 `const void* SHARK_DATA::precon_data`
- 4.46.1.34 `std::vector<Reaction> SHARK_DATA::ReactionList`
- 4.46.1.35 `int(* SHARK_DATA::Residual)(const Matrix< double > &x, Matrix< double > &F, const void *data)`
- 4.46.1.36 `const void* SHARK_DATA::residual_data`
- 4.46.1.37 `double SHARK_DATA::simulationtime = 0.0`
- 4.46.1.38 `bool SHARK_DATA::SpeciationCurve = false`
- 4.46.1.39 `bool SHARK_DATA::steadystate = true`
- 4.46.1.40 `double SHARK_DATA::t_count = 0.0`
- 4.46.1.41 `double SHARK_DATA::t_out = 0.0`

- 4.46.1.42 double SHARK_DATA::temperature = 298.15
- 4.46.1.43 double SHARK_DATA::time = 0.0
- 4.46.1.44 double SHARK_DATA::time_old = 0.0
- 4.46.1.45 bool SHARK_DATA::TimeAdaptivity = false
- 4.46.1.46 int SHARK_DATA::timesteps = 0
- 4.46.1.47 int SHARK_DATA::totalsteps = 0
- 4.46.1.48 std::vector<UnsteadyReaction> SHARK_DATA::UnsteadyList
- 4.46.1.49 Matrix<double> SHARK_DATA::X_new
- 4.46.1.50 Matrix<double> SHARK_DATA::X_old
- 4.46.1.51 yaml_cpp_class SHARK_DATA::yaml_object

The documentation for this struct was generated from the following file:

- [shark.h](#)

4.47 SKUA_DATA Struct Reference

```
#include <skua.h>
```

Public Attributes

- unsigned long int [total_steps](#)
- int [coord](#)
- double [sim_time](#)
- double [t_old](#)
- double [t](#)
- double [t_counter](#) = 0.0
- double [t_print](#)
- double [qTn](#)
- double [qTnp1](#)
- bool [Print2File](#) = true
- bool [Print2Console](#) = true
- double [gas_velocity](#)
- double [pellet_radius](#)
- double [char_measure](#)
- bool [DirichletBC](#) = true
- bool [NonLinear](#) = true
- std::vector< double > [y](#)
- FILE * [OutputFile](#)
- double(* [eval_diff](#))(int i, int l, const void *[user_data](#))
- double(* [eval_kf](#))(int i, const void *[user_data](#))
- const void * [user_data](#)
- [MAGPIE_DATA](#) [magpie_dat](#)
- [MIXED_GAS](#) * [gas_dat](#)
- std::vector< [FINCH_DATA](#) > [finch_dat](#)
- std::vector< [SKUA_PARAM](#) > [param_dat](#)

4.47.1 Member Data Documentation

- 4.47.1.1 `double SKUA_DATA::char_measure`
- 4.47.1.2 `int SKUA_DATA::coord`
- 4.47.1.3 `bool SKUA_DATA::DirichletBC = true`
- 4.47.1.4 `double(* SKUA_DATA::eval_diff)(int i, int l, const void *user_data)`
- 4.47.1.5 `double(* SKUA_DATA::eval_kf)(int i, const void *user_data)`
- 4.47.1.6 `std::vector<FINCH_DATA> SKUA_DATA::finch_dat`
- 4.47.1.7 `MIXED_GAS* SKUA_DATA::gas_dat`
- 4.47.1.8 `double SKUA_DATA::gas_velocity`
- 4.47.1.9 `MAGPIE_DATA SKUA_DATA::magpie_dat`
- 4.47.1.10 `bool SKUA_DATA::NonLinear = true`
- 4.47.1.11 `FILE* SKUA_DATA::OutputFile`
- 4.47.1.12 `std::vector<SKUA_PARAM> SKUA_DATA::param_dat`
- 4.47.1.13 `double SKUA_DATA::pellet_radius`
- 4.47.1.14 `bool SKUA_DATA::Print2Console = true`
- 4.47.1.15 `bool SKUA_DATA::Print2File = true`
- 4.47.1.16 `double SKUA_DATA::qTn`
- 4.47.1.17 `double SKUA_DATA::qTnp1`
- 4.47.1.18 `double SKUA_DATA::sim_time`
- 4.47.1.19 `double SKUA_DATA::t`
- 4.47.1.20 `double SKUA_DATA::t_counter = 0.0`
- 4.47.1.21 `double SKUA_DATA::t_old`
- 4.47.1.22 `double SKUA_DATA::t_print`
- 4.47.1.23 `unsigned long int SKUA_DATA::total_steps`
- 4.47.1.24 `const void* SKUA_DATA::user_data`
- 4.47.1.25 `std::vector<double> SKUA_DATA::y`

The documentation for this struct was generated from the following file:

- [skua.h](#)

4.48 SKUA_OPT_DATA Struct Reference

```
#include <skua_opt.h>
```

Public Attributes

- int [num_curves](#)
- int [evaluation](#)
- unsigned long int [total_eval](#)
- int [current_points](#)
- int [num_params](#) = 1
- int [diffusion_type](#)
- int [adsorb_index](#)
- int [max_guess_iter](#) = 20
- bool [Optimize](#)
- bool [Rough](#)
- double [current_temp](#)
- double [current_press](#)
- double [current_equil](#)
- double [simulation_equil](#)
- double [max_bias](#)
- double [min_bias](#)
- double [e_norm](#)
- double [f_bias](#)
- double [e_norm_old](#)
- double [f_bias_old](#)
- double [param_guess](#)
- double [param_guess_old](#)
- double [rel_tol_norm](#) = 0.1
- double [abs_tol_bias](#) = 0.1
- std::vector< double > [y_base](#)
- std::vector< double > [q_data](#)
- std::vector< double > [q_sim](#)
- std::vector< double > [t](#)
- FILE * [ParamFile](#)
- FILE * [CompareFile](#)
- [SKUA_DATA](#) [skua_dat](#)

4.48.1 Member Data Documentation

4.48.1.1 double SKUA_OPT_DATA::abs_tol_bias = 0.1

4.48.1.2 int SKUA_OPT_DATA::adsorb_index

4.48.1.3 FILE* SKUA_OPT_DATA::CompareFile

4.48.1.4 double SKUA_OPT_DATA::current_equil

4.48.1.5 int SKUA_OPT_DATA::current_points

4.48.1.6 double SKUA_OPT_DATA::current_press

4.48.1.7 double SKUA_OPT_DATA::current_temp

```

4.48.1.8  int SKUA_OPT_DATA::diffusion_type
4.48.1.9  double SKUA_OPT_DATA::e_norm
4.48.1.10 double SKUA_OPT_DATA::e_norm_old
4.48.1.11 int SKUA_OPT_DATA::evaluation
4.48.1.12 double SKUA_OPT_DATA::f_bias
4.48.1.13 double SKUA_OPT_DATA::f_bias_old
4.48.1.14 double SKUA_OPT_DATA::max_bias
4.48.1.15 int SKUA_OPT_DATA::max_guess_iter = 20
4.48.1.16 double SKUA_OPT_DATA::min_bias
4.48.1.17 int SKUA_OPT_DATA::num_curves
4.48.1.18 int SKUA_OPT_DATA::num_params = 1
4.48.1.19 bool SKUA_OPT_DATA::Optimize
4.48.1.20 double SKUA_OPT_DATA::param_guess
4.48.1.21 double SKUA_OPT_DATA::param_guess_old
4.48.1.22 FILE* SKUA_OPT_DATA::ParamFile
4.48.1.23 std::vector<double> SKUA_OPT_DATA::q_data
4.48.1.24 std::vector<double> SKUA_OPT_DATA::q_sim
4.48.1.25 double SKUA_OPT_DATA::rel_tol_norm = 0.1
4.48.1.26 bool SKUA_OPT_DATA::Rough
4.48.1.27 double SKUA_OPT_DATA::simulation_equil
4.48.1.28 SKUA_DATA SKUA_OPT_DATA::skua_dat
4.48.1.29 std::vector<double> SKUA_OPT_DATA::t
4.48.1.30 unsigned long int SKUA_OPT_DATA::total_eval
4.48.1.31 std::vector<double> SKUA_OPT_DATA::y_base

```

The documentation for this struct was generated from the following file:

- [skua_opt.h](#)

4.49 SKUA_PARAM Struct Reference

```
#include <skua.h>
```


Public Attributes

- double [activation_energy](#)
- double [ref_diffusion](#)
- double [ref_temperature](#)
- double [affinity](#)
- double [ref_pressure](#)
- double [film_transfer](#)
- double [xIC](#)
- double [y_eff](#)
- double [Qstn](#)
- double [Qstnp1](#)
- double [xn](#)
- double [xnp1](#)
- bool [Adsorbable](#)
- std::string [speciesName](#)

4.49.1 Member Data Documentation

4.49.1.1 double SKUA_PARAM::activation_energy

4.49.1.2 bool SKUA_PARAM::Adsorbable

4.49.1.3 double SKUA_PARAM::affinity

4.49.1.4 double SKUA_PARAM::film_transfer

4.49.1.5 double SKUA_PARAM::Qstn

4.49.1.6 double SKUA_PARAM::Qstnp1

4.49.1.7 double SKUA_PARAM::ref_diffusion

4.49.1.8 double SKUA_PARAM::ref_pressure

4.49.1.9 double SKUA_PARAM::ref_temperature

4.49.1.10 std::string SKUA_PARAM::speciesName

4.49.1.11 double SKUA_PARAM::xIC

4.49.1.12 double SKUA_PARAM::xn

4.49.1.13 double SKUA_PARAM::xnp1

4.49.1.14 double SKUA_PARAM::y_eff

The documentation for this struct was generated from the following file:

- [skua.h](#)

4.50 Speciation_Test01_Data Struct Reference

```
#include <sandbox.h>
```

Public Attributes

- int `N` = 4
- const double `logKw` = -14.0
- const double `logKa1` = -6.35
- const double `logKa2` = -10.33
- double `CT` = 0.1786
- double `NaT` = 0.1786
- std::vector< `Molecule` > `x`
- `Matrix`< double > `Jacobian`
- `Matrix`< double > `NumJac`
- `Matrix`< double > `logC`
- `Matrix`< double > `C`

4.50.1 Member Data Documentation

4.50.1.1 `Matrix`<double> `Speciation_Test01_Data::C`

4.50.1.2 double `Speciation_Test01_Data::CT` = 0.1786

4.50.1.3 `Matrix`<double> `Speciation_Test01_Data::Jacobian`

4.50.1.4 `Matrix`<double> `Speciation_Test01_Data::logC`

4.50.1.5 const double `Speciation_Test01_Data::logKa1` = -6.35

4.50.1.6 const double `Speciation_Test01_Data::logKa2` = -10.33

4.50.1.7 const double `Speciation_Test01_Data::logKw` = -14.0

4.50.1.8 int `Speciation_Test01_Data::N` = 4

4.50.1.9 double `Speciation_Test01_Data::NaT` = 0.1786

4.50.1.10 `Matrix`<double> `Speciation_Test01_Data::NumJac`

4.50.1.11 std::vector<`Molecule`> `Speciation_Test01_Data::x`

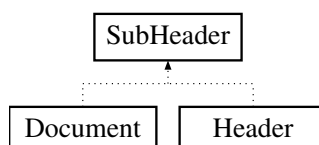
The documentation for this struct was generated from the following file:

- [sandbox.h](#)

4.51 SubHeader Class Reference

```
#include <yaml_wrapper.h>
```

Inheritance diagram for SubHeader:



Public Member Functions

- [SubHeader](#) ()
- [~SubHeader](#) ()
- [SubHeader](#) (const [SubHeader](#) &subheader)
- [SubHeader](#) (const [KeyValueMap](#) &map)
- [SubHeader](#) (std::string [name](#))
- [SubHeader](#) (std::string [name](#), const [KeyValueMap](#) &map)
- [SubHeader](#) & [operator=](#) (const [SubHeader](#) &sub)
- [ValueTypePair](#) & [operator\[\]](#) (const std::string key)
- [ValueTypePair](#) [operator\[\]](#) (const std::string key) const
- [KeyValueMap](#) & [getMap](#) ()
- void [clear](#) ()
- void [addPair](#) (std::string key, std::string val)
- void [addPair](#) (std::string key, std::string val, int type)
- void [setName](#) (std::string [name](#))
- void [setAlias](#) (std::string [alias](#))
- void [setAlias](#) (std::string [alias](#), int [state](#))
- void [setNameAliasPair](#) (std::string [name](#), std::string [alias](#), int [state](#))
- void [setState](#) (int [state](#))
- void [DisplayContents](#) ()
- std::string [getName](#) ()
- std::string [getAlias](#) ()
- bool [isAlias](#) ()
- bool [isAnchor](#) ()
- int [getState](#) ()

Protected Attributes

- [KeyValueMap](#) [Data_Map](#)
- std::string [name](#)
- std::string [alias](#)
- int [state](#)

4.51.1 Constructor & Destructor Documentation

4.51.1.1 [SubHeader::SubHeader](#) ()

4.51.1.2 [SubHeader::~SubHeader](#) ()

4.51.1.3 [SubHeader::SubHeader](#) (const [SubHeader](#) & *subheader*)

4.51.1.4 [SubHeader::SubHeader](#) (const [KeyValueMap](#) & *map*)

4.51.1.5 [SubHeader::SubHeader](#) (std::string *name*)

4.51.1.6 [SubHeader::SubHeader](#) (std::string *name*, const [KeyValueMap](#) & *map*)

4.51.2 Member Function Documentation

4.51.2.1 void [SubHeader::addPair](#) (std::string *key*, std::string *val*)

4.51.2.2 void [SubHeader::addPair](#) (std::string *key*, std::string *val*, int *type*)

- 4.51.2.3 `void SubHeader::clear ()`
- 4.51.2.4 `void SubHeader::DisplayContents ()`
- 4.51.2.5 `std::string SubHeader::getAlias ()`
- 4.51.2.6 `KeyValueMap & SubHeader::getMap ()`
- 4.51.2.7 `std::string SubHeader::getName ()`
- 4.51.2.8 `int SubHeader::getState ()`
- 4.51.2.9 `bool SubHeader::isAlias ()`
- 4.51.2.10 `bool SubHeader::isAnchor ()`
- 4.51.2.11 `SubHeader & SubHeader::operator= (const SubHeader & sub)`
- 4.51.2.12 `ValueTypePair & SubHeader::operator[] (const std::string key)`
- 4.51.2.13 `ValueTypePair SubHeader::operator[] (const std::string key) const`
- 4.51.2.14 `void SubHeader::setAlias (std::string alias)`
- 4.51.2.15 `void SubHeader::setAlias (std::string alias, int state)`
- 4.51.2.16 `void SubHeader::setName (std::string name)`
- 4.51.2.17 `void SubHeader::setNameAliasPair (std::string name, std::string alias, int state)`
- 4.51.2.18 `void SubHeader::setState (int state)`

4.51.3 Member Data Documentation

- 4.51.3.1 `std::string SubHeader::alias` [protected]
- 4.51.3.2 `KeyValueMap SubHeader::Data_Map` [protected]
- 4.51.3.3 `std::string SubHeader::name` [protected]
- 4.51.3.4 `int SubHeader::state` [protected]

The documentation for this class was generated from the following files:

- [yaml_wrapper.h](#)
- [yaml_wrapper.cpp](#)

4.52 SYSTEM_DATA Struct Reference

```
#include <magpie.h>
```

Public Attributes

- `double T`

- double [PT](#)
- double [qT](#)
- double [PI](#)
- double [pi](#)
- double [As](#)
- int [N](#)
- int [I](#)
- int [J](#)
- int [K](#)
- unsigned long int [total_eval](#)
- double [avg_norm](#)
- double [max_norm](#)
- int [Sys](#)
- int [Par](#)
- bool [Recover](#)
- bool [Carrier](#)
- bool [Ideal](#)
- bool [Output](#)

4.52.1 Member Data Documentation

4.52.1.1 double SYSTEM_DATA::As

4.52.1.2 double SYSTEM_DATA::avg_norm

4.52.1.3 bool SYSTEM_DATA::Carrier

4.52.1.4 int SYSTEM_DATA::I

4.52.1.5 bool SYSTEM_DATA::Ideal

4.52.1.6 int SYSTEM_DATA::J

4.52.1.7 int SYSTEM_DATA::K

4.52.1.8 double SYSTEM_DATA::max_norm

4.52.1.9 int SYSTEM_DATA::N

4.52.1.10 bool SYSTEM_DATA::Output

4.52.1.11 int SYSTEM_DATA::Par

4.52.1.12 double SYSTEM_DATA::PI

4.52.1.13 double SYSTEM_DATA::pi

4.52.1.14 double SYSTEM_DATA::PT

4.52.1.15 double SYSTEM_DATA::qT

4.52.1.16 bool SYSTEM_DATA::Recover

4.52.1.17 int SYSTEM_DATA::Sys

4.52.1.18 double SYSTEM_DATA::T

4.52.1.19 unsigned long int SYSTEM_DATA::total_eval

The documentation for this struct was generated from the following file:

- [magpie.h](#)

4.53 TRAJECTORY_DATA Struct Reference

```
#include <Trajectory.h>
```

Public Attributes

- double [mu_0](#) = 12.57e-7
- double [rho_f](#) = 1000.0
- double [eta](#) = 0.001
- double [Hamaker](#) = 1.3e-21
- double [Temp](#) = 298
- double [k](#) = 1.38e-23
- double [Rs](#) = 0.0026925
- double [L](#) = 0.0611
- double [porosity](#) = 0.8979
- double [V_separator](#)
- double [a](#) = 33.0e-6
- double [V_wire](#)
- double [L_wire](#)
- double [A_separator](#)
- double [A_wire](#)
- double [B0](#) = 1.0
- double [H0](#)
- double [Ms](#) = 0.6
- double [b](#) = 0.25e-6
- double [chi_p](#) = 3.87e-6
- double [rho_p](#) = 8700.0
- double [Q_in](#)
- double [V0](#)
- double [Y_initial](#) = 20.0
- double [dt](#)
- double [M](#)
- double [mp](#)
- double [beta](#)
- double [q_bar](#)
- double [sigma_v](#)
- double [sigma_vz](#)
- double [sigma_z](#)
- double [sigma_n](#)
- double [sigma_m](#)
- double [n_rand](#)
- double [m_rand](#)
- double [s_rand](#)
- double [t_rand](#)
- [Matrix](#)< double > [POL](#)

- [Matrix](#)< double > [H](#)
- [Matrix](#)< double > [dX](#)
- [Matrix](#)< double > [dY](#)
- [Matrix](#)< double > [X](#)
- [Matrix](#)< double > [Y](#)
- [Matrix](#)< int > [Cap](#)

4.53.1 Member Data Documentation

4.53.1.1 double TRAJECTORY_DATA::a = 33.0e-6

4.53.1.2 double TRAJECTORY_DATA::A_separator

4.53.1.3 double TRAJECTORY_DATA::A_wire

4.53.1.4 double TRAJECTORY_DATA::b = 0.25e-6

4.53.1.5 double TRAJECTORY_DATA::B0 = 1.0

4.53.1.6 double TRAJECTORY_DATA::beta

4.53.1.7 [Matrix](#)<int> TRAJECTORY_DATA::Cap

4.53.1.8 double TRAJECTORY_DATA::chi_p = 3.87e-6

4.53.1.9 double TRAJECTORY_DATA::dt

4.53.1.10 [Matrix](#)<double> TRAJECTORY_DATA::dX

4.53.1.11 [Matrix](#)<double> TRAJECTORY_DATA::dY

4.53.1.12 double TRAJECTORY_DATA::eta = 0.001

4.53.1.13 [Matrix](#)<double> TRAJECTORY_DATA::H

4.53.1.14 double TRAJECTORY_DATA::H0

4.53.1.15 double TRAJECTORY_DATA::Hamaker = 1.3e-21

4.53.1.16 double TRAJECTORY_DATA::k = 1.38e-23

4.53.1.17 double TRAJECTORY_DATA::L = 0.0611

4.53.1.18 double TRAJECTORY_DATA::L_wire

4.53.1.19 double TRAJECTORY_DATA::M

4.53.1.20 double TRAJECTORY_DATA::m_rand

4.53.1.21 double TRAJECTORY_DATA::mp

4.53.1.22 double TRAJECTORY_DATA::Ms = 0.6

4.53.1.23 double TRAJECTORY_DATA::mu_0 = 12.57e-7

- 4.53.1.24 `double TRAJECTORY_DATA::n_rand`
- 4.53.1.25 `Matrix<double> TRAJECTORY_DATA::POL`
- 4.53.1.26 `double TRAJECTORY_DATA::porosity = 0.8979`
- 4.53.1.27 `double TRAJECTORY_DATA::q_bar`
- 4.53.1.28 `double TRAJECTORY_DATA::Q_in`
- 4.53.1.29 `double TRAJECTORY_DATA::rho_f = 1000.0`
- 4.53.1.30 `double TRAJECTORY_DATA::rho_p = 8700.0`
- 4.53.1.31 `double TRAJECTORY_DATA::Rs = 0.0026925`
- 4.53.1.32 `double TRAJECTORY_DATA::s_rand`
- 4.53.1.33 `double TRAJECTORY_DATA::sigma_m`
- 4.53.1.34 `double TRAJECTORY_DATA::sigma_n`
- 4.53.1.35 `double TRAJECTORY_DATA::sigma_v`
- 4.53.1.36 `double TRAJECTORY_DATA::sigma_vz`
- 4.53.1.37 `double TRAJECTORY_DATA::sigma_z`
- 4.53.1.38 `double TRAJECTORY_DATA::t_rand`
- 4.53.1.39 `double TRAJECTORY_DATA::Temp = 298`
- 4.53.1.40 `double TRAJECTORY_DATA::V0`
- 4.53.1.41 `double TRAJECTORY_DATA::V_separator`
- 4.53.1.42 `double TRAJECTORY_DATA::V_wire`
- 4.53.1.43 `Matrix<double> TRAJECTORY_DATA::X`
- 4.53.1.44 `Matrix<double> TRAJECTORY_DATA::Y`
- 4.53.1.45 `double TRAJECTORY_DATA::Y_initial = 20.0`

The documentation for this struct was generated from the following file:

- [Trajectory.h](#)

4.54 UI_DATA Struct Reference

```
#include <ui.h>
```

Public Attributes

- [ValueTypePair value_type](#)

- `std::vector< std::string > user_input`
- `std::vector< std::string > input_files`
- `std::string path`
- `int count = 0`
- `int max = 3`
- `int option`
- `bool Path = false`
- `bool Files = false`
- `bool MissingArg = true`
- `bool BasicUI = true`
- `int argc`
- `const char * argv []`

4.54.1 Member Data Documentation

4.54.1.1 `int UI_DATA::argc`

4.54.1.2 `const char* UI_DATA::argv[]`

4.54.1.3 `bool UI_DATA::BasicUI = true`

4.54.1.4 `int UI_DATA::count = 0`

4.54.1.5 `bool UI_DATA::Files = false`

4.54.1.6 `std::vector<std::string> UI_DATA::input_files`

4.54.1.7 `int UI_DATA::max = 3`

4.54.1.8 `bool UI_DATA::MissingArg = true`

4.54.1.9 `int UI_DATA::option`

4.54.1.10 `std::string UI_DATA::path`

4.54.1.11 `bool UI_DATA::Path = false`

4.54.1.12 `std::vector<std::string> UI_DATA::user_input`

4.54.1.13 `ValueTypePair UI_DATA::value_type`

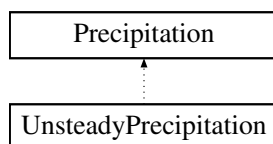
The documentation for this struct was generated from the following file:

- [ui.h](#)

4.55 UnsteadyPrecipitation Class Reference

```
#include <shark.h>
```

Inheritance diagram for UnsteadyPrecipitation:



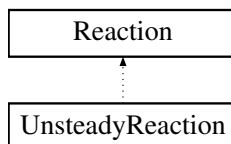
The documentation for this class was generated from the following file:

- [shark.h](#)

4.56 UnsteadyReaction Class Reference

```
#include <shark.h>
```

Inheritance diagram for UnsteadyReaction:



Public Member Functions

- [UnsteadyReaction](#) ()
- [~UnsteadyReaction](#) ()
- void [Initialize_List](#) (MasterSpeciesList &List)
- void [Display_Info](#) ()
- void [Set_Species_Index](#) (int i)
- void [Set_Species_Index](#) (std::string formula)
- void [Set_Stoichiometric](#) (int i, double v)
- void [Set_Equilibrium](#) (double v)
- void [Set_Enthalpy](#) (double H)
- void [Set_Entropy](#) (double S)
- void [Set_EnthalpyANDEntropy](#) (double H, double S)
- void [Set_Energy](#) (double G)
- void [Set_InitialValue](#) (double ic)
- void [Set_MaximumValue](#) (double max)
- void [Set_Forward](#) (double forward)
- void [Set_Reverse](#) (double reverse)
- void [Set_ForwardRef](#) (double Fref)
- void [Set_ReverseRef](#) (double Rref)
- void [Set_ActivationEnergy](#) (double E)
- void [Set_Affinity](#) (double b)
- void [Set_TimeStep](#) (double dt)
- void [checkSpeciesEnergies](#) ()
- void [calculateEnergies](#) ()
- void [calculateEquilibrium](#) (double T)
- void [calculateRate](#) (double T)
- bool [haveEquilibrium](#) ()
- bool [haveRate](#) ()
- int [Get_Species_Index](#) ()

- double [Get_Stoichiometric](#) (int i)
- double [Get_Equilibrium](#) ()
- double [Get_Enthalpy](#) ()
- double [Get_Entropy](#) ()
- double [Get_Energy](#) ()
- double [Get_InitialValue](#) ()
- double [Get_MaximumValue](#) ()
- double [Get_Forward](#) ()
- double [Get_Reverse](#) ()
- double [Get_ForwardRef](#) ()
- double [Get_ReverseRef](#) ()
- double [Get_ActivationEnergy](#) ()
- double [Get_Affinity](#) ()
- double [Get_TimeStep](#) ()
- double [Eval_ReactionRate](#) (const [Matrix](#)< double > &x, const [Matrix](#)< double > &gama)
- double [Eval_Residual](#) (const [Matrix](#)< double > &x_new, const [Matrix](#)< double > &x_old, const [Matrix](#)< double > &gama_new, const [Matrix](#)< double > &gama_old)
- double [Eval_Residual](#) (const [Matrix](#)< double > &x, const [Matrix](#)< double > &gama)
- double [Eval_IC_Residual](#) (const [Matrix](#)< double > &x)
- double [Explicit_Eval](#) (const [Matrix](#)< double > &x, const [Matrix](#)< double > &gama)

Protected Attributes

- double [initial_value](#)
- double [max_value](#)
- double [forward_rate](#)
- double [reverse_rate](#)
- double [forward_ref_rate](#)
- double [reverse_ref_rate](#)
- double [activation_energy](#)
- double [temperature_affinity](#)
- double [time_step](#)
- bool [HaveForward](#)
- bool [HaveReverse](#)
- bool [HaveForRef](#)
- bool [HaveRevRef](#)
- int [species_index](#)

Additional Inherited Members

4.56.1 Constructor & Destructor Documentation

4.56.1.1 [UnsteadyReaction::UnsteadyReaction](#) ()

4.56.1.2 [UnsteadyReaction::~~UnsteadyReaction](#) ()

4.56.2 Member Function Documentation

4.56.2.1 [void UnsteadyReaction::calculateEnergies](#) ()

4.56.2.2 [void UnsteadyReaction::calculateEquilibrium](#) (double T)

4.56.2.3 [void UnsteadyReaction::calculateRate](#) (double T)

- 4.56.2.4 void UnsteadyReaction::checkSpeciesEnergies ()
- 4.56.2.5 void UnsteadyReaction::Display_Info ()
- 4.56.2.6 double UnsteadyReaction::Eval_IC_Residual (const Matrix< double > & x)
- 4.56.2.7 double UnsteadyReaction::Eval_ReactionRate (const Matrix< double > & x, const Matrix< double > & gama)
- 4.56.2.8 double UnsteadyReaction::Eval_Residual (const Matrix< double > & x_new, const Matrix< double > & x_old, const Matrix< double > & gama_new, const Matrix< double > & gama_old)
- 4.56.2.9 double UnsteadyReaction::Eval_Residual (const Matrix< double > & x, const Matrix< double > & gama)
- 4.56.2.10 double UnsteadyReaction::Explicit_Eval (const Matrix< double > & x, const Matrix< double > & gama)
- 4.56.2.11 double UnsteadyReaction::Get_ActivationEnergy ()
- 4.56.2.12 double UnsteadyReaction::Get_Affinity ()
- 4.56.2.13 double UnsteadyReaction::Get_Energy ()
- 4.56.2.14 double UnsteadyReaction::Get_Enthalpy ()
- 4.56.2.15 double UnsteadyReaction::Get_Entropy ()
- 4.56.2.16 double UnsteadyReaction::Get_Equilibrium ()
- 4.56.2.17 double UnsteadyReaction::Get_Forward ()
- 4.56.2.18 double UnsteadyReaction::Get_ForwardRef ()
- 4.56.2.19 double UnsteadyReaction::Get_InitialValue ()
- 4.56.2.20 double UnsteadyReaction::Get_MaximumValue ()
- 4.56.2.21 double UnsteadyReaction::Get_Reverse ()
- 4.56.2.22 double UnsteadyReaction::Get_ReverseRef ()
- 4.56.2.23 int UnsteadyReaction::Get_Species_Index ()
- 4.56.2.24 double UnsteadyReaction::Get_Stoichiometric (int i)
- 4.56.2.25 double UnsteadyReaction::Get_TimeStep ()
- 4.56.2.26 bool UnsteadyReaction::haveEquilibrium ()
- 4.56.2.27 bool UnsteadyReaction::haveRate ()
- 4.56.2.28 void UnsteadyReaction::Initialize_List (MasterSpeciesList & List)
- 4.56.2.29 void UnsteadyReaction::Set_ActivationEnergy (double E)
- 4.56.2.30 void UnsteadyReaction::Set_Affinity (double b)
- 4.56.2.31 void UnsteadyReaction::Set_Energy (double G)

- 4.56.2.32 void UnsteadyReaction::Set_Enthalpy (double *H*)
- 4.56.2.33 void UnsteadyReaction::Set_EnthalpyANDEntropy (double *H*, double *S*)
- 4.56.2.34 void UnsteadyReaction::Set_Entropy (double *S*)
- 4.56.2.35 void UnsteadyReaction::Set_Equilibrium (double *v*)
- 4.56.2.36 void UnsteadyReaction::Set_Forward (double *forward*)
- 4.56.2.37 void UnsteadyReaction::Set_ForwardRef (double *Fref*)
- 4.56.2.38 void UnsteadyReaction::Set_InitialValue (double *ic*)
- 4.56.2.39 void UnsteadyReaction::Set_MaximumValue (double *max*)
- 4.56.2.40 void UnsteadyReaction::Set_Reverse (double *reverse*)
- 4.56.2.41 void UnsteadyReaction::Set_ReverseRef (double *Rref*)
- 4.56.2.42 void UnsteadyReaction::Set_Species_Index (int *i*)
- 4.56.2.43 void UnsteadyReaction::Set_Species_Index (std::string *formula*)
- 4.56.2.44 void UnsteadyReaction::Set_Stoichiometric (int *i*, double *v*)
- 4.56.2.45 void UnsteadyReaction::Set_TimeStep (double *dt*)

4.56.3 Member Data Documentation

- 4.56.3.1 double UnsteadyReaction::activation_energy [protected]
- 4.56.3.2 double UnsteadyReaction::forward_rate [protected]
- 4.56.3.3 double UnsteadyReaction::forward_ref_rate [protected]
- 4.56.3.4 bool UnsteadyReaction::HaveForRef [protected]
- 4.56.3.5 bool UnsteadyReaction::HaveForward [protected]
- 4.56.3.6 bool UnsteadyReaction::HaveReverse [protected]
- 4.56.3.7 bool UnsteadyReaction::HaveRevRef [protected]
- 4.56.3.8 double UnsteadyReaction::initial_value [protected]
- 4.56.3.9 double UnsteadyReaction::max_value [protected]
- 4.56.3.10 double UnsteadyReaction::reverse_rate [protected]
- 4.56.3.11 double UnsteadyReaction::reverse_ref_rate [protected]
- 4.56.3.12 int UnsteadyReaction::species_index [protected]
- 4.56.3.13 double UnsteadyReaction::temperature_affinity [protected]

4.56.3.14 double UnsteadyReaction::time_step [protected]

The documentation for this class was generated from the following files:

- [shark.h](#)
- [shark.cpp](#)

4.57 ValueTypePair Class Reference

```
#include <yaml_wrapper.h>
```

Public Member Functions

- [ValueTypePair](#) ()
- [~ValueTypePair](#) ()
- [ValueTypePair](#) (const std::pair< std::string, int > &vt)
- [ValueTypePair](#) (std::string value, int [type](#))
- [ValueTypePair](#) (const [ValueTypePair](#) &vt)
- [ValueTypePair](#) & [operator=](#) (const [ValueTypePair](#) &vt)
- void [editValue](#) (std::string value)
- void [editPair](#) (std::string value, int [type](#))
- void [findType](#) ()
- void [assertType](#) (int [type](#))
- void [DisplayPair](#) ()
- std::string [getString](#) ()
- bool [getBool](#) ()
- double [getDouble](#) ()
- int [getInt](#) ()
- std::string [getValue](#) ()
- int [getType](#) ()
- std::pair< std::string, int > & [getPair](#) ()

Private Attributes

- std::pair< std::string, int > [Value_Type](#)
- int [type](#)

4.57.1 Constructor & Destructor Documentation

4.57.1.1 [ValueTypePair::ValueTypePair](#) ()

4.57.1.2 [ValueTypePair::~~ValueTypePair](#) ()

4.57.1.3 [ValueTypePair::ValueTypePair](#) (const std::pair< std::string, int > & vt)

4.57.1.4 [ValueTypePair::ValueTypePair](#) (std::string value, int type)

4.57.1.5 [ValueTypePair::ValueTypePair](#) (const [ValueTypePair](#) & vt)

4.57.2 Member Function Documentation

- 4.57.2.1 void ValueTypePair::assertType (int *type*)
- 4.57.2.2 void ValueTypePair::DisplayPair ()
- 4.57.2.3 void ValueTypePair::editPair (std::string *value*, int *type*)
- 4.57.2.4 void ValueTypePair::editValue (std::string *value*)
- 4.57.2.5 void ValueTypePair::findType ()
- 4.57.2.6 bool ValueTypePair::getBool ()
- 4.57.2.7 double ValueTypePair::getDouble ()
- 4.57.2.8 int ValueTypePair::getInt ()
- 4.57.2.9 std::pair< std::string, int > & ValueTypePair::getPair ()
- 4.57.2.10 std::string ValueTypePair::getString ()
- 4.57.2.11 int ValueTypePair::getType ()
- 4.57.2.12 std::string ValueTypePair::getValue ()
- 4.57.2.13 ValueTypePair & ValueTypePair::operator= (const ValueTypePair & *vt*)

4.57.3 Member Data Documentation

- 4.57.3.1 int ValueTypePair::type [private]
- 4.57.3.2 std::pair<std::string,int> ValueTypePair::Value_Type [private]

The documentation for this class was generated from the following files:

- [yaml_wrapper.h](#)
- [yaml_wrapper.cpp](#)

4.58 yaml_cpp_class Class Reference

```
#include <yaml_wrapper.h>
```

Public Member Functions

- [yaml_cpp_class](#) ()
- [~yaml_cpp_class](#) ()
- int [setInputFile](#) (const char *file)
- int [readInputFile](#) ()
- int [cleanup](#) ()
- int [executeYamlRead](#) (const char *file)
- [YamlWrapper](#) & [getYamlWrapper](#) ()
- void [DisplayContents](#) ()

Private Attributes

- [YamlWrapper](#) `yaml_wrapper`
- FILE * `input_file`
- const char * `file_name`
- `yaml_parser_t` `token_parser`
- `yaml_token_t` `current_token`
- `yaml_token_t` `previous_token`

4.58.1 Constructor & Destructor Documentation

4.58.1.1 `yaml_cpp_class::yaml_cpp_class ()`

4.58.1.2 `yaml_cpp_class::~~yaml_cpp_class ()`

4.58.2 Member Function Documentation

4.58.2.1 `int yaml_cpp_class::cleanup ()`

4.58.2.2 `void yaml_cpp_class::DisplayContents ()`

4.58.2.3 `int yaml_cpp_class::executeYamlRead (const char * file)`

4.58.2.4 `YamlWrapper & yaml_cpp_class::getYamlWrapper ()`

4.58.2.5 `int yaml_cpp_class::readInputFile ()`

4.58.2.6 `int yaml_cpp_class::setInputFile (const char * file)`

4.58.3 Member Data Documentation

4.58.3.1 `yaml_token_t yaml_cpp_class::current_token` [private]

4.58.3.2 `const char* yaml_cpp_class::file_name` [private]

4.58.3.3 `FILE* yaml_cpp_class::input_file` [private]

4.58.3.4 `yaml_token_t yaml_cpp_class::previous_token` [private]

4.58.3.5 `yaml_parser_t yaml_cpp_class::token_parser` [private]

4.58.3.6 `YamlWrapper yaml_cpp_class::yaml_wrapper` [private]

The documentation for this class was generated from the following files:

- [yaml_wrapper.h](#)
- [yaml_wrapper.cpp](#)

4.59 YamlWrapper Class Reference

```
#include <yaml_wrapper.h>
```


Public Member Functions

- [YamlWrapper](#) ()
- [~YamlWrapper](#) ()
- [YamlWrapper](#) (const [YamlWrapper](#) &yaml)
- [YamlWrapper](#) (std::string key, const [Document](#) &doc)
- [YamlWrapper](#) & [operator=](#) (const [YamlWrapper](#) &yaml)
- [Document](#) & [operator\(\)](#) (const std::string key)
- [Document](#) [operator\(\)](#) (const std::string key) const
- std::map< std::string, [Document](#) > & [getDocMap](#) ()
- [Document](#) & [getDocument](#) (std::string key)
- std::map< std::string, [Document](#) >::const_iterator [end](#) () const
- std::map< std::string, [Document](#) >::iterator [end](#) ()
- std::map< std::string, [Document](#) >::const_iterator [begin](#) () const
- std::map< std::string, [Document](#) >::iterator [begin](#) ()
- void [clear](#) ()
- void [resetKeys](#) ()
- void [changeKey](#) (std::string oldKey, std::string newKey)
- void [revalidateAllKeys](#) ()
- void [DisplayContents](#) ()
- void [addDocKey](#) (std::string key)
- void [copyAnchor2Alias](#) (std::string alias, [Document](#) &ref)
- int [size](#) ()
- [Document](#) & [getAnchoredDoc](#) (std::string alias)
- [Document](#) & [getDocFromHeadAlias](#) (std::string alias)
- [Document](#) & [getDocFromSubAlias](#) (std::string alias)

Private Attributes

- std::map< std::string, [Document](#) > [Doc_Map](#)

4.59.1 Constructor & Destructor Documentation

4.59.1.1 [YamlWrapper::YamlWrapper](#) ()

4.59.1.2 [YamlWrapper::~~YamlWrapper](#) ()

4.59.1.3 [YamlWrapper::YamlWrapper](#) (const [YamlWrapper](#) & *yaml*)

4.59.1.4 [YamlWrapper::YamlWrapper](#) (std::string *key*, const [Document](#) & *doc*)

4.59.2 Member Function Documentation

4.59.2.1 void [YamlWrapper::addDocKey](#) (std::string *key*)

4.59.2.2 std::map< std::string, [Document](#) >::const_iterator [YamlWrapper::begin](#) () const

4.59.2.3 std::map< std::string, [Document](#) >::iterator [YamlWrapper::begin](#) ()

4.59.2.4 void [YamlWrapper::changeKey](#) (std::string *oldKey*, std::string *newKey*)

- 4.59.2.5 `void YamlWrapper::clear ()`
- 4.59.2.6 `void YamlWrapper::copyAnchor2Alias (std::string alias, Document & ref)`
- 4.59.2.7 `void YamlWrapper::DisplayContents ()`
- 4.59.2.8 `std::map< std::string, Document >::const_iterator YamlWrapper::end () const`
- 4.59.2.9 `std::map< std::string, Document >::iterator YamlWrapper::end ()`
- 4.59.2.10 `Document & YamlWrapper::getAnchoredDoc (std::string alias)`
- 4.59.2.11 `Document & YamlWrapper::getDocFromHeadAlias (std::string alias)`
- 4.59.2.12 `Document & YamlWrapper::getDocFromSubAlias (std::string alias)`
- 4.59.2.13 `std::map< std::string, Document > & YamlWrapper::getDocMap ()`
- 4.59.2.14 `Document & YamlWrapper::getDocument (std::string key)`
- 4.59.2.15 `Document & YamlWrapper::operator() (const std::string key)`
- 4.59.2.16 `Document YamlWrapper::operator() (const std::string key) const`
- 4.59.2.17 `YamlWrapper & YamlWrapper::operator= (const YamlWrapper & yaml)`
- 4.59.2.18 `void YamlWrapper::resetKeys ()`
- 4.59.2.19 `void YamlWrapper::revalidateAllKeys ()`
- 4.59.2.20 `int YamlWrapper::size ()`

4.59.3 Member Data Documentation

- 4.59.3.1 `std::map<std::string, Document> YamlWrapper::Doc_Map [private]`

The documentation for this class was generated from the following files:

- [yaml_wrapper.h](#)
- [yaml_wrapper.cpp](#)

Chapter 5

File Documentation

5.1 dogfish.cpp File Reference

```
#include "dogfish.h"
```

Functions

- void [print2file_species_header](#) (FILE *Output, [DOGFISH_DATA](#) *dog_dat, int i)
- void [print2file_DOGFISH_header](#) ([DOGFISH_DATA](#) *dog_dat)
- void [print2file_DOGFISH_result_old](#) ([DOGFISH_DATA](#) *dog_dat)
- void [print2file_DOGFISH_result_new](#) ([DOGFISH_DATA](#) *dog_dat)
- double [default_Retardation](#) (int i, int l, const void *data)
- double [default_IntraDiffusion](#) (int i, int l, const void *data)
- double [default_FilmMTCoeff](#) (int i, const void *data)
- double [default_SurfaceConcentration](#) (int i, const void *data)
- int [setup_DOGFISH_DATA](#) (FILE *file, double(*eval_R)(int i, int l, const void *user_data), double(*eval_DI)(int i, int l, const void *user_data), double(*eval_kf)(int i, const void *user_data), double(*eval_qs)(int i, const void *user_data), const void *user_data, [DOGFISH_DATA](#) *dog_dat)
- int [DOGFISH_Executioner](#) ([DOGFISH_DATA](#) *dog_dat)
- int [set_DOGFISH_ICs](#) ([DOGFISH_DATA](#) *dog_dat)
- int [set_DOGFISH_timestep](#) ([DOGFISH_DATA](#) *dog_dat)
- int [DOGFISH_preprocesses](#) ([DOGFISH_DATA](#) *dog_dat)
- int [set_DOGFISH_params](#) (const void *user_data)
- int [DOGFISH_postprocesses](#) ([DOGFISH_DATA](#) *dog_dat)
- int [DOGFISH_reset](#) ([DOGFISH_DATA](#) *dog_dat)
- int [DOGFISH](#) ([DOGFISH_DATA](#) *dog_dat)
- int [DOGFISH_TESTS](#) ()

5.1.1 Function Documentation

5.1.1.1 double [default_FilmMTCoeff](#) (int *i*, const void * *data*)

5.1.1.2 double [default_IntraDiffusion](#) (int *i*, int *l*, const void * *data*)

5.1.1.3 double [default_Retardation](#) (int *i*, int *l*, const void * *data*)

5.1.1.4 double [default_SurfaceConcentration](#) (int *i*, const void * *data*)

- 5.1.1.5 int DOGFISH (DOGFISH_DATA * *dog_dat*)
- 5.1.1.6 int DOGFISH_Executioner (DOGFISH_DATA * *dog_dat*)
- 5.1.1.7 int DOGFISH_postprocesses (DOGFISH_DATA * *dog_dat*)
- 5.1.1.8 int DOGFISH_preprocesses (DOGFISH_DATA * *dog_dat*)
- 5.1.1.9 int DOGFISH_reset (DOGFISH_DATA * *dog_dat*)
- 5.1.1.10 int DOGFISH_TESTS ()
- 5.1.1.11 void print2file_DOGFISH_header (DOGFISH_DATA * *dog_dat*)
- 5.1.1.12 void print2file_DOGFISH_result_new (DOGFISH_DATA * *dog_dat*)
- 5.1.1.13 void print2file_DOGFISH_result_old (DOGFISH_DATA * *dog_dat*)
- 5.1.1.14 void print2file_species_header (FILE * *Output*, DOGFISH_DATA * *dog_dat*, int *i*)
- 5.1.1.15 int set_DOGFISH_ICs (DOGFISH_DATA * *dog_dat*)
- 5.1.1.16 int set_DOGFISH_params (const void * *user_data*)
- 5.1.1.17 int set_DOGFISH_timestep (DOGFISH_DATA * *dog_dat*)
- 5.1.1.18 int setup_DOGFISH_DATA (FILE * *file*, double(*)(int i, int l, const void **user_data*) *eval_R*, double(*)(int i, int l, const void **user_data*) *eval_DI*, double(*)(int i, const void **user_data*) *eval_kf*, double(*)(int i, const void **user_data*) *eval_qs*, const void * *user_data*, DOGFISH_DATA * *dog_dat*)

5.2 dogfish.h File Reference

```
#include "finch.h"
#include "mola.h"
```

Classes

- struct [DOGFISH_PARAM](#)
- struct [DOGFISH_DATA](#)

Functions

- void [print2file_species_header](#) (FILE **Output*, [DOGFISH_DATA](#) **dog_dat*, int *i*)
- void [print2file_DOGFISH_header](#) ([DOGFISH_DATA](#) **dog_dat*)
- void [print2file_DOGFISH_result_old](#) ([DOGFISH_DATA](#) **dog_dat*)
- void [print2file_DOGFISH_result_new](#) ([DOGFISH_DATA](#) **dog_dat*)
- double [default_Retardation](#) (int *i*, int *l*, const void **data*)
- double [default_IntraDiffusion](#) (int *i*, int *l*, const void **data*)
- double [default_FilmMTCoeff](#) (int *i*, const void **data*)
- double [default_SurfaceConcentration](#) (int *i*, const void **data*)
- int [setup_DOGFISH_DATA](#) (FILE **file*, double(**eval_R*)(int *i*, int *l*, const void **user_data*), double(**eval_DI*)(int *i*, int *l*, const void **user_data*), double(**eval_kf*)(int *i*, const void **user_data*), double(**eval_qs*)(int *i*, const void **user_data*), const void **user_data*, [DOGFISH_DATA](#) **dog_dat*)

- int DOGFISH_Executioner (DOGFISH_DATA *dog_dat)
- int set_DOGFISH_ICs (DOGFISH_DATA *dog_dat)
- int set_DOGFISH_timestep (DOGFISH_DATA *dog_dat)
- int DOGFISH_preprocesses (DOGFISH_DATA *dog_dat)
- int set_DOGFISH_params (const void *user_data)
- int DOGFISH_postprocesses (DOGFISH_DATA *dog_dat)
- int DOGFISH_reset (DOGFISH_DATA *dog_dat)
- int DOGFISH (DOGFISH_DATA *dog_dat)
- int DOGFISH_TESTS ()

5.2.1 Function Documentation

5.2.1.1 double default_FilmMTCoeff (int *i*, const void * *data*)

5.2.1.2 double default_IntraDiffusion (int *i*, int *l*, const void * *data*)

5.2.1.3 double default_Retardation (int *i*, int *l*, const void * *data*)

5.2.1.4 double default_SurfaceConcentration (int *i*, const void * *data*)

5.2.1.5 int DOGFISH (DOGFISH_DATA * *dog_dat*)

5.2.1.6 int DOGFISH_Executioner (DOGFISH_DATA * *dog_dat*)

5.2.1.7 int DOGFISH_postprocesses (DOGFISH_DATA * *dog_dat*)

5.2.1.8 int DOGFISH_preprocesses (DOGFISH_DATA * *dog_dat*)

5.2.1.9 int DOGFISH_reset (DOGFISH_DATA * *dog_dat*)

5.2.1.10 int DOGFISH_TESTS ()

5.2.1.11 void print2file_DOGFISH_header (DOGFISH_DATA * *dog_dat*)

5.2.1.12 void print2file_DOGFISH_result_new (DOGFISH_DATA * *dog_dat*)

5.2.1.13 void print2file_DOGFISH_result_old (DOGFISH_DATA * *dog_dat*)

5.2.1.14 void print2file_species_header (FILE * *Output*, DOGFISH_DATA * *dog_dat*, int *i*)

5.2.1.15 int set_DOGFISH_ICs (DOGFISH_DATA * *dog_dat*)

5.2.1.16 int set_DOGFISH_params (const void * *user_data*)

5.2.1.17 int set_DOGFISH_timestep (DOGFISH_DATA * *dog_dat*)

5.2.1.18 int setup_DOGFISH_DATA (FILE * *file*, double(*) (int *i*, int *l*, const void * *user_data*) *eval_R*, double(*) (int *i*, int *l*, const void * *user_data*) *eval_DI*, double(*) (int *i*, const void * *user_data*) *eval_kf*, double(*) (int *i*, const void * *user_data*) *eval_qs*, const void * *user_data*, DOGFISH_DATA * *dog_dat*)

5.3 eel.cpp File Reference

```
#include "eel.h"
```

Functions

- int [EEL_TESTS](#) ()

5.3.1 Function Documentation

5.3.1.1 int EEL_TESTS ()

5.4 eel.h File Reference

```
#include <stdio.h>
#include <math.h>
#include <iostream>
#include <fstream>
#include <stdlib.h>
#include <vector>
#include <time.h>
#include <float.h>
#include <string>
#include "error.h"
```

Classes

- class [Atom](#)
- class [PeriodicTable](#)

Functions

- int [EEL_TESTS](#) ()

5.4.1 Function Documentation

5.4.1.1 int EEL_TESTS ()

5.5 egret.cpp File Reference

```
#include "egret.h"
```

Functions

- int [initialize_data](#) (int N, [MIXED_GAS](#) *gas_dat)
- int [set_variables](#) (double PT, double T, double us, double L, std::vector< double > &y, [MIXED_GAS](#) *gas_dat)
- int [calculate_properties](#) ([MIXED_GAS](#) *gas_dat)
- int [EGRET_TESTS](#) ()

5.5.1 Function Documentation

5.5.1.1 int calculate_properties ([MIXED_GAS](#) * *gas_dat*)

5.5.1.2 int EGRET_TESTS ()

5.5.1.3 int initialize_data (int *N*, MIXED_GAS * *gas_dat*)

5.5.1.4 int set_variables (double *PT*, double *T*, double *us*, double *L*, std::vector< double > & *y*, MIXED_GAS * *gas_dat*)

5.6 egret.h File Reference

```
#include "macaw.h"
```

Classes

- struct [PURE_GAS](#)
- struct [MIXED_GAS](#)

Macros

- #define [Rstd](#) 8.3144621
- #define [RE3](#) 8.3144621E+3
- #define [Po](#) 100.0
- #define [Cstd](#)(*p*, *T*) ((*p*)/([Rstd](#)**T*))
- #define [CE3](#)(*p*, *T*) ((*p*)/([RE3](#)**T*))
- #define [Pstd](#)(*c*, *T*) ((*c*)*[Rstd](#)**T*)
- #define [PE3](#)(*c*, *T*) ((*c*)*[RE3](#)**T*)
- #define [Nu](#)(*mu*, *rho*) ((*mu*)/(*rho*))
- #define [PSI](#)(*T*) (0.873143 + (0.000072375**T*))
- #define [Dp_ij](#)(*Dij*, *PT*) ((*PT***Dij*)/[Po](#))
- #define [D_ij](#)(*MWi*, *MWj*, *rhoi*, *rhoj*, *mui*, *muj*) ((4.0 / sqrt(2.0)) * pow((((1/*MWi*)+(1/*MWj*)),0.5)) / pow((pow((pow((*rhoi*/(1.385**mui*))),2.0)/*MWi*),0.25)+ pow((pow((*rhoj*/(1.385**muj*))),2.0)/*MWj*),0.25)),2.0)
- #define [Mu](#)(*muo*, *To*, *C*, *T*) (*muo* * ((*To* + *C*)/(*T* + *C*)) * pow((*T*/*To*),1.5))
- #define [D_ii](#)(*rhoi*, *mui*) (1.385**mui*/*rhoi*)
- #define [ReNum](#)(*u*, *L*, *nu*) (*u***L*/*nu*)
- #define [ScNum](#)(*nu*, *D*) (*nu*/*D*)
- #define [FilmMTCoeff](#)(*D*, *L*, *Re*, *Sc*) ((*D*/*L*)*(2.0 + (1.1*pow(*Re*,0.6)*pow(*Sc*,0.3))))

Functions

- int [initialize_data](#) (int *N*, MIXED_GAS **gas_dat*)
- int [set_variables](#) (double *PT*, double *T*, double *us*, double *L*, std::vector< double > &*y*, MIXED_GAS **gas_dat*)
- int [calculate_properties](#) (MIXED_GAS **gas_dat*)
- int [EGRET_TESTS](#) ()

5.6.1 Macro Definition Documentation

5.6.1.1 #define [CE3](#)(*p*, *T*) ((*p*)/([RE3](#)**T*))

5.6.1.2 #define [Cstd](#)(*p*, *T*) ((*p*)/([Rstd](#)**T*))

5.6.1.3 #define [D_ii](#)(*rhoi*, *mui*) (1.385**mui*/*rhoi*)

```
5.6.1.4 #define D_ij( MWi, MWj, rhoi, rhoj, mui, muj ) ( (4.0 / sqrt(2.0)) * pow(((1/MWi)+(1/MWj)),0.5) ) / pow(
        (pow((pow((rhoi/(1.385*mui)),2.0)/MWi),0.25)+ pow((pow((rhoj/(1.385*muj)),2.0)/MWj),0.25)),2.0 )
```

```
5.6.1.5 #define Dp_ij( Dij, PT ) ((PT*Dij)/Po)
```

```
5.6.1.6 #define FilmMTCoeff( D, L, Re, Sc ) ((D/L)*(2.0 + (1.1*pow(Re,0.6)*pow(Sc,0.3))))
```

```
5.6.1.7 #define Mu( muo, To, C, T ) (muo * ((To + C)/(T + C)) * pow((T/To),1.5) )
```

```
5.6.1.8 #define Nu( mu, rho ) ((mu)/(rho))
```

```
5.6.1.9 #define PE3( c, T ) ((c)*RE3*T)
```

```
5.6.1.10 #define Po 100.0
```

```
5.6.1.11 #define PSI( T ) (0.873143 + (0.000072375*T))
```

```
5.6.1.12 #define Pstd( c, T ) ((c)*Rstd*T)
```

```
5.6.1.13 #define RE3 8.3144621E+3
```

```
5.6.1.14 #define ReNum( u, L, nu ) (u*L/nu)
```

```
5.6.1.15 #define Rstd 8.3144621
```

```
5.6.1.16 #define ScNum( nu, D ) (nu/D)
```

5.6.2 Function Documentation

```
5.6.2.1 int calculate_properties ( MIXED_GAS * gas_dat )
```

```
5.6.2.2 int EGRET_TESTS ( )
```

```
5.6.2.3 int initialize_data ( int N, MIXED_GAS * gas_dat )
```

```
5.6.2.4 int set_variables ( double PT, double T, double us, double L, std::vector< double > & y, MIXED_GAS * gas_dat )
```

5.7 error.cpp File Reference

```
#include "error.h"
```

Functions

- void [error](#) (int flag)

5.7.1 Function Documentation

```
5.7.1.1 void error ( int flag )
```

5.8 error.h File Reference

```
#include <iostream>
```


Macros

- #define `mError(i)`

Enumerations

- enum `error_type` {
 `generic_error`, `file_dne`, `indexing_error`, `magpie_reverse_error`,
 `simulation_fail`, `invalid_components`, `invalid_boolean`, `invalid_molefraction`,
 `invalid_gas_sum`, `invalid_solid_sum`, `scenario_fail`, `out_of_bounds`,
 `non_square_matrix`, `dim_mis_match`, `empty_matrix`, `opt_no_support`,
 `invalid_fraction`, `ortho_check_fail`, `unstable_matrix`, `no_diffusion`,
 `negative_mass`, `negative_time`, `matvec_mis_match`, `arg_matrix_same`,
 `singular_matrix`, `matrix_too_small`, `invalid_size`, `nullptr_func`,
 `invalid_norm`, `vector_out_of_bounds`, `zero_vector`, `tensor_out_of_bounds`,
 `non_real_edge`, `nullptr_error`, `invalid_atom`, `invalid_proton`,
 `invalid_neutron`, `invalid_electron`, `invalid_valence`, `string_parse_error`,
 `unregistered_name`, `rxn_rate_error`, `invalid_species`, `duplicate_variable`,
 `missing_information`, `invalid_type`, `key_not_found`, `anchor_alias_dne`,
 `initial_error`, `not_a_token`, `read_error`, `invalid_console_input` }

Functions

- void `error` (int flag)

5.8.1 Macro Definition Documentation

5.8.1.1 #define `mError(i)`

Value:

```
{error(i);  
std::cout << "Source: " << __FILE__ << "\nLine: " << __LINE__ << std::endl;}
```

5.8.2 Enumeration Type Documentation

5.8.2.1 enum `error_type`

Enumerator

generic_error
file_dne
indexing_error
magpie_reverse_error
simulation_fail
invalid_components
invalid_boolean
invalid_molefraction
invalid_gas_sum
invalid_solid_sum
scenario_fail
out_of_bounds

non_square_matrix
dim_mis_match
empty_matrix
opt_no_support
invalid_fraction
ortho_check_fail
unstable_matrix
no_diffusion
negative_mass
negative_time
matvec_mis_match
arg_matrix_same
singular_matrix
matrix_too_small
invalid_size
nullptr_func
invalid_norm
vector_out_of_bounds
zero_vector
tensor_out_of_bounds
non_real_edge
nullptr_error
invalid_atom
invalid_proton
invalid_neutron
invalid_electron
invalid_valence
string_parse_error
unregistered_name
rxn_rate_error
invalid_species
duplicate_variable
missing_information
invalid_type
key_not_found
anchor_alias_dne
initial_error
not_a_token
read_error
invalid_console_input

5.8.3 Function Documentation

5.8.3.1 void error (int *flag*)

5.9 finch.cpp File Reference

```
#include "finch.h"
```

Functions

- double [max](#) (std::vector< double > &values)
- double [min](#) (std::vector< double > &values)
- double [minmod](#) (std::vector< double > &values)
- int [uTotal](#) (FINCH_DATA *dat)
- int [uAverage](#) (FINCH_DATA *dat)
- int [check_Mass](#) (FINCH_DATA *dat)
- int [l_direct](#) (FINCH_DATA *dat)
- int [lark_picard_step](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &G, const void *data)
- int [nl_picard](#) (FINCH_DATA *dat)
- int [setup_FINCH_DATA](#) (int(*user_callroutine)(const void *user_data), int(*user_setic)(const void *user_data), int(*user_timestep)(const void *user_data), int(*user_preprocess)(const void *user_data), int(*user_solve)(const void *user_data), int(*user_setparams)(const void *user_data), int(*user_discretize)(const void *user_data), int(*user_bcs)(const void *user_data), int(*user_res)(const [Matrix](#)< double > &x, [Matrix](#)< double > &res, const void *user_data), int(*user_precon)(const [Matrix](#)< double > &b, [Matrix](#)< double > &p, const void *user_data), int(*user_postprocess)(const void *user_data), int(*user_reset)(const void *user_data), FINCH_DATA *dat, const void *param_data)
- void [print2file_dim_header](#) (FILE *Output, FINCH_DATA *dat)
- void [print2file_time_header](#) (FILE *Output, FINCH_DATA *dat)
- void [print2file_result_old](#) (FILE *Output, FINCH_DATA *dat)
- void [print2file_result_new](#) (FILE *Output, FINCH_DATA *dat)
- void [print2file_newline](#) (FILE *Output, FINCH_DATA *dat)
- void [print2file_tab](#) (FILE *Output, FINCH_DATA *dat)
- int [default_execution](#) (const void *user_data)
- int [default_ic](#) (const void *user_data)
- int [default_timestep](#) (const void *user_data)
- int [default_preprocess](#) (const void *user_data)
- int [default_solve](#) (const void *user_data)
- int [default_params](#) (const void *user_data)
- int [minmod_discretization](#) (const void *user_data)
- int [vanAlbada_discretization](#) (const void *user_data)
- int [ospre_discretization](#) (const void *user_data)
- int [default_bcs](#) (const void *user_data)
- int [default_res](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &res, const void *user_data)
- int [default_precon](#) (const [Matrix](#)< double > &b, [Matrix](#)< double > &p, const void *user_data)
- int [default_postprocess](#) (const void *user_data)
- int [default_reset](#) (const void *user_data)
- int [buckley_leverett_ic](#) (const void *user_data)
- int [buckley_leverett_params](#) (const void *user_data)
- int [burgers_ic](#) (const void *user_data)
- int [burgers_params](#) (const void *user_data)
- int [burgers_bcs](#) (const void *user_data)
- int [FINCH_TESTS](#) ()

5.9.1 Function Documentation

5.9.1.1 [int buckley_leverett_ic \(const void * user_data \)](#)

5.9.1.2 [int buckley_leverett_params \(const void * user_data \)](#)

5.9.1.3 [int burgers_bcs \(const void * user_data \)](#)

5.9.1.4 [int burgers_ic \(const void * user_data \)](#)

- 5.9.1.5 `int burgers_params (const void * user_data)`
- 5.9.1.6 `int check_Mass (FINCH_DATA * dat)`
- 5.9.1.7 `int default_bcs (const void * user_data)`
- 5.9.1.8 `int default_execution (const void * user_data)`
- 5.9.1.9 `int default_ic (const void * user_data)`
- 5.9.1.10 `int default_params (const void * user_data)`
- 5.9.1.11 `int default_postprocess (const void * user_data)`
- 5.9.1.12 `int default_precon (const Matrix< double > & b, Matrix< double > & p, const void * user_data)`
- 5.9.1.13 `int default_preprocess (const void * user_data)`
- 5.9.1.14 `int default_res (const Matrix< double > & x, Matrix< double > & res, const void * user_data)`
- 5.9.1.15 `int default_reset (const void * user_data)`
- 5.9.1.16 `int default_solve (const void * user_data)`
- 5.9.1.17 `int default_timestep (const void * user_data)`
- 5.9.1.18 `int FINCH_TESTS ()`
- 5.9.1.19 `int l_direct (FINCH_DATA * dat)`
- 5.9.1.20 `int lark_picard_step (const Matrix< double > & x, Matrix< double > & G, const void * data)`
- 5.9.1.21 `double max (std::vector< double > & values)`
- 5.9.1.22 `double min (std::vector< double > & values)`
- 5.9.1.23 `double minmod (std::vector< double > & values)`
- 5.9.1.24 `int minmod_discretization (const void * user_data)`
- 5.9.1.25 `int nl_picard (FINCH_DATA * dat)`
- 5.9.1.26 `int ospre_discretization (const void * user_data)`
- 5.9.1.27 `void print2file_dim_header (FILE * Output, FINCH_DATA * dat)`
- 5.9.1.28 `void print2file_newline (FILE * Output, FINCH_DATA * dat)`
- 5.9.1.29 `void print2file_result_new (FILE * Output, FINCH_DATA * dat)`
- 5.9.1.30 `void print2file_result_old (FILE * Output, FINCH_DATA * dat)`
- 5.9.1.31 `void print2file_tab (FILE * Output, FINCH_DATA * dat)`
- 5.9.1.32 `void print2file_time_header (FILE * Output, FINCH_DATA * dat)`

- 5.9.1.33 `int setup_FINCH_DATA (int (*)(const void *user_data) user_callroutine, int (*)(const void *user_data) user_setic, int (*)(const void *user_data) user_timestep, int (*)(const void *user_data) user_preprocess, int (*)(const void *user_data) user_solve, int (*)(const void *user_data) user_setparams, int (*)(const void *user_data) user_discretize, int (*)(const void *user_data) user_bcs, int (*)(const Matrix< double > &x, Matrix< double > &res, const void *user_data) user_res, int (*)(const Matrix< double > &b, Matrix< double > &p, const void *user_data) user_precon, int (*)(const void *user_data) user_postprocess, int (*)(const void *user_data) user_reset, FINCH_DATA * dat, const void * param_data)`
- 5.9.1.34 `int uAverage (FINCH_DATA * dat)`
- 5.9.1.35 `int uTotal (FINCH_DATA * dat)`
- 5.9.1.36 `int vanAlbada_discretization (const void * user_data)`

5.10 finch.h File Reference

```
#include "macaw.h"
#include "lark.h"
```

Classes

- struct [FINCH_DATA](#)

Macros

- `#define` [FINCH_Picard](#) 0
- `#define` [LARK_Picard](#) 1
- `#define` [LARK_PJFNK](#) 2
- `#define` [Cartesian](#) 0
- `#define` [Cylindrical](#) 1
- `#define` [Spherical](#) 2

Functions

- double [max](#) (std::vector< double > &values)
- double [min](#) (std::vector< double > &values)
- double [minmod](#) (std::vector< double > &values)
- int [uTotal](#) (FINCH_DATA *dat)
- int [uAverage](#) (FINCH_DATA *dat)
- int [check_Mass](#) (FINCH_DATA *dat)
- int [l_direct](#) (FINCH_DATA *dat)
- int [lark_picard_step](#) (const Matrix< double > &x, Matrix< double > &G, const void *data)
- int [nl_picard](#) (FINCH_DATA *dat)
- int [setup_FINCH_DATA](#) (int(*user_callroutine)(const void *user_data), int(*user_setic)(const void *user_data), int(*user_timestep)(const void *user_data), int(*user_preprocess)(const void *user_data), int(*user_solve)(const void *user_data), int(*user_setparams)(const void *user_data), int(*user_discretize)(const void *user_data), int(*user_bcs)(const void *user_data), int(*user_res)(const Matrix< double > &x, Matrix< double > &res, const void *user_data), int(*user_precon)(const Matrix< double > &b, Matrix< double > &p, const void *user_data), int(*user_postprocess)(const void *user_data), int(*user_reset)(const void *user_data), FINCH_DATA *dat, const void *param_data)
- void [print2file_dim_header](#) (FILE *Output, FINCH_DATA *dat)
- void [print2file_time_header](#) (FILE *Output, FINCH_DATA *dat)
- void [print2file_result_old](#) (FILE *Output, FINCH_DATA *dat)

- void [print2file_result_new](#) (FILE *Output, [FINCH_DATA](#) *dat)
- void [print2file_newline](#) (FILE *Output, [FINCH_DATA](#) *dat)
- void [print2file_tab](#) (FILE *Output, [FINCH_DATA](#) *dat)
- int [default_execution](#) (const void *user_data)
- int [default_ic](#) (const void *user_data)
- int [default_timestep](#) (const void *user_data)
- int [default_preprocess](#) (const void *user_data)
- int [default_solve](#) (const void *user_data)
- int [default_params](#) (const void *user_data)
- int [minmod_discretization](#) (const void *user_data)
- int [vanAlbada_discretization](#) (const void *user_data)
- int [ospre_discretization](#) (const void *user_data)
- int [default_bcs](#) (const void *user_data)
- int [default_res](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &res, const void *user_data)
- int [default_precon](#) (const [Matrix](#)< double > &b, [Matrix](#)< double > &p, const void *user_data)
- int [default_postprocess](#) (const void *user_data)
- int [default_reset](#) (const void *user_data)
- int [buckley_leverett_ic](#) (const void *user_data)
- int [buckley_leverett_params](#) (const void *user_data)
- int [burgers_ic](#) (const void *user_data)
- int [burgers_params](#) (const void *user_data)
- int [burgers_bcs](#) (const void *user_data)
- int [FINCH_TESTS](#) ()

5.10.1 Macro Definition Documentation

5.10.1.1 [#define Cartesian](#) 0

5.10.1.2 [#define Cylindrical](#) 1

5.10.1.3 [#define FINCH_Picard](#) 0

5.10.1.4 [#define LARK_Picard](#) 1

5.10.1.5 [#define LARK_PJFNK](#) 2

5.10.1.6 [#define Spherical](#) 2

5.10.2 Function Documentation

5.10.2.1 int [buckley_leverett_ic](#) (const void * *user_data*)

5.10.2.2 int [buckley_leverett_params](#) (const void * *user_data*)

5.10.2.3 int [burgers_bcs](#) (const void * *user_data*)

5.10.2.4 int [burgers_ic](#) (const void * *user_data*)

5.10.2.5 int [burgers_params](#) (const void * *user_data*)

5.10.2.6 int [check_Mass](#) ([FINCH_DATA](#) * *dat*)

5.10.2.7 int [default_bcs](#) (const void * *user_data*)

- 5.10.2.8 int default_execution (const void * *user_data*)
- 5.10.2.9 int default_ic (const void * *user_data*)
- 5.10.2.10 int default_params (const void * *user_data*)
- 5.10.2.11 int default_postprocess (const void * *user_data*)
- 5.10.2.12 int default_precon (const Matrix< double > & *b*, Matrix< double > & *p*, const void * *user_data*)
- 5.10.2.13 int default_preprocess (const void * *user_data*)
- 5.10.2.14 int default_res (const Matrix< double > & *x*, Matrix< double > & *res*, const void * *user_data*)
- 5.10.2.15 int default_reset (const void * *user_data*)
- 5.10.2.16 int default_solve (const void * *user_data*)
- 5.10.2.17 int default_timestep (const void * *user_data*)
- 5.10.2.18 int FINCH_TESTS ()
- 5.10.2.19 int l_direct (FINCH_DATA * *dat*)
- 5.10.2.20 int lark_picard_step (const Matrix< double > & *x*, Matrix< double > & *G*, const void * *data*)
- 5.10.2.21 double max (std::vector< double > & *values*)
- 5.10.2.22 double min (std::vector< double > & *values*)
- 5.10.2.23 double minmod (std::vector< double > & *values*)
- 5.10.2.24 int minmod_discretization (const void * *user_data*)
- 5.10.2.25 int nl_picard (FINCH_DATA * *dat*)
- 5.10.2.26 int ospre_discretization (const void * *user_data*)
- 5.10.2.27 void print2file_dim_header (FILE * *Output*, FINCH_DATA * *dat*)
- 5.10.2.28 void print2file_newline (FILE * *Output*, FINCH_DATA * *dat*)
- 5.10.2.29 void print2file_result_new (FILE * *Output*, FINCH_DATA * *dat*)
- 5.10.2.30 void print2file_result_old (FILE * *Output*, FINCH_DATA * *dat*)
- 5.10.2.31 void print2file_tab (FILE * *Output*, FINCH_DATA * *dat*)
- 5.10.2.32 void print2file_time_header (FILE * *Output*, FINCH_DATA * *dat*)

```
5.10.2.33 int setup_FINCH_DATA ( int(*) (const void *user_data) user_callroutine, int(*) (const void *user_data) user_setic,
int(*) (const void *user_data) user_timestep, int(*) (const void *user_data) user_preprocess, int(*) (const
void *user_data) user_solve, int(*) (const void *user_data) user_setparams, int(*) (const void *user_data)
user_discretize, int(*) (const void *user_data) user_bcs, int(*) (const Matrix< double > &x, Matrix< double >
&res, const void *user_data) user_res, int(*) (const Matrix< double > &b, Matrix< double > &p, const void
*user_data) user_precon, int(*) (const void *user_data) user_postprocess, int(*) (const void *user_data) user_reset,
FINCH_DATA * dat, const void * param_data )
```

```
5.10.2.34 int uAverage ( FINCH_DATA * dat )
```

```
5.10.2.35 int uTotal ( FINCH_DATA * dat )
```

```
5.10.2.36 int vanAlbada_discretization ( const void * user_data )
```

5.11 flock.h File Reference

```
#include "macaw.h"
#include "egret.h"
#include "finch.h"
#include "lark.h"
#include "skua.h"
#include "scopsowl.h"
#include "gsta_opt.h"
#include "magpie.h"
#include "skua_opt.h"
#include "scopsowl_opt.h"
#include "yaml_wrapper.h"
```

5.12 gsta_opt.cpp File Reference

```
#include "gsta_opt.h"
```

Functions

- int [roundIt](#) (double d)
- int [twoFifths](#) (int m)
- int [orderMag](#) (double x)
- int [minValue](#) (std::vector< int > array)
- int [minIndex](#) (std::vector< double > array)
- int [avgPar](#) (std::vector< int > array)
- double [avgValue](#) (std::vector< double > array)
- double [weightedAvg](#) (double *enorm, double *x, int n)
- double [rSq](#) (double *x, double *y, double slope, double vint, int m_dat)
- bool [isSmooth](#) (double *par, void *data)
- void [orthoLinReg](#) (double *x, double *y, double *par, int m_dat, int n_par)
- void [eduGuess](#) (double *P, double *q, double *par, int k, int m_dat, void *data)
- double [gstaFunc](#) (double p, const double *K, double qmax, int n_par)
- double [gstaObjFunc](#) (double *t, double *y, double *par, int m_dat, void *data)
- void [eval_GSTA](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- int [gsta_optimize](#) (const char *fileName)

5.12.1 Function Documentation

5.12.1.1 `int avgPar (std::vector< int > array)`

5.12.1.2 `double avgValue (std::vector< double > array)`

5.12.1.3 `void eduGuess (double * P, double * q, double * par, int k, int m_dat, void * data)`

5.12.1.4 `void eval_GSTA (const double * par, int m_dat, const void * data, double * fvec, int * info)`

5.12.1.5 `int gsta_optimize (const char * fileName)`

5.12.1.6 `double gstaFunc (double p, const double * K, double qmax, int n_par)`

5.12.1.7 `double gstaObjFunc (double * t, double * y, double * par, int m_dat, void * data)`

5.12.1.8 `bool isSmooth (double * par, void * data)`

5.12.1.9 `int minIndex (std::vector< double > array)`

5.12.1.10 `int minValue (std::vector< int > array)`

5.12.1.11 `int orderMag (double x)`

5.12.1.12 `void orthoLinReg (double * x, double * y, double * par, int m_dat, int n_par)`

5.12.1.13 `int roundIt (double d)`

5.12.1.14 `double rSq (double * x, double * y, double slope, double vint, int m_dat)`

5.12.1.15 `int twoFifths (int m)`

5.12.1.16 `double weightedAvg (double * enorm, double * x, int n)`

5.13 gsta_opt.h File Reference

```
#include "lmcurve.h"
#include <stdio.h>
#include <math.h>
#include <iostream>
#include <fstream>
#include <stdlib.h>
#include <vector>
#include <time.h>
#include <float.h>
#include <string>
#include "error.h"
```

Classes

- struct [GSTA_OPT_DATA](#)

Macros

- `#define Po 100.0`
- `#define R 8.3144621`
- `#define Na 6.0221413E+23`

Functions

- `void error ()`
- `int roundIt (double d)`
- `int twoFifths (int m)`
- `int orderMag (double x)`
- `int minVal (std::vector< int > array)`
- `int minIndex (std::vector< double > array)`
- `int avgPar (std::vector< int > array)`
- `double avgValue (std::vector< double > array)`
- `double weightedAvg (double *enorm, double *x, int n)`
- `double rSq (double *x, double *y, double slope, double vint, int m_dat)`
- `bool isSmooth (double *par, void *data)`
- `void orthoLinReg (double *x, double *y, double *par, int m_dat, int n_par)`
- `void eduGuess (double *P, double *q, double *par, int k, int m_dat, void *data)`
- `double gstaFunc (double p, const double *K, double qmax, int n_par)`
- `double gstaObjFunc (double *t, double *y, double *par, int m_dat, void *data)`
- `void eval_GSTA (const double *par, int m_dat, const void *data, double *fvec, int *info)`
- `int gsta_optimize (const char *fileName)`

5.13.1 Macro Definition Documentation

5.13.1.1 `#define Na 6.0221413E+23`

5.13.1.2 `#define Po 100.0`

5.13.1.3 `#define R 8.3144621`

5.13.2 Function Documentation

5.13.2.1 `int avgPar (std::vector< int > array)`

5.13.2.2 `double avgValue (std::vector< double > array)`

5.13.2.3 `void eduGuess (double * P, double * q, double * par, int k, int m_dat, void * data)`

5.13.2.4 `void error ()`

5.13.2.5 `void eval_GSTA (const double * par, int m_dat, const void * data, double * fvec, int * info)`

5.13.2.6 `int gsta_optimize (const char * fileName)`

5.13.2.7 `double gstaFunc (double p, const double * K, double qmax, int n_par)`

5.13.2.8 `double gstaObjFunc (double * t, double * y, double * par, int m_dat, void * data)`

5.13.2.9 `bool isSmooth (double * par, void * data)`

- 5.13.2.10 int minIndex (std::vector< double > array)
- 5.13.2.11 int minValue (std::vector< int > array)
- 5.13.2.12 int orderMag (double x)
- 5.13.2.13 void orthoLinReg (double * x, double * y, double * par, int m_dat, int n_par)
- 5.13.2.14 int roundlt (double d)
- 5.13.2.15 double rSq (double * x, double * y, double slope, double vint, int m_dat)
- 5.13.2.16 int twoFifths (int m)
- 5.13.2.17 double weightedAvg (double * enorm, double * x, int n)

5.14 lark.cpp File Reference

```
#include "lark.h"
```

Functions

- int [matvec_ex01](#) (const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data)
- int [precon_ex01](#) (const [Matrix](#)< double > &b, [Matrix](#)< double > &p, const void *data)
- int [matvec_ex02](#) (const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data)
- int [matvec_ex04](#) (const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data)
- int [precon_ex04](#) (const [Matrix](#)< double > &b, [Matrix](#)< double > &p, const void *data)
- int [evalx_ex09](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &G, const void *data)
- int [funeval_ex09](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &F, const void *data)
- int [funeval_ex10](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &F, const void *data)
- int [precon_ex10](#) (const [Matrix](#)< double > &r, [Matrix](#)< double > &p, const void *data)
- int [matvec_ex15](#) (const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data)
- int [precon_ex15](#) (const [Matrix](#)< double > &w, [Matrix](#)< double > &p, const void *data)
- int [update_arnoldi_solution](#) ([Matrix](#)< double > &x, [Matrix](#)< double > &x0, [ARNOLDI_DATA](#) *arnoldi_dat)
- int [arnoldi](#) (int(*matvec)(const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data), int(*precon)(const [Matrix](#)< double > &b, [Matrix](#)< double > &p, const void *data), [Matrix](#)< double > &r0, [ARNOLDI_DATA](#) *arnoldi_dat, const void *matvec_data, const void *precon_data)
- int [gmresLeftPreconditioned](#) (int(*matvec)(const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data), int(*precon)(const [Matrix](#)< double > &b, [Matrix](#)< double > &P, const void *data), [Matrix](#)< double > &b, [GMRESLP_DATA](#) *gmreslp_dat, const void *matvec_data, const void *precon_data)
- int [fom](#) (int(*matvec)(const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data), int(*precon)(const [Matrix](#)< double > &b, [Matrix](#)< double > &P, const void *data), [Matrix](#)< double > &b, [GMRESLP_DATA](#) *gmreslp_dat, const void *matvec_data, const void *precon_data)
- int [gmresRightPreconditioned](#) (int(*matvec)(const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data), int(*precon)(const [Matrix](#)< double > &b, [Matrix](#)< double > &p, const void *data), [Matrix](#)< double > &b, [GMRESRP_DATA](#) *gmresrp_dat, const void *matvec_data, const void *precon_data)
- int [pcg](#) (int(*matvec)(const [Matrix](#)< double > &p, [Matrix](#)< double > &Ap, const void *data), int(*precon)(const [Matrix](#)< double > &r, [Matrix](#)< double > &z, const void *data), [Matrix](#)< double > &b, [PCG_DATA](#) *pcg_dat, const void *matvec_data, const void *precon_data)
- int [bicgstab](#) (int(*matvec)(const [Matrix](#)< double > &p, [Matrix](#)< double > &Ap, const void *data), int(*precon)(const [Matrix](#)< double > &r, [Matrix](#)< double > &z, const void *data), [Matrix](#)< double > &b, [BICGSTAB_DATA](#) *bicg_dat, const void *matvec_data, const void *precon_data)

- `int cgs (int(*matvec)(const Matrix< double > &p, Matrix< double > &Ap, const void *data), int(*precon)(const Matrix< double > &r, Matrix< double > &z, const void *data), Matrix< double > &b, CGS_DATA *cgs_dat, const void *matvec_data, const void *precon_data)`
- `int operatorTranspose (int(*matvec)(const Matrix< double > &v, Matrix< double > &Av, const void *data), Matrix< double > &r, Matrix< double > &u, OPTRANS_DATA *transpose_dat, const void *matvec_data)`
- `int gcr (int(*matvec)(const Matrix< double > &x, Matrix< double > &Ax, const void *data), int(*precon)(const Matrix< double > &r, Matrix< double > &Mr, const void *data), Matrix< double > &b, GCR_DATA *gcr_dat, const void *matvec_data, const void *precon_data)`
- `int gmresPreconditioner (const Matrix< double > &r, Matrix< double > &Mr, const void *data)`
- `int gmres (int(*matvec)(const Matrix< double > &x, Matrix< double > &Ax, const void *data), int(*terminal_precon)(const Matrix< double > &r, Matrix< double > &Mr, const void *data), Matrix< double > &b, GMRESR_DATA *gmresr_dat, const void *matvec_data, const void *term_precon_data)`
- `int picard (int(*res)(const Matrix< double > &x, Matrix< double > &r, const void *data), int(*evalx)(const Matrix< double > &x0, Matrix< double > &x, const void *data), Matrix< double > &x, PICARD_DATA *picard_dat, const void *res_data, const void *evalx_data)`
- `int jacvec (const Matrix< double > &v, Matrix< double > &Jv, const void *data)`
- `int backtrackLineSearch (int(*feval)(const Matrix< double > &x, Matrix< double > &F, const void *data), Matrix< double > &Fkp1, Matrix< double > &xkp1, Matrix< double > &pk, double normFk, BACKTRACK_DATA *backtrack_dat, const void *feval_data)`
- `int pjfnk (int(*res)(const Matrix< double > &x, Matrix< double > &F, const void *data), int(*precon)(const Matrix< double > &r, Matrix< double > &p, const void *data), Matrix< double > &x, PJFNK_DATA *pjfnk_dat, const void *res_data, const void *precon_data)`
- `int NumericalJacobian (int(*Func)(const Matrix< double > &x, Matrix< double > &F, const void *user_data), const Matrix< double > &x, Matrix< double > &J, int Nx, int Nf, NUM_JAC_DATA *jac_dat, const void *user_data)`
- `int LARK_TESTS ()`

5.14.1 Function Documentation

- 5.14.1.1 `int arnoldi (int(*) (const Matrix< double > &v, Matrix< double > &w, const void *data) matvec, int(*) (const Matrix< double > &b, Matrix< double > &p, const void *data) precon, Matrix< double > &r0, ARNOLDI_DATA * arnoldi_dat, const void * matvec_data, const void * precon_data)`
- 5.14.1.2 `int backtrackLineSearch (int(*) (const Matrix< double > &x, Matrix< double > &F, const void *data) feval, Matrix< double > &Fkp1, Matrix< double > &xkp1, Matrix< double > &pk, double normFk, BACKTRACK_DATA * backtrack_dat, const void * feval_data)`
- 5.14.1.3 `int bicgstab (int(*) (const Matrix< double > &p, Matrix< double > &Ap, const void *data) matvec, int(*) (const Matrix< double > &r, Matrix< double > &z, const void *data) precon, Matrix< double > &b, BiCGSTAB_DATA * bicg_dat, const void * matvec_data, const void * precon_data)`
- 5.14.1.4 `int cgs (int(*) (const Matrix< double > &p, Matrix< double > &Ap, const void *data) matvec, int(*) (const Matrix< double > &r, Matrix< double > &z, const void *data) precon, Matrix< double > &b, CGS_DATA * cgs_dat, const void * matvec_data, const void * precon_data)`
- 5.14.1.5 `int evalx_ex09 (const Matrix< double > &x, Matrix< double > &G, const void * data)`
- 5.14.1.6 `int fom (int(*) (const Matrix< double > &v, Matrix< double > &w, const void *data) matvec, int(*) (const Matrix< double > &b, Matrix< double > &P, const void *data) precon, Matrix< double > &b, GMRESLP_DATA * gmreslp_dat, const void * matvec_data, const void * precon_data)`
- 5.14.1.7 `int funeval_ex09 (const Matrix< double > &x, Matrix< double > &F, const void * data)`
- 5.14.1.8 `int funeval_ex10 (const Matrix< double > &x, Matrix< double > &F, const void * data)`

- 5.14.1.9 `int gcr (int(*) (const Matrix< double > &x, Matrix< double > &Ax, const void *data) matvec, int(*) (const Matrix< double > &r, Matrix< double > &Mr, const void *data) precon, Matrix< double > &b, GCR_DATA * gcr_dat, const void * matvec_data, const void * precon_data)`
- 5.14.1.10 `int gmresLeftPreconditioned (int(*) (const Matrix< double > &v, Matrix< double > &w, const void *data) matvec, int(*) (const Matrix< double > &b, Matrix< double > &P, const void *data) precon, Matrix< double > &b, GMRESLP_DATA * gmreslp_dat, const void * matvec_data, const void * precon_data)`
- 5.14.1.11 `int gmresPreconditioner (const Matrix< double > &r, Matrix< double > &Mr, const void * data)`
- 5.14.1.12 `int gmresr (int(*) (const Matrix< double > &x, Matrix< double > &Ax, const void *data) matvec, int(*) (const Matrix< double > &r, Matrix< double > &Mr, const void *data) terminal_precon, Matrix< double > &b, GMRESR_DATA * gmresr_dat, const void * matvec_data, const void * term_precon_data)`
- 5.14.1.13 `int gmresRightPreconditioned (int(*) (const Matrix< double > &v, Matrix< double > &w, const void *data) matvec, int(*) (const Matrix< double > &b, Matrix< double > &p, const void *data) precon, Matrix< double > &b, GMRESRP_DATA * gmresrp_dat, const void * matvec_data, const void * precon_data)`
- 5.14.1.14 `int jacvec (const Matrix< double > &v, Matrix< double > &Jv, const void * data)`
- 5.14.1.15 `int LARK_TESTS ()`
- 5.14.1.16 `int matvec_ex01 (const Matrix< double > &v, Matrix< double > &w, const void * data)`
- 5.14.1.17 `int matvec_ex02 (const Matrix< double > &v, Matrix< double > &w, const void * data)`
- 5.14.1.18 `int matvec_ex04 (const Matrix< double > &v, Matrix< double > &w, const void * data)`
- 5.14.1.19 `int matvec_ex15 (const Matrix< double > &v, Matrix< double > &w, const void * data)`
- 5.14.1.20 `int NumericalJacobian (int(*) (const Matrix< double > &x, Matrix< double > &F, const void *user_data) Func, const Matrix< double > &x, Matrix< double > &J, int Nx, int Nf, NUM_JAC_DATA * jac_dat, const void * user_data)`
- 5.14.1.21 `int operatorTranspose (int(*) (const Matrix< double > &v, Matrix< double > &Av, const void *data) matvec, Matrix< double > &r, Matrix< double > &u, OPTRANS_DATA * transpose_dat, const void * matvec_data)`
- 5.14.1.22 `int pcg (int(*) (const Matrix< double > &p, Matrix< double > &Ap, const void *data) matvec, int(*) (const Matrix< double > &r, Matrix< double > &z, const void *data) precon, Matrix< double > &b, PCG_DATA * pcg_dat, const void * matvec_data, const void * precon_data)`
- 5.14.1.23 `int picard (int(*) (const Matrix< double > &x, Matrix< double > &r, const void *data) res, int(*) (const Matrix< double > &x0, Matrix< double > &x, const void *data) evalx, Matrix< double > &x, PICARD_DATA * picard_dat, const void * res_data, const void * evalx_data)`
- 5.14.1.24 `int pjfnk (int(*) (const Matrix< double > &x, Matrix< double > &F, const void *data) res, int(*) (const Matrix< double > &r, Matrix< double > &p, const void *data) precon, Matrix< double > &x, PJFNK_DATA * pjfnk_dat, const void * res_data, const void * precon_data)`
- 5.14.1.25 `int precon_ex01 (const Matrix< double > &b, Matrix< double > &p, const void * data)`
- 5.14.1.26 `int precon_ex04 (const Matrix< double > &b, Matrix< double > &p, const void * data)`
- 5.14.1.27 `int precon_ex10 (const Matrix< double > &r, Matrix< double > &p, const void * data)`
- 5.14.1.28 `int precon_ex15 (const Matrix< double > &w, Matrix< double > &p, const void * data)`

5.14.1.29 `int update_arnoldi_solution (Matrix< double > & x, Matrix< double > & x0, ARNOLDI_DATA * arnoldi_dat)`

5.15 lark.h File Reference

```
#include "macaw.h"
#include <float.h>
```

Classes

- struct [ARNOLDI_DATA](#)
- struct [GMRESLP_DATA](#)
- struct [GMRESRP_DATA](#)
- struct [PCG_DATA](#)
- struct [BiCGSTAB_DATA](#)
- struct [CGS_DATA](#)
- struct [OPTRANS_DATA](#)
- struct [GCR_DATA](#)
- struct [GMRESR_DATA](#)
- struct [PICARD_DATA](#)
- struct [BACKTRACK_DATA](#)
- struct [PJFNK_DATA](#)
- struct [NUM_JAC_DATA](#)
- struct [EX01_DATA](#)
- struct [EX02_DATA](#)
- struct [EX04_DATA](#)
- struct [EX09_DATA](#)
- struct [EX15_DATA](#)

Enumerations

- enum [krylov_method](#) {
[GMRESLP](#), [PCG](#), [BiCGSTAB](#), [CGS](#),
[FOM](#), [GMRESRP](#), [GCR](#), [GMRESR](#) }

Functions

- `int matvec_ex01 (const Matrix< double > &v, Matrix< double > &w, const void *data)`
- `int precon_ex01 (const Matrix< double > &b, Matrix< double > &p, const void *data)`
- `int matvec_ex02 (const Matrix< double > &v, Matrix< double > &w, const void *data)`
- `int matvec_ex04 (const Matrix< double > &v, Matrix< double > &w, const void *data)`
- `int precon_ex04 (const Matrix< double > &b, Matrix< double > &p, const void *data)`
- `int evalx_ex09 (const Matrix< double > &x, Matrix< double > &G, const void *data)`
- `int funeval_ex09 (const Matrix< double > &x, Matrix< double > &F, const void *data)`
- `int funeval_ex10 (const Matrix< double > &x, Matrix< double > &F, const void *data)`
- `int precon_ex10 (const Matrix< double > &r, Matrix< double > &p, const void *data)`
- `int matvec_ex15 (const Matrix< double > &v, Matrix< double > &w, const void *data)`
- `int precon_ex15 (const Matrix< double > &w, Matrix< double > &p, const void *data)`
- `int update_arnoldi_solution (Matrix< double > &x, Matrix< double > &x0, ARNOLDI_DATA *arnoldi_dat)`
- `int arnoldi (int(*matvec)(const Matrix< double > &v, Matrix< double > &w, const void *data),
int(*precon)(const Matrix< double > &b, Matrix< double > &p, const void *data), Matrix< double >
&r0, ARNOLDI_DATA *arnoldi_dat, const void *matvec_data, const void *precon_data)`

- int [gmresLeftPreconditioned](#) (int(*matvec)(const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data), int(*precon)(const [Matrix](#)< double > &b, [Matrix](#)< double > &p, const void *data), [Matrix](#)< double > &b, [GMRESLP_DATA](#) *gmreslp_dat, const void *matvec_data, const void *precon_data)
- int [fom](#) (int(*matvec)(const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data), int(*precon)(const [Matrix](#)< double > &b, [Matrix](#)< double > &p, const void *data), [Matrix](#)< double > &b, [GMRESLP_DATA](#) *gmreslp_dat, const void *matvec_data, const void *precon_data)
- int [gmresRightPreconditioned](#) (int(*matvec)(const [Matrix](#)< double > &v, [Matrix](#)< double > &w, const void *data), int(*precon)(const [Matrix](#)< double > &b, [Matrix](#)< double > &p, const void *data), [Matrix](#)< double > &b, [GMRESRP_DATA](#) *gmresrp_dat, const void *matvec_data, const void *precon_data)
- int [pcg](#) (int(*matvec)(const [Matrix](#)< double > &p, [Matrix](#)< double > &Ap, const void *data), int(*precon)(const [Matrix](#)< double > &r, [Matrix](#)< double > &z, const void *data), [Matrix](#)< double > &b, [PCG_DATA](#) *pcg_dat, const void *matvec_data, const void *precon_data)
- int [bicgstab](#) (int(*matvec)(const [Matrix](#)< double > &p, [Matrix](#)< double > &Ap, const void *data), int(*precon)(const [Matrix](#)< double > &r, [Matrix](#)< double > &z, const void *data), [Matrix](#)< double > &b, [BiCGSTAB_DATA](#) *bicg_dat, const void *matvec_data, const void *precon_data)
- int [cgs](#) (int(*matvec)(const [Matrix](#)< double > &p, [Matrix](#)< double > &Ap, const void *data), int(*precon)(const [Matrix](#)< double > &r, [Matrix](#)< double > &z, const void *data), [Matrix](#)< double > &b, [CGS_DATA](#) *cgs_dat, const void *matvec_data, const void *precon_data)
- int [operatorTranspose](#) (int(*matvec)(const [Matrix](#)< double > &v, [Matrix](#)< double > &Av, const void *data), [Matrix](#)< double > &r, [Matrix](#)< double > &u, [OPTRANS_DATA](#) *transpose_dat, const void *matvec_data)
- int [gcr](#) (int(*matvec)(const [Matrix](#)< double > &x, [Matrix](#)< double > &Ax, const void *data), int(*precon)(const [Matrix](#)< double > &r, [Matrix](#)< double > &Mr, const void *data), [Matrix](#)< double > &b, [GCR_DATA](#) *gcr_dat, const void *matvec_data, const void *precon_data)
- int [gmresPreconditioner](#) (const [Matrix](#)< double > &r, [Matrix](#)< double > &Mr, const void *data)
- int [gmresr](#) (int(*matvec)(const [Matrix](#)< double > &x, [Matrix](#)< double > &Ax, const void *data), int(*terminal_precon)(const [Matrix](#)< double > &r, [Matrix](#)< double > &Mr, const void *data), [Matrix](#)< double > &b, [GMRESR_DATA](#) *gmresr_dat, const void *matvec_data, const void *term_precon_data)
- int [picard](#) (int(*res)(const [Matrix](#)< double > &x, [Matrix](#)< double > &r, const void *data), int(*evalx)(const [Matrix](#)< double > &x0, [Matrix](#)< double > &x, const void *data), [Matrix](#)< double > &x, [PICARD_DATA](#) *picard_dat, const void *res_data, const void *evalx_data)
- int [jacvec](#) (const [Matrix](#)< double > &v, [Matrix](#)< double > &Jv, const void *data)
- int [backtrackLineSearch](#) (int(*feval)(const [Matrix](#)< double > &x, [Matrix](#)< double > &F, const void *data), [Matrix](#)< double > &Fkp1, [Matrix](#)< double > &xkp1, [Matrix](#)< double > &pk, double normFk, [BACKTRACK_DATA](#) *backtrack_dat, const void *feval_data)
- int [pjfnk](#) (int(*res)(const [Matrix](#)< double > &x, [Matrix](#)< double > &F, const void *data), int(*precon)(const [Matrix](#)< double > &r, [Matrix](#)< double > &p, const void *data), [Matrix](#)< double > &x, [PJFNK_DATA](#) *pjfnk_dat, const void *res_data, const void *precon_data)
- int [NumericalJacobian](#) (int(*Func)(const [Matrix](#)< double > &x, [Matrix](#)< double > &F, const void *user_data), const [Matrix](#)< double > &x, [Matrix](#)< double > &J, int Nx, int Nf, [NUM_JAC_DATA](#) *jac_dat, const void *user_data)
- int [LARK_TESTS](#) ()

5.15.1 Enumeration Type Documentation

5.15.1.1 enum krylov_method

Enumerator

GMRESLP
PCG
BiCGSTAB
CGS
FOM
GMRESRP
GCR
GMRESR

5.15.2 Function Documentation

- 5.15.2.1 `int arnoldi (int(*) (const Matrix< double > &v, Matrix< double > &w, const void *data) matvec, int(*) (const Matrix< double > &b, Matrix< double > &p, const void *data) precon, Matrix< double > &r0, ARNOLDI_DATA * arnoldi_dat, const void * matvec_data, const void * precon_data)`
- 5.15.2.2 `int backtrackLineSearch (int(*) (const Matrix< double > &x, Matrix< double > &F, const void *data) feval, Matrix< double > &Fkp1, Matrix< double > &xkp1, Matrix< double > &pk, double normFk, BACKTRACK_DATA * backtrack_dat, const void * feval_data)`
- 5.15.2.3 `int bicgstab (int(*) (const Matrix< double > &p, Matrix< double > &Ap, const void *data) matvec, int(*) (const Matrix< double > &r, Matrix< double > &z, const void *data) precon, Matrix< double > &b, BiCGSTAB_DATA * bicgstab_dat, const void * matvec_data, const void * precon_data)`
- 5.15.2.4 `int cgs (int(*) (const Matrix< double > &p, Matrix< double > &Ap, const void *data) matvec, int(*) (const Matrix< double > &r, Matrix< double > &z, const void *data) precon, Matrix< double > &b, CGS_DATA * cgs_dat, const void * matvec_data, const void * precon_data)`
- 5.15.2.5 `int evalx_ex09 (const Matrix< double > &x, Matrix< double > &G, const void * data)`
- 5.15.2.6 `int fom (int(*) (const Matrix< double > &v, Matrix< double > &w, const void *data) matvec, int(*) (const Matrix< double > &b, Matrix< double > &p, const void *data) precon, Matrix< double > &b, GMRESLP_DATA * gmreslp_dat, const void * matvec_data, const void * precon_data)`
- 5.15.2.7 `int funeval_ex09 (const Matrix< double > &x, Matrix< double > &F, const void * data)`
- 5.15.2.8 `int funeval_ex10 (const Matrix< double > &x, Matrix< double > &F, const void * data)`
- 5.15.2.9 `int gcr (int(*) (const Matrix< double > &x, Matrix< double > &Ax, const void *data) matvec, int(*) (const Matrix< double > &r, Matrix< double > &Mr, const void *data) precon, Matrix< double > &b, GCR_DATA * gcr_dat, const void * matvec_data, const void * precon_data)`
- 5.15.2.10 `int gmresLeftPreconditioned (int(*) (const Matrix< double > &v, Matrix< double > &w, const void *data) matvec, int(*) (const Matrix< double > &b, Matrix< double > &p, const void *data) precon, Matrix< double > &b, GMRESLP_DATA * gmreslp_dat, const void * matvec_data, const void * precon_data)`
- 5.15.2.11 `int gmresPreconditioner (const Matrix< double > &r, Matrix< double > &Mr, const void * data)`
- 5.15.2.12 `int gmresr (int(*) (const Matrix< double > &x, Matrix< double > &Ax, const void *data) matvec, int(*) (const Matrix< double > &r, Matrix< double > &Mr, const void *data) terminal_precon, Matrix< double > &b, GMRESR_DATA * gmresr_dat, const void * matvec_data, const void * term_precon_data)`
- 5.15.2.13 `int gmresRightPreconditioned (int(*) (const Matrix< double > &v, Matrix< double > &w, const void *data) matvec, int(*) (const Matrix< double > &b, Matrix< double > &p, const void *data) precon, Matrix< double > &b, GMRESRP_DATA * gmresrp_dat, const void * matvec_data, const void * precon_data)`
- 5.15.2.14 `int jacvec (const Matrix< double > &v, Matrix< double > &Jv, const void * data)`
- 5.15.2.15 `int LARK_TESTS ()`
- 5.15.2.16 `int matvec_ex01 (const Matrix< double > &v, Matrix< double > &w, const void * data)`
- 5.15.2.17 `int matvec_ex02 (const Matrix< double > &v, Matrix< double > &w, const void * data)`
- 5.15.2.18 `int matvec_ex04 (const Matrix< double > &v, Matrix< double > &w, const void * data)`

- 5.15.2.19 `int matvec_ex15 (const Matrix< double > & v, Matrix< double > & w, const void * data)`
- 5.15.2.20 `int NumericalJacobian (int(*) (const Matrix< double > &x, Matrix< double > &F, const void *user_data) Func, const Matrix< double > & x, Matrix< double > & J, int Nx, int Nf, NUM_JAC_DATA * jac_dat, const void * user_data)`
- 5.15.2.21 `int operatorTranspose (int(*) (const Matrix< double > &v, Matrix< double > &Av, const void *data) matvec, Matrix< double > & r, Matrix< double > & u, OPTRANS_DATA * transpose_dat, const void * matvec.data)`
- 5.15.2.22 `int pcg (int(*) (const Matrix< double > &p, Matrix< double > &Ap, const void *data) matvec, int(*) (const Matrix< double > &r, Matrix< double > &z, const void *data) precon, Matrix< double > & b, PCG_DATA * pcg_dat, const void * matvec.data, const void * precon.data)`
- 5.15.2.23 `int picard (int(*) (const Matrix< double > &x, Matrix< double > &r, const void *data) res, int(*) (const Matrix< double > &x0, Matrix< double > &x, const void *data) evalx, Matrix< double > & x, PICARD_DATA * picard_dat, const void * res.data, const void * evalx.data)`
- 5.15.2.24 `int pjfnk (int(*) (const Matrix< double > &x, Matrix< double > &F, const void *data) res, int(*) (const Matrix< double > &r, Matrix< double > &p, const void *data) precon, Matrix< double > & x, PJFNK_DATA * pjfnk_dat, const void * res.data, const void * precon.data)`
- 5.15.2.25 `int precon_ex01 (const Matrix< double > & b, Matrix< double > & p, const void * data)`
- 5.15.2.26 `int precon_ex04 (const Matrix< double > & b, Matrix< double > & p, const void * data)`
- 5.15.2.27 `int precon_ex10 (const Matrix< double > & r, Matrix< double > & p, const void * data)`
- 5.15.2.28 `int precon_ex15 (const Matrix< double > & w, Matrix< double > & p, const void * data)`
- 5.15.2.29 `int update_arnoldi_solution (Matrix< double > & x, Matrix< double > & x0, ARNOLDI_DATA * arnoldi_dat)`

5.16 macaw.cpp File Reference

```
#include "macaw.h"
```

Functions

- `int MACAW_TESTS ()`

5.16.1 Function Documentation

5.16.1.1 `int MACAW_TESTS ()`

5.17 macaw.h File Reference

```
#include <stdio.h>
```

```
#include <math.h>
#include <iostream>
#include <fstream>
#include <stdlib.h>
#include <vector>
#include <time.h>
#include <float.h>
#include <string>
#include <exception>
#include "error.h"
```

Classes

- class [Matrix< T >](#)

Macros

- `#define M_PI 3.14159265358979323846264338327950288 /* pi */`

Functions

- int [MACAW_TESTS](#) ()

5.17.1 Macro Definition Documentation

5.17.1.1 `#define M_PI 3.14159265358979323846264338327950288 /* pi */`

5.17.2 Function Documentation

5.17.2.1 int [MACAW_TESTS](#) ()

5.18 magpie.cpp File Reference

```
#include "magpie.h"
```

Functions

- double [qo](#) (double po, const void *data, int i)
- double [dq_dp](#) (double p, const void *data, int i)
- double [q_p](#) (double p, const void *data, int i)
- double [PI](#) (double po, const void *data, int i)
- double [eMax](#) (const void *data, int i)
- double [Qst](#) (double po, const void *data, int i)
- double [Inact_mSPD](#) (const double *par, const void *data, int i, volatile double [PI](#))
- double [grad_mSPD](#) (const double *par, const void *data, int i)
- double [qT](#) (const double *par, const void *data)
- void [initialGuess_mSPD](#) (double *par, const void *data)
- void [eval_po_PI](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- void [eval_po_qo](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- void [eval_po](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)

- void [eval_eta](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- void [eval_GPAST](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- int [MAGPIE](#) (const void *data)
- int [MAGPIE_SCENARIOS](#) (const char *inputFileName, const char *sceneFileName)

5.18.1 Function Documentation

5.18.1.1 double dq_dp (double *p*, const void * *data*, int *i*)

5.18.1.2 double eMax (const void * *data*, int *i*)

5.18.1.3 void eval_eta (const double * *par*, int *m_dat*, const void * *data*, double * *fvec*, int * *info*)

5.18.1.4 void eval_GPAST (const double * *par*, int *m_dat*, const void * *data*, double * *fvec*, int * *info*)

5.18.1.5 void eval_po (const double * *par*, int *m_dat*, const void * *data*, double * *fvec*, int * *info*)

5.18.1.6 void eval_po_PI (const double * *par*, int *m_dat*, const void * *data*, double * *fvec*, int * *info*)

5.18.1.7 void eval_po_qo (const double * *par*, int *m_dat*, const void * *data*, double * *fvec*, int * *info*)

5.18.1.8 double grad_mSPD (const double * *par*, const void * *data*, int *i*)

5.18.1.9 void initialGuess_mSPD (double * *par*, const void * *data*)

5.18.1.10 double lnact_mSPD (const double * *par*, const void * *data*, int *i*, volatile double *PI*)

5.18.1.11 int MAGPIE (const void * *data*)

5.18.1.12 int MAGPIE_SCENARIOS (const char * *inputFileName*, const char * *sceneFileName*)

5.18.1.13 double PI (double *po*, const void * *data*, int *i*)

5.18.1.14 double q_p (double *p*, const void * *data*, int *i*)

5.18.1.15 double qo (double *po*, const void * *data*, int *i*)

5.18.1.16 double Qst (double *po*, const void * *data*, int *i*)

5.18.1.17 double qT (const double * *par*, const void * *data*)

5.19 magpie.h File Reference

```
#include "lmcurve.h"
#include <stdio.h>
#include <math.h>
#include <iostream>
#include <fstream>
#include <stdlib.h>
#include <vector>
#include <time.h>
#include <float.h>
#include <string>
#include "error.h"
```

Classes

- struct [GSTA_DATA](#)
- struct [mSPD_DATA](#)
- struct [GPAST_DATA](#)
- struct [SYSTEM_DATA](#)
- struct [MAGPIE_DATA](#)

Macros

- `#define DBL_EPSILON 2.2204460492503131e-016`
- `#define Z 10.0`
- `#define A 3.13E+09`
- `#define V 18.92`
- `#define Po 100.0`
- `#define R 8.3144621`
- `#define Na 6.0221413E+23`
- `#define kB 1.3806488E-23`
- `#define shapeFactor(v_i) (((Z - 2) * v_i) / (Z * V)) + (2 / Z)`
- `#define lnKo(H, S, T) -(H / (R * T)) + (S / R)`
- `#define He(qm, K1, m) (qm * K1) / (m * Po)`

Functions

- double [qo](#) (double po, const void *data, int i)
- double [dq_dp](#) (double p, const void *data, int i)
- double [q_p](#) (double p, const void *data, int i)
- double [PI](#) (double po, const void *data, int i)
- double [Qst](#) (double po, const void *data, int i)
- double [eMax](#) (const void *data, int i)
- double [lnact_mSPD](#) (const double *par, const void *data, int i, volatile double [PI](#))
- double [grad_mSPD](#) (const double *par, const void *data, int i)
- double [qT](#) (const double *par, const void *data)
- void [initialGuess_mSPD](#) (double *par, const void *data)
- void [eval_po_PI](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- void [eval_po_qo](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- void [eval_po](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- void [eval_eta](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- void [eval_GPAST](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- int [MAGPIE](#) (const void *data)
- int [MAGPIE_SCENARIOS](#) (const char *inputFileName, const char *sceneFileName)

5.19.1 Macro Definition Documentation

5.19.1.1 `#define A 3.13E+09`

5.19.1.2 `#define DBL_EPSILON 2.2204460492503131e-016`

5.19.1.3 `#define He(qm, K1, m) (qm * K1) / (m * Po)`

5.19.1.4 `#define kB 1.3806488E-23`

5.19.1.5 `#define lnKo(H, S, T) -(H / (R * T)) + (S / R)`

5.19.1.6 `#define Na 6.0221413E+23`

5.19.1.7 `#define Po 100.0`

5.19.1.8 `#define R 8.3144621`

5.19.1.9 `#define shapeFactor(v_i) (((Z - 2) * v_i) / (Z * V)) + (2 / Z)`

5.19.1.10 `#define V 18.92`

5.19.1.11 `#define Z 10.0`

5.19.2 Function Documentation

5.19.2.1 `double dq_dp (double p, const void * data, int i)`

5.19.2.2 `double eMax (const void * data, int i)`

5.19.2.3 `void eval_eta (const double * par, int m_dat, const void * data, double * fvec, int * info)`

5.19.2.4 `void eval_GPAST (const double * par, int m_dat, const void * data, double * fvec, int * info)`

5.19.2.5 `void eval_po (const double * par, int m_dat, const void * data, double * fvec, int * info)`

5.19.2.6 `void eval_po_PI (const double * par, int m_dat, const void * data, double * fvec, int * info)`

5.19.2.7 `void eval_po_qo (const double * par, int m_dat, const void * data, double * fvec, int * info)`

5.19.2.8 `double grad_mSPD (const double * par, const void * data, int i)`

5.19.2.9 `void initialGuess_mSPD (double * par, const void * data)`

5.19.2.10 `double lnact_mSPD (const double * par, const void * data, int i, volatile double PI)`

5.19.2.11 `int MAGPIE (const void * data)`

5.19.2.12 `int MAGPIE_SCENARIOS (const char * inputFileName, const char * sceneFileName)`

5.19.2.13 `double PI (double po, const void * data, int i)`

5.19.2.14 `double q_p (double p, const void * data, int i)`

5.19.2.15 `double qo (double po, const void * data, int i)`

5.19.2.16 `double Qst (double po, const void * data, int i)`

5.19.2.17 `double qT (const double * par, const void * data)`

5.20 main.cpp File Reference

```
#include "ui.h"
```

Functions

- int [main](#) (int argc, const char *argv[])

5.20.1 Function Documentation

5.20.1.1 int main (int argc, const char * argv[])

5.21 mola.cpp File Reference

```
#include "mola.h"
```

Functions

- int [MOLA_TESTS](#) ()

5.21.1 Function Documentation

5.21.1.1 int MOLA_TESTS ()

5.22 mola.h File Reference

```
#include <ctype.h>
#include "eel.h"
```

Classes

- class [Molecule](#)

Functions

- int [MOLA_TESTS](#) ()

5.22.1 Function Documentation

5.22.1.1 int MOLA_TESTS ()

5.23 monkfish.cpp File Reference

```
#include "monkfish.h"
```

Functions

- double [default_porosity](#) (int i, int l, const void *user_data)
- double [default_density](#) (int i, int l, const void *user_data)
- double [default_interparticle_diffusion](#) (int i, int l, const void *user_data)

- double [default_monk_adsorption](#) (int i, int l, const void *user_data)
- double [default_monk_equilibrium](#) (int i, int l, const void *user_data)
- double [default_monkfish_retardation](#) (int i, int l, const void *user_data)
- double [default_exterior_concentration](#) (int i, const void *user_data)
- double [default_film_transfer](#) (int i, const void *user_data)
- int [MONKFISH_TESTS](#) ()

5.23.1 Function Documentation

5.23.1.1 double [default_density](#) (int i, int l, const void * *user_data*)

5.23.1.2 double [default_exterior_concentration](#) (int i, const void * *user_data*)

5.23.1.3 double [default_film_transfer](#) (int i, const void * *user_data*)

5.23.1.4 double [default_interparticle_diffusion](#) (int i, int l, const void * *user_data*)

5.23.1.5 double [default_monk_adsorption](#) (int i, int l, const void * *user_data*)

5.23.1.6 double [default_monk_equilibrium](#) (int i, int l, const void * *user_data*)

5.23.1.7 double [default_monkfish_retardation](#) (int i, int l, const void * *user_data*)

5.23.1.8 double [default_porosity](#) (int i, int l, const void * *user_data*)

5.23.1.9 int [MONKFISH_TESTS](#) ()

5.24 monkfish.h File Reference

```
#include "dogfish.h"
```

Classes

- struct [MONKFISH_PARAM](#)
- struct [MONKFISH_DATA](#)

Functions

- double [default_porosity](#) (int i, int l, const void *user_data)
- double [default_density](#) (int i, int l, const void *user_data)
- double [default_interparticle_diffusion](#) (int i, int l, const void *user_data)
- double [default_monk_adsorption](#) (int i, int l, const void *user_data)
- double [default_monk_equilibrium](#) (int i, int l, const void *user_data)
- double [default_monkfish_retardation](#) (int i, int l, const void *user_data)
- double [default_exterior_concentration](#) (int i, const void *user_data)
- double [default_film_transfer](#) (int i, const void *user_data)
- int [setup_MONKFISH_DATA](#) (FILE *file, double(*eval_porosity)(int i, int l, const void *user_data), double(*eval_density)(int i, int l, const void *user_data), double(*eval_ext_diff)(int i, int l, const void *user_data), double(*eval_adsorb)(int i, int l, const void *user_data), double(*eval_retard)(int i, int l, const void *user_data), double(*eval_ext_conc)(int i, const void *user_data), double(*eval_ext_film)(int i, const void *user_data), double(*dog_diffusion)(int i, int l, const void *user_data), double(*dog_ext_film)(int i, const void *user_data), double(*dog_surf_conc)(int i, const void *user_data), const void *user_data, [MONKFISH_DATA](#) *monk_dat)
- int [MONKFISH_TESTS](#) ()

5.24.1 Function Documentation

5.24.1.1 `double default_density (int i, int l, const void * user_data)`

5.24.1.2 `double default_exterior_concentration (int i, const void * user_data)`

5.24.1.3 `double default_film_transfer (int i, const void * user_data)`

5.24.1.4 `double default_interparticle_diffusion (int i, int l, const void * user_data)`

5.24.1.5 `double default_monk_adsorption (int i, int l, const void * user_data)`

5.24.1.6 `double default_monk_equilibrium (int i, int l, const void * user_data)`

5.24.1.7 `double default_monkfish_retardation (int i, int l, const void * user_data)`

5.24.1.8 `double default_porosity (int i, int l, const void * user_data)`

5.24.1.9 `int MONKFISH_TESTS ()`

5.24.1.10 `int setup_MONKFISH_DATA (FILE * file, double (*)(int i, int l, const void * user_data) eval_porosity, double (*)(int i, int l, const void * user_data) eval_density, double (*)(int i, int l, const void * user_data) eval_ext_diff, double (*)(int i, int l, const void * user_data) eval_adsorb, double (*)(int i, int l, const void * user_data) eval_retard, double (*)(int i, const void * user_data) eval_ext_conc, double (*)(int i, const void * user_data) eval_ext_film, double (*)(int i, int l, const void * user_data) dog_diffusion, double (*)(int i, const void * user_data) dog_ext_film, double (*)(int i, const void * user_data) dog_surf_conc, const void * user_data, MONKFISH_DATA * monk_dat)`

5.25 sandbox.cpp File Reference

```
#include "sandbox.h"
```

Functions

- int [Speciation_Test01_Function](#) (const [Matrix](#)< double > &*x*, [Matrix](#)< double > &*F*, const void * *res_data*)
- int [Speciation_Test01_Jacobian](#) (const [Matrix](#)< double > &*x*, [Matrix](#)< double > &*J*, const void * *precon_data*)
- int [Speciation_Test01_Guess](#) (const void * *user_data*)
- int [Speciation_Test01_MatVec](#) (const [Matrix](#)< double > &*x*, [Matrix](#)< double > &*Ax*, const void * *matvec_data*)
- int [RUN_SANDBOX](#) ()

5.25.1 Function Documentation

5.25.1.1 `int RUN_SANDBOX ()`

5.25.1.2 `int Speciation_Test01_Function (const Matrix< double > & x, Matrix< double > & F, const void * res_data)`

5.25.1.3 `int Speciation_Test01_Guess (const void * user_data)`

5.25.1.4 `int Speciation_Test01_Jacobian (const Matrix< double > & x, Matrix< double > & J, const void * precon_data)`

5.25.1.5 `int Speciation_Test01_MatVec (const Matrix< double > & x, Matrix< double > & Ax, const void * matvec_data)`

5.26 sandbox.h File Reference

```
#include "flock.h"
#include "school.h"
```

Classes

- struct [Speciation_Test01_Data](#)

Functions

- int [Speciation_Test01_Function](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &F, const void *res_data)
- int [Speciation_Test01_Jacobian](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &J, const void *precon_data)
- int [Speciation_Test01_Guess](#) (const void *user_data)
- int [Speciation_Test01_MatVec](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &Ax, const void *matvec_data)
- int [RUN_SANDBOX](#) ()

5.26.1 Function Documentation

5.26.1.1 int [RUN_SANDBOX](#) ()

5.26.1.2 int [Speciation_Test01_Function](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &F, const void * *res_data*)

5.26.1.3 int [Speciation_Test01_Guess](#) (const void * *user_data*)

5.26.1.4 int [Speciation_Test01_Jacobian](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &J, const void * *precon_data*)

5.26.1.5 int [Speciation_Test01_MatVec](#) (const [Matrix](#)< double > &x, [Matrix](#)< double > &Ax, const void * *matvec_data*)

5.27 school.h File Reference

```
#include "eel.h"
#include "mola.h"
#include "shark.h"
#include "dogfish.h"
#include "monkfish.h"
#include "yaml_wrapper.h"
```

5.28 scopsowl.cpp File Reference

```
#include "scopsowl.h"
```

Functions

- void [print2file_species_header](#) (FILE *Output, [SCOPSOWL_DATA](#) *owl_dat, int i)
- void [print2file_SCOPSOWL_time_header](#) (FILE *Output, [SCOPSOWL_DATA](#) *owl_dat, int i)
- void [print2file_SCOPSOWL_header](#) ([SCOPSOWL_DATA](#) *owl_dat)

- void [print2file_SCOPSOWL_result_old](#) ([SCOPSOWL_DATA](#) *owl_dat)
- void [print2file_SCOPSOWL_result_new](#) ([SCOPSOWL_DATA](#) *owl_dat)
- double [default_adsorption](#) (int i, int l, const void *user_data)
- double [default_retardation](#) (int i, int l, const void *user_data)
- double [default_pore_diffusion](#) (int i, int l, const void *user_data)
- double [default_surf_diffusion](#) (int i, int l, const void *user_data)
- double [default_effective_diffusion](#) (int i, int l, const void *user_data)
- double [const_pore_diffusion](#) (int i, int l, const void *user_data)
- double [default_filmMassTransfer](#) (int i, const void *user_data)
- double [const_filmMassTransfer](#) (int i, const void *user_data)
- int [setup_SCOPSOWL_DATA](#) (FILE *file, double(*eval_sorption)(int i, int l, const void *user_data), double(*eval_retardation)(int i, int l, const void *user_data), double(*eval_pore_diff)(int i, int l, const void *user_data), double(*eval_filmMT)(int i, const void *user_data), double(*eval_surface_diff)(int i, int l, const void *user_data), const void *user_data, [MIXED_GAS](#) *gas_data, [SCOPSOWL_DATA](#) *owl_data)
- int [SCOPSOWL_Executioner](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [set_SCOPSOWL_ICs](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [set_SCOPSOWL_timestep](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [SCOPSOWL_preprocesses](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [set_SCOPSOWL_params](#) (const void *user_data)
- int [SCOPSOWL_postprocesses](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [SCOPSOWL_reset](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [SCOPSOWL](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [LARGE_CYCLE_TEST01](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [SMALL_CYCLE_TEST02](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [CURVE_TEST03](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [CURVE_TEST04](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [CURVE_TEST05](#) ([SCOPSOWL_DATA](#) *owl_dat)
- int [SCOPSOWL_SCENARIOS](#) (const char *scene, const char *sorbent, const char *comp, const char *sorbate)
- int [SCOPSOWL_TESTS](#) ()

5.28.1 Function Documentation

5.28.1.1 double [const_filmMassTransfer](#) (int i, const void * user_data)

5.28.1.2 double [const_pore_diffusion](#) (int i, int l, const void * user_data)

5.28.1.3 int [CURVE_TEST03](#) ([SCOPSOWL_DATA](#) * owl_dat)

5.28.1.4 int [CURVE_TEST04](#) ([SCOPSOWL_DATA](#) * owl_dat)

5.28.1.5 int [CURVE_TEST05](#) ([SCOPSOWL_DATA](#) * owl_dat)

5.28.1.6 double [default_adsorption](#) (int i, int l, const void * user_data)

5.28.1.7 double [default_effective_diffusion](#) (int i, int l, const void * user_data)

5.28.1.8 double [default_filmMassTransfer](#) (int i, const void * user_data)

5.28.1.9 double [default_pore_diffusion](#) (int i, int l, const void * user_data)

5.28.1.10 double [default_retardation](#) (int i, int l, const void * user_data)

5.28.1.11 double [default_surf_diffusion](#) (int i, int l, const void * user_data)

- 5.28.1.12 int LARGE_CYCLE_TEST01 (SCOPSOWL_DATA * owl_dat)
- 5.28.1.13 void print2file_SCOPSOWL_header (SCOPSOWL_DATA * owl_dat)
- 5.28.1.14 void print2file_SCOPSOWL_result_new (SCOPSOWL_DATA * owl_dat)
- 5.28.1.15 void print2file_SCOPSOWL_result_old (SCOPSOWL_DATA * owl_dat)
- 5.28.1.16 void print2file_SCOPSOWL_time_header (FILE * Output, SCOPSOWL_DATA * owl_dat, int i)
- 5.28.1.17 void print2file_species_header (FILE * Output, SCOPSOWL_DATA * owl_dat, int i)
- 5.28.1.18 int SCOPSOWL (SCOPSOWL_DATA * owl_dat)
- 5.28.1.19 int SCOPSOWL_Executioner (SCOPSOWL_DATA * owl_dat)
- 5.28.1.20 int SCOPSOWL_postprocesses (SCOPSOWL_DATA * owl_dat)
- 5.28.1.21 int SCOPSOWL_preprocesses (SCOPSOWL_DATA * owl_dat)
- 5.28.1.22 int SCOPSOWL_reset (SCOPSOWL_DATA * owl_dat)
- 5.28.1.23 int SCOPSOWL_SCENARIOS (const char * scene, const char * sorbent, const char * comp, const char * sorbate)
- 5.28.1.24 int SCOPSOWL_TESTS ()
- 5.28.1.25 int set_SCOPSOWL_ICs (SCOPSOWL_DATA * owl_dat)
- 5.28.1.26 int set_SCOPSOWL_params (const void * user_data)
- 5.28.1.27 int set_SCOPSOWL_timestep (SCOPSOWL_DATA * owl_dat)
- 5.28.1.28 int setup_SCOPSOWL_DATA (FILE * file, double(*) (int i, int l, const void * user_data) eval_sorption, double(*) (int i, int l, const void * user_data) eval_retardation, double(*) (int i, int l, const void * user_data) eval_pore_diff, double(*) (int i, const void * user_data) eval_filmMT, double(*) (int i, int l, const void * user_data) eval_surface_diff, const void * user_data, MIXED_GAS * gas_data, SCOPSOWL_DATA * owl_data)
- 5.28.1.29 int SMALL_CYCLE_TEST02 (SCOPSOWL_DATA * owl_dat)

5.29 scopsowl.h File Reference

```
#include "egret.h"
#include "skua.h"
```

Classes

- struct [SCOPSOWL_PARAM_DATA](#)
- struct [SCOPSOWL_DATA](#)

Macros

- #define [SCOPSOWL_HPP_](#)
- #define [Dp](#)(Dm, ep) (ep*ep*Dm)

- `#define Dk(rp, T, MW) (9700.0*rp*pow((T/MW),0.5))`
- `#define avgDp(Dp, Dk) (pow(((1/Dp)+(1/Dk)),-1.0))`

Functions

- void `print2file_species_header` (FILE *Output, SCOPSOWL_DATA *owl_dat, int i)
- void `print2file_SCOPSOWL_time_header` (FILE *Output, SCOPSOWL_DATA *owl_dat, int i)
- void `print2file_SCOPSOWL_header` (SCOPSOWL_DATA *owl_dat)
- void `print2file_SCOPSOWL_result_old` (SCOPSOWL_DATA *owl_dat)
- void `print2file_SCOPSOWL_result_new` (SCOPSOWL_DATA *owl_dat)
- double `default_adsorption` (int i, int l, const void *user_data)
- double `default_retardation` (int i, int l, const void *user_data)
- double `default_pore_diffusion` (int i, int l, const void *user_data)
- double `default_surf_diffusion` (int i, int l, const void *user_data)
- double `default_effective_diffusion` (int i, int l, const void *user_data)
- double `const_pore_diffusion` (int i, int l, const void *user_data)
- double `default_filmMassTransfer` (int i, const void *user_data)
- double `const_filmMassTransfer` (int i, const void *user_data)
- int `setup_SCOPSOWL_DATA` (FILE *file, double(*eval_sorption)(int i, int l, const void *user_data), double(*eval_retardation)(int i, int l, const void *user_data), double(*eval_pore_diff)(int i, int l, const void *user_data), double(*eval_filmMT)(int i, const void *user_data), double(*eval_surface_diff)(int i, int l, const void *user_data), const void *user_data, MIXED_GAS *gas_data, SCOPSOWL_DATA *owl_data)
- int `SCOPSOWL_Executioner` (SCOPSOWL_DATA *owl_dat)
- int `set_SCOPSOWL_ICs` (SCOPSOWL_DATA *owl_dat)
- int `set_SCOPSOWL_timestep` (SCOPSOWL_DATA *owl_dat)
- int `SCOPSOWL_preprocesses` (SCOPSOWL_DATA *owl_dat)
- int `set_SCOPSOWL_params` (const void *user_data)
- int `SCOPSOWL_postprocesses` (SCOPSOWL_DATA *owl_dat)
- int `SCOPSOWL_reset` (SCOPSOWL_DATA *owl_dat)
- int `SCOPSOWL` (SCOPSOWL_DATA *owl_dat)
- int `LARGE_CYCLE_TEST01` (SCOPSOWL_DATA *owl_dat)
- int `SMALL_CYCLE_TEST02` (SCOPSOWL_DATA *owl_dat)
- int `CURVE_TEST03` (SCOPSOWL_DATA *owl_dat)
- int `CURVE_TEST04` (SCOPSOWL_DATA *owl_dat)
- int `CURVE_TEST05` (SCOPSOWL_DATA *owl_dat)
- int `SCOPSOWL_SCENARIOS` (const char *scene, const char *sorbent, const char *comp, const char *sorbate)
- int `SCOPSOWL_TESTS` ()

5.29.1 Macro Definition Documentation

5.29.1.1 `#define avgDp(Dp, Dk) (pow(((1/Dp)+(1/Dk)),-1.0))`

5.29.1.2 `#define Dk(rp, T, MW) (9700.0*rp*pow((T/MW),0.5))`

5.29.1.3 `#define Dp(Dm, ep) (ep*ep*Dm)`

5.29.1.4 `#define SCOPSOWL_HPP_`

5.29.2 Function Documentation

5.29.2.1 `double const_filmMassTransfer (int i, const void * user_data)`

- 5.29.2.2 double const_pore_diffusion (int *i*, int *l*, const void * *user_data*)
- 5.29.2.3 int CURVE_TEST03 (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.4 int CURVE_TEST04 (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.5 int CURVE_TEST05 (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.6 double default_adsorption (int *i*, int *l*, const void * *user_data*)
- 5.29.2.7 double default_effective_diffusion (int *i*, int *l*, const void * *user_data*)
- 5.29.2.8 double default_filmMassTransfer (int *i*, const void * *user_data*)
- 5.29.2.9 double default_pore_diffusion (int *i*, int *l*, const void * *user_data*)
- 5.29.2.10 double default_retardation (int *i*, int *l*, const void * *user_data*)
- 5.29.2.11 double default_surf_diffusion (int *i*, int *l*, const void * *user_data*)
- 5.29.2.12 int LARGE_CYCLE_TEST01 (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.13 void print2file_SCOPSOWL_header (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.14 void print2file_SCOPSOWL_result_new (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.15 void print2file_SCOPSOWL_result_old (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.16 void print2file_SCOPSOWL_time_header (FILE * *Output*, SCOPSOWL_DATA * *owl_dat*, int *i*)
- 5.29.2.17 void print2file_species_header (FILE * *Output*, SCOPSOWL_DATA * *owl_dat*, int *i*)
- 5.29.2.18 int SCOPSOWL (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.19 int SCOPSOWL_Executioner (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.20 int SCOPSOWL_postprocesses (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.21 int SCOPSOWL_preprocesses (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.22 int SCOPSOWL_reset (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.23 int SCOPSOWL_SCENARIOS (const char * *scene*, const char * *sorbent*, const char * *comp*, const char * *sorbate*)
- 5.29.2.24 int SCOPSOWL_TESTS ()
- 5.29.2.25 int set_SCOPSOWL_ICs (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.26 int set_SCOPSOWL_params (const void * *user_data*)
- 5.29.2.27 int set_SCOPSOWL_timestep (SCOPSOWL_DATA * *owl_dat*)
- 5.29.2.28 int setup_SCOPSOWL_DATA (FILE * *file*, double(*)(int *i*, int *l*, const void **user_data*) *eval_sorption*, double(*)(int *i*, int *l*, const void **user_data*) *eval_retardation*, double(*)(int *i*, int *l*, const void **user_data*) *eval_pore_diff*, double(*)(int *i*, const void **user_data*) *eval_filmMT*, double(*)(int *i*, int *l*, const void **user_data*) *eval_surface_diff*, const void * *user_data*, MIXED_GAS * *gas_data*, SCOPSOWL_DATA * *owl_data*)

5.29.2.29 int SMALL_CYCLE_TEST02 (SCOPSOWL_DATA * owl_dat)

5.30 scopsowl_opt.cpp File Reference

```
#include "scopsowl_opt.h"
```

Functions

- int [SCOPSOWL_OPT_set_y](#) (SCOPSOWL_OPT_DATA *owl_opt)
- int [initial_guess_SCOPSOWL](#) (SCOPSOWL_OPT_DATA *owl_opt)
- void [eval_SCOPSOWL_Uptake](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- int [SCOPSOWL_OPTIMIZE](#) (const char *scene, const char *sor bent, const char *comp, const char *sor bate, const char *data)

5.30.1 Function Documentation

5.30.1.1 void eval_SCOPSOWL_Uptake (const double * *par*, int *m_dat*, const void * *data*, double * *fvec*, int * *info*)

5.30.1.2 int initial_guess_SCOPSOWL (SCOPSOWL_OPT_DATA * owl_opt)

5.30.1.3 int SCOPSOWL_OPT_set_y (SCOPSOWL_OPT_DATA * owl_opt)

5.30.1.4 int SCOPSOWL_OPTIMIZE (const char * *scene*, const char * *sor bent*, const char * *comp*, const char * *sor bate*, const char * *data*)

5.31 scopsowl_opt.h File Reference

```
#include "scopsowl.h"
```

Classes

- struct [SCOPSOWL_OPT_DATA](#)

Functions

- int [SCOPSOWL_OPT_set_y](#) (SCOPSOWL_OPT_DATA *owl_opt)
- int [initial_guess_SCOPSOWL](#) (SCOPSOWL_OPT_DATA *owl_opt)
- void [eval_SCOPSOWL_Uptake](#) (const double *par, int m_dat, const void *data, double *fvec, int *info)
- int [SCOPSOWL_OPTIMIZE](#) (const char *scene, const char *sor bent, const char *comp, const char *sor bate, const char *data)

5.31.1 Function Documentation

5.31.1.1 void eval_SCOPSOWL_Uptake (const double * *par*, int *m_dat*, const void * *data*, double * *fvec*, int * *info*)

5.31.1.2 int initial_guess_SCOPSOWL (SCOPSOWL_OPT_DATA * owl_opt)

5.31.1.3 int SCOPSOWL_OPT_set_y (SCOPSOWL_OPT_DATA * owl_opt)

5.31.1.4 int SCOPSOWL_OPTIMIZE (const char * *scene*, const char * *sorbent*, const char * *comp*, const char * *sorbate*,
const char * *data*)

5.32 shark.cpp File Reference

```
#include "shark.h"
```

Functions

- void [print2file_shark_info](#) (SHARK_DATA *shark_dat)
- void [print2file_shark_header](#) (SHARK_DATA *shark_dat)
- void [print2file_shark_results_new](#) (SHARK_DATA *shark_dat)
- void [print2file_shark_results_old](#) (SHARK_DATA *shark_dat)
- int [ideal_solution](#) (const Matrix< double > &x, Matrix< double > &F, const void *data)
- int [Davies_equation](#) (const Matrix< double > &x, Matrix< double > &F, const void *data)
- int [DebyeHuckel_equation](#) (const Matrix< double > &x, Matrix< double > &F, const void *data)
- int [DaviesLadshaw_equation](#) (const Matrix< double > &x, Matrix< double > &F, const void *data)
- int [act_choice](#) (const std::string &input)
- bool [linesearch_choice](#) (const std::string &input)
- int [linearsolve_choice](#) (const std::string &input)
- int [Convert2LogConcentration](#) (const Matrix< double > &x, Matrix< double > &logx)
- int [Convert2Concentration](#) (const Matrix< double > &logx, Matrix< double > &x)
- int [read_scenario](#) (SHARK_DATA *shark_dat)
- int [read_options](#) (SHARK_DATA *shark_dat)
- int [read_species](#) (SHARK_DATA *shark_dat)
- int [read_massbalance](#) (SHARK_DATA *shark_dat)
- int [read_equilrxn](#) (SHARK_DATA *shark_dat)
- int [read_unsteadyrxn](#) (SHARK_DATA *shark_dat)
- int [setup_SHARK_DATA](#) (FILE *file, int(*residual)(const Matrix< double > &x, Matrix< double > &res, const void *data), int(*activity)(const Matrix< double > &x, Matrix< double > &gama, const void *data), int(*precond)(const Matrix< double > &r, Matrix< double > &p, const void *data), SHARK_DATA *dat, const void *activity_data, const void *residual_data, const void *precon_data, const void *other_data)
- int [shark_add_customResidual](#) (int i, double(*other_res)(const Matrix< double > &x, SHARK_DATA *shark_dat, const void *other_data), SHARK_DATA *shark_dat)
- int [shark_parameter_check](#) (SHARK_DATA *shark_dat)
- int [shark_energy_calculations](#) (SHARK_DATA *shark_dat)
- int [shark_temperature_calculations](#) (SHARK_DATA *shark_dat)
- int [shark_pH_finder](#) (SHARK_DATA *shark_dat)
- int [shark_guess](#) (SHARK_DATA *shark_dat)
- int [shark_initial_conditions](#) (SHARK_DATA *shark_dat)
- int [shark_executioner](#) (SHARK_DATA *shark_dat)
- int [shark_timestep_const](#) (SHARK_DATA *shark_dat)
- int [shark_timestep_adapt](#) (SHARK_DATA *shark_dat)
- int [shark_preprocesses](#) (SHARK_DATA *shark_dat)
- int [shark_solver](#) (SHARK_DATA *shark_dat)
- int [shark_postprocesses](#) (SHARK_DATA *shark_dat)
- int [shark_reset](#) (SHARK_DATA *shark_dat)
- int [shark_residual](#) (const Matrix< double > &x, Matrix< double > &F, const void *data)
- int [SHARK](#) (SHARK_DATA *shark_dat)
- int [SHARK_SCENARIO](#) (const char *yaml_input)
- int [SHARK_TESTS](#) ()

5.32.1 Function Documentation

- 5.32.1.1 `int act_choice (const std::string & input)`
- 5.32.1.2 `int Convert2Concentration (const Matrix< double > & logx, Matrix< double > & x)`
- 5.32.1.3 `int Convert2LogConcentration (const Matrix< double > & x, Matrix< double > & logx)`
- 5.32.1.4 `int Davies_equation (const Matrix< double > & x, Matrix< double > & F, const void * data)`
- 5.32.1.5 `int DaviesLadshaw_equation (const Matrix< double > & x, Matrix< double > & F, const void * data)`
- 5.32.1.6 `int DebyeHuckel_equation (const Matrix< double > & x, Matrix< double > & F, const void * data)`
- 5.32.1.7 `int ideal_solution (const Matrix< double > & x, Matrix< double > & F, const void * data)`
- 5.32.1.8 `int linearsolve_choice (const std::string & input)`
- 5.32.1.9 `bool linesearch_choice (const std::string & input)`
- 5.32.1.10 `void print2file_shark_header (SHARK_DATA * shark_dat)`
- 5.32.1.11 `void print2file_shark_info (SHARK_DATA * shark_dat)`
- 5.32.1.12 `void print2file_shark_results_new (SHARK_DATA * shark_dat)`
- 5.32.1.13 `void print2file_shark_results_old (SHARK_DATA * shark_dat)`
- 5.32.1.14 `int read_equilrxn (SHARK_DATA * shark_dat)`
- 5.32.1.15 `int read_massbalance (SHARK_DATA * shark_dat)`
- 5.32.1.16 `int read_options (SHARK_DATA * shark_dat)`
- 5.32.1.17 `int read_scenario (SHARK_DATA * shark_dat)`
- 5.32.1.18 `int read_species (SHARK_DATA * shark_dat)`
- 5.32.1.19 `int read_unsteadyrxn (SHARK_DATA * shark_dat)`
- 5.32.1.20 `int setup_SHARK_DATA (FILE * file, int(*) (const Matrix< double > &x, Matrix< double > &res, const void *data) residual, int(*) (const Matrix< double > &x, Matrix< double > &gama, const void *data) activity, int(*) (const Matrix< double > &r, Matrix< double > &p, const void *data) precon, SHARK_DATA * dat, const void * activity_data, const void * residual_data, const void * precon_data, const void * other_data)`
- 5.32.1.21 `int SHARK (SHARK_DATA * shark_dat)`
- 5.32.1.22 `int shark_add_customResidual (int i, double(*) (const Matrix< double > &x, SHARK_DATA *shark_dat, const void *other_data) other_res, SHARK_DATA * shark_dat)`
- 5.32.1.23 `int shark_energy_calculations (SHARK_DATA * shark_dat)`
- 5.32.1.24 `int shark_executioner (SHARK_DATA * shark_dat)`
- 5.32.1.25 `int shark_guess (SHARK_DATA * shark_dat)`

- 5.32.1.26 int shark_initial_conditions (SHARK_DATA * shark_dat)
- 5.32.1.27 int shark_parameter_check (SHARK_DATA * shark_dat)
- 5.32.1.28 int shark_pH_finder (SHARK_DATA * shark_dat)
- 5.32.1.29 int shark_postprocesses (SHARK_DATA * shark_dat)
- 5.32.1.30 int shark_preprocesses (SHARK_DATA * shark_dat)
- 5.32.1.31 int shark_reset (SHARK_DATA * shark_dat)
- 5.32.1.32 int shark_residual (const Matrix< double > & x, Matrix< double > & F, const void * data)
- 5.32.1.33 int SHARK_SCENARIO (const char * yaml_input)
- 5.32.1.34 int shark_solver (SHARK_DATA * shark_dat)
- 5.32.1.35 int shark_temperature_calculations (SHARK_DATA * shark_dat)
- 5.32.1.36 int SHARK_TESTS ()
- 5.32.1.37 int shark_timestep_adapt (SHARK_DATA * shark_dat)
- 5.32.1.38 int shark_timestep_const (SHARK_DATA * shark_dat)

5.33 shark.h File Reference

```
#include "mola.h"
#include "macaw.h"
#include "lark.h"
#include "yaml_wrapper.h"
```

Classes

- class [MasterSpeciesList](#)
- class [Reaction](#)
- class [MassBalance](#)
- class [UnsteadyReaction](#)
- class [Mechanism](#)
- class [Precipitation](#)
- class [UnsteadyPrecipitation](#)
- struct [SHARK_DATA](#)

Macros

- #define [Rstd](#) 8.3144621

Typedefs

- typedef struct [SHARK_DATA](#) SHARK_DATA

Enumerations

- enum `valid_act` {
`IDEAL`, `DAVIES`, `DEBYE_HUCKEL`, `DAVIES_LADSHAW`,
`SIT`, `PITZER` }

Functions

- void `print2file_shark_info` (`SHARK_DATA` *shark_dat)
- void `print2file_shark_header` (`SHARK_DATA` *shark_dat)
- void `print2file_shark_results_new` (`SHARK_DATA` *shark_dat)
- void `print2file_shark_results_old` (`SHARK_DATA` *shark_dat)
- int `ideal_solution` (const `Matrix`< double > &x, `Matrix`< double > &F, const void *data)
- int `Davies_equation` (const `Matrix`< double > &x, `Matrix`< double > &F, const void *data)
- int `DebyeHuckel_equation` (const `Matrix`< double > &x, `Matrix`< double > &F, const void *data)
- int `DaviesLadshaw_equation` (const `Matrix`< double > &x, `Matrix`< double > &F, const void *data)
- int `act_choice` (const std::string &input)
- bool `linesearch_choice` (const std::string &input)
- int `linearsolve_choice` (const std::string &input)
- int `Convert2LogConcentration` (const `Matrix`< double > &x, `Matrix`< double > &logx)
- int `Convert2Concentration` (const `Matrix`< double > &logx, `Matrix`< double > &x)
- int `read_scenario` (`SHARK_DATA` *shark_dat)
- int `read_options` (`SHARK_DATA` *shark_dat)
- int `read_species` (`SHARK_DATA` *shark_dat)
- int `read_massbalance` (`SHARK_DATA` *shark_dat)
- int `read_equilrxn` (`SHARK_DATA` *shark_dat)
- int `read_unsteadyrxn` (`SHARK_DATA` *shark_dat)
- int `setup_SHARK_DATA` (FILE *file, int(*residual)(const `Matrix`< double > &x, `Matrix`< double > &res, const void *data), int(*activity)(const `Matrix`< double > &x, `Matrix`< double > &gama, const void *data), int(*precond)(const `Matrix`< double > &r, `Matrix`< double > &p, const void *data), `SHARK_DATA` *dat, const void *activity_data, const void *residual_data, const void *precon_data, const void *other_data)
- int `shark_add_customResidual` (int i, double(*other_res)(const `Matrix`< double > &x, `SHARK_DATA` *shark_dat, const void *other_data), `SHARK_DATA` *shark_dat)
- int `shark_parameter_check` (`SHARK_DATA` *shark_dat)
- int `shark_energy_calculations` (`SHARK_DATA` *shark_dat)
- int `shark_temperature_calculations` (`SHARK_DATA` *shark_dat)
- int `shark_pH_finder` (`SHARK_DATA` *shark_dat)
- int `shark_guess` (`SHARK_DATA` *shark_dat)
- int `shark_initial_conditions` (`SHARK_DATA` *shark_dat)
- int `shark_executioner` (`SHARK_DATA` *shark_dat)
- int `shark_timestep_const` (`SHARK_DATA` *shark_dat)
- int `shark_timestep_adapt` (`SHARK_DATA` *shark_dat)
- int `shark_preprocesses` (`SHARK_DATA` *shark_dat)
- int `shark_solver` (`SHARK_DATA` *shark_dat)
- int `shark_postprocesses` (`SHARK_DATA` *shark_dat)
- int `shark_reset` (`SHARK_DATA` *shark_dat)
- int `shark_residual` (const `Matrix`< double > &x, `Matrix`< double > &F, const void *data)
- int `SHARK` (`SHARK_DATA` *shark_dat)
- int `SHARK_SCENARIO` (const char *yaml_input)
- int `SHARK_TESTS` ()

5.33.1 Macro Definition Documentation

5.33.1.1 `#define Rstd 8.3144621`

5.33.2 Typedef Documentation

5.33.2.1 `typedef struct SHARK_DATA SHARK_DATA`

5.33.3 Enumeration Type Documentation

5.33.3.1 `enum valid_act`

Enumerator

IDEAL

DAVIES

DEBYE_HUCKEL

DAVIES_LADSHAW

SIT

PITZER

5.33.4 Function Documentation

5.33.4.1 `int act_choice (const std::string & input)`

5.33.4.2 `int Convert2Concentration (const Matrix< double > & logx, Matrix< double > & x)`

5.33.4.3 `int Convert2LogConcentration (const Matrix< double > & x, Matrix< double > & logx)`

5.33.4.4 `int Davies_equation (const Matrix< double > & x, Matrix< double > & F, const void * data)`

5.33.4.5 `int DaviesLadshaw_equation (const Matrix< double > & x, Matrix< double > & F, const void * data)`

5.33.4.6 `int DebyeHuckel_equation (const Matrix< double > & x, Matrix< double > & F, const void * data)`

5.33.4.7 `int ideal_solution (const Matrix< double > & x, Matrix< double > & F, const void * data)`

5.33.4.8 `int linearsolve_choice (const std::string & input)`

5.33.4.9 `bool linesearch_choice (const std::string & input)`

5.33.4.10 `void print2file_shark_header (SHARK_DATA * shark.dat)`

5.33.4.11 `void print2file_shark_info (SHARK_DATA * shark.dat)`

5.33.4.12 `void print2file_shark_results_new (SHARK_DATA * shark.dat)`

5.33.4.13 `void print2file_shark_results_old (SHARK_DATA * shark.dat)`

5.33.4.14 `int read_equilrxn (SHARK_DATA * shark.dat)`

5.33.4.15 `int read_massbalance (SHARK_DATA * shark.dat)`

5.33.4.16 `int read_options (SHARK_DATA * shark.dat)`

- 5.33.4.17 int read_scenario (SHARK_DATA * shark_dat)
- 5.33.4.18 int read_species (SHARK_DATA * shark_dat)
- 5.33.4.19 int read_unsteadyrxn (SHARK_DATA * shark_dat)
- 5.33.4.20 int setup_SHARK_DATA (FILE * file, int(*) (const Matrix< double > &x, Matrix< double > &res, const void *data) residual, int(*) (const Matrix< double > &x, Matrix< double > &gama, const void *data) activity, int(*) (const Matrix< double > &r, Matrix< double > &p, const void *data) precon, SHARK_DATA * dat, const void * activity_data, const void * residual_data, const void * precon_data, const void * other_data)
- 5.33.4.21 int SHARK (SHARK_DATA * shark_dat)
- 5.33.4.22 int shark_add_customResidual (int i, double(*) (const Matrix< double > &x, SHARK_DATA *shark_dat, const void *other_data) other_res, SHARK_DATA * shark_dat)
- 5.33.4.23 int shark_energy_calculations (SHARK_DATA * shark_dat)
- 5.33.4.24 int shark_executioner (SHARK_DATA * shark_dat)
- 5.33.4.25 int shark_guess (SHARK_DATA * shark_dat)
- 5.33.4.26 int shark_initial_conditions (SHARK_DATA * shark_dat)
- 5.33.4.27 int shark_parameter_check (SHARK_DATA * shark_dat)
- 5.33.4.28 int shark_pH_finder (SHARK_DATA * shark_dat)
- 5.33.4.29 int shark_postprocesses (SHARK_DATA * shark_dat)
- 5.33.4.30 int shark_preprocesses (SHARK_DATA * shark_dat)
- 5.33.4.31 int shark_reset (SHARK_DATA * shark_dat)
- 5.33.4.32 int shark_residual (const Matrix< double > & x, Matrix< double > & F, const void * data)
- 5.33.4.33 int SHARK_SCENARIO (const char * yml_input)
- 5.33.4.34 int shark_solver (SHARK_DATA * shark_dat)
- 5.33.4.35 int shark_temperature_calculations (SHARK_DATA * shark_dat)
- 5.33.4.36 int SHARK_TESTS ()
- 5.33.4.37 int shark_timestep_adapt (SHARK_DATA * shark_dat)
- 5.33.4.38 int shark_timestep_const (SHARK_DATA * shark_dat)

5.34 skua.cpp File Reference

```
#include "skua.h"
```

Functions

- void [print2file_species_header](#) (FILE *Output, [SKUA_DATA](#) *skua_dat, int i)
- void [print2file_SKUA_time_header](#) (FILE *Output, [SKUA_DATA](#) *skua_dat, int i)
- void [print2file_SKUA_header](#) ([SKUA_DATA](#) *skua_dat)
- void [print2file_SKUA_results_old](#) ([SKUA_DATA](#) *skua_dat)
- void [print2file_SKUA_results_new](#) ([SKUA_DATA](#) *skua_dat)
- double [default_Dc](#) (int i, int l, const void *data)
- double [default_kf](#) (int i, const void *data)
- double [const_Dc](#) (int i, int l, const void *data)
- double [simple_darken_Dc](#) (int i, int l, const void *data)
- double [theoretical_darken_Dc](#) (int i, int l, const void *data)
- double [empirical_kf](#) (int i, const void *data)
- double [const_kf](#) (int i, const void *data)
- int [molefractionCheck](#) ([SKUA_DATA](#) *skua_dat)
- int [setup_SKUA_DATA](#) (FILE *file, double(*eval_Dc)(int i, int l, const void *user_data), double(*eval_Kf)(int i, const void *user_data), const void *user_data, [MIXED_GAS](#) *gas_data, [SKUA_DATA](#) *skua_dat)
- int [SKUA_Executioner](#) ([SKUA_DATA](#) *skua_dat)
- int [set_SKUA_ICs](#) ([SKUA_DATA](#) *skua_dat)
- int [set_SKUA_timestep](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_preprocesses](#) ([SKUA_DATA](#) *skua_dat)
- int [set_SKUA_params](#) (const void *user_data)
- int [SKUA_postprocesses](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_reset](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_CYCLE_TEST01](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_CYCLE_TEST02](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_LOW_TEST03](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_MID_TEST04](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_SCENARIOS](#) (const char *scene, const char *sorber, const char *comp, const char *sorbate)
- int [SKUA_TESTS](#) ()

5.34.1 Function Documentation

5.34.1.1 double [const_Dc](#) (int *i*, int *l*, const void * *data*)

5.34.1.2 double [const_kf](#) (int *i*, const void * *data*)

5.34.1.3 double [default_Dc](#) (int *i*, int *l*, const void * *data*)

5.34.1.4 double [default_kf](#) (int *i*, const void * *data*)

5.34.1.5 double [empirical_kf](#) (int *i*, const void * *data*)

5.34.1.6 int [molefractionCheck](#) ([SKUA_DATA](#) * *skua_dat*)

5.34.1.7 void [print2file_SKUA_header](#) ([SKUA_DATA](#) * *skua_dat*)

5.34.1.8 void [print2file_SKUA_results_new](#) ([SKUA_DATA](#) * *skua_dat*)

5.34.1.9 void [print2file_SKUA_results_old](#) ([SKUA_DATA](#) * *skua_dat*)

5.34.1.10 void [print2file_SKUA_time_header](#) (FILE * *Output*, [SKUA_DATA](#) * *skua_dat*, int *i*)

```

5.34.1.11 void print2file_species_header ( FILE * Output, SKUA_DATA * skua_dat, int i )

5.34.1.12 int set_SKUA_ICs ( SKUA_DATA * skua_dat )

5.34.1.13 int set_SKUA_params ( const void * user_data )

5.34.1.14 int set_SKUA_timestep ( SKUA_DATA * skua_dat )

5.34.1.15 int setup_SKUA_DATA ( FILE * file, double (*)(int i, int l, const void *user_data) eval_Dc, double (*)(int i, const void *user_data) eval_Kf, const void * user_data, MIXED_GAS * gas_data, SKUA_DATA * skua_dat )

5.34.1.16 double simple_darken_Dc ( int i, int l, const void * data )

5.34.1.17 int SKUA ( SKUA_DATA * skua_dat )

5.34.1.18 int SKUA_CYCLE_TEST01 ( SKUA_DATA * skua_dat )

5.34.1.19 int SKUA_CYCLE_TEST02 ( SKUA_DATA * skua_dat )

5.34.1.20 int SKUA_Executioner ( SKUA_DATA * skua_dat )

5.34.1.21 int SKUA_LOW_TEST03 ( SKUA_DATA * skua_dat )

5.34.1.22 int SKUA_MID_TEST04 ( SKUA_DATA * skua_dat )

5.34.1.23 int SKUA_postprocesses ( SKUA_DATA * skua_dat )

5.34.1.24 int SKUA_preprocesses ( SKUA_DATA * skua_dat )

5.34.1.25 int SKUA_reset ( SKUA_DATA * skua_dat )

5.34.1.26 int SKUA_SCENARIOS ( const char * scene, const char * sorbent, const char * comp, const char * sorbate )

5.34.1.27 int SKUA_TESTS ( )

5.34.1.28 double theoretical_darken_Dc ( int i, int l, const void * data )

```

5.35 skua.h File Reference

```

#include "finch.h"
#include "magpie.h"
#include "egret.h"

```

Classes

- struct [SKUA_PARAM](#)
- struct [SKUA_DATA](#)

Macros

- #define [SKUA_HPP_](#)
- #define [D_inf](#)(Dref, Tref, B, p, T) (Dref * pow(p+sqrt([DBL_EPSILON](#)),(Tref/T)-B))
- #define [D_o](#)(Diff, E, T) (Diff * exp(-E/([Rstd](#)*T)))
- #define [D_c](#)(Diff, phi) (Diff * (1.0/((1.0+1.1E-6)-phi)))

Functions

- void [print2file_species_header](#) (FILE *Output, [SKUA_DATA](#) *skua_dat, int i)
- void [print2file_SKUA_time_header](#) (FILE *Output, [SKUA_DATA](#) *skua_dat, int i)
- void [print2file_SKUA_header](#) ([SKUA_DATA](#) *skua_dat)
- void [print2file_SKUA_results_old](#) ([SKUA_DATA](#) *skua_dat)
- void [print2file_SKUA_results_new](#) ([SKUA_DATA](#) *skua_dat)
- double [default_Dc](#) (int i, int l, const void *data)
- double [default_kf](#) (int i, const void *data)
- double [const_Dc](#) (int i, int l, const void *data)
- double [simple_darken_Dc](#) (int i, int l, const void *data)
- double [theoretical_darken_Dc](#) (int i, int l, const void *data)
- double [empirical_kf](#) (int i, const void *data)
- double [const_kf](#) (int i, const void *data)
- int [molefractionCheck](#) ([SKUA_DATA](#) *skua_dat)
- int [setup_SKUA_DATA](#) (FILE *file, double(*eval_Dc)(int i, int l, const void *user_data), double(*eval_Kf)(int i, const void *user_data), const void *user_data, [MIXED_GAS](#) *gas_data, [SKUA_DATA](#) *skua_dat)
- int [SKUA_Executioner](#) ([SKUA_DATA](#) *skua_dat)
- int [set_SKUA_ICs](#) ([SKUA_DATA](#) *skua_dat)
- int [set_SKUA_timestep](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_preprocesses](#) ([SKUA_DATA](#) *skua_dat)
- int [set_SKUA_params](#) (const void *user_data)
- int [SKUA_postprocesses](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_reset](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_CYCLE_TEST01](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_CYCLE_TEST02](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_LOW_TEST03](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_MID_TEST04](#) ([SKUA_DATA](#) *skua_dat)
- int [SKUA_SCENARIOS](#) (const char *scene, const char *sorbent, const char *comp, const char *sorbate)
- int [SKUA_TESTS](#) ()

5.35.1 Macro Definition Documentation

- 5.35.1.1 `#define D_c(Diff, phi)(Diff * (1.0/((1.0+1.1E-6)-phi)))`
- 5.35.1.2 `#define D_inf(Dref, Tref, B, p, T)(Dref * pow(p+sqrt(DBL_EPSILON),(Tref/T)-B))`
- 5.35.1.3 `#define D_o(Diff, E, T)(Diff * exp(-E/(Rstd*T)))`
- 5.35.1.4 `#define SKUA_HPP_`

5.35.2 Function Documentation

- 5.35.2.1 `double const_Dc (int i, int l, const void * data)`
- 5.35.2.2 `double const_kf (int i, const void * data)`
- 5.35.2.3 `double default_Dc (int i, int l, const void * data)`
- 5.35.2.4 `double default_kf (int i, const void * data)`
- 5.35.2.5 `double empirical_kf (int i, const void * data)`

```

5.35.2.6  int molefractionCheck ( SKUA_DATA * skua_dat )

5.35.2.7  void print2file_SKUA_header ( SKUA_DATA * skua_dat )

5.35.2.8  void print2file_SKUA_results_new ( SKUA_DATA * skua_dat )

5.35.2.9  void print2file_SKUA_results_old ( SKUA_DATA * skua_dat )

5.35.2.10 void print2file_SKUA_time_header ( FILE * Output, SKUA_DATA * skua_dat, int i )

5.35.2.11 void print2file_species_header ( FILE * Output, SKUA_DATA * skua_dat, int i )

5.35.2.12 int set_SKUA_ICs ( SKUA_DATA * skua_dat )

5.35.2.13 int set_SKUA_params ( const void * user_data )

5.35.2.14 int set_SKUA_timestep ( SKUA_DATA * skua_dat )

5.35.2.15 int setup_SKUA_DATA ( FILE * file, double (*)(int i, int l, const void *user_data) eval_Dc, double (*)(int i, const void
    *user_data) eval_Kf, const void * user_data, MIXED_GAS * gas_data, SKUA_DATA * skua_dat )

5.35.2.16 double simple_darken_Dc ( int i, int l, const void * data )

5.35.2.17 int SKUA ( SKUA_DATA * skua_dat )

5.35.2.18 int SKUA_CYCLE_TEST01 ( SKUA_DATA * skua_dat )

5.35.2.19 int SKUA_CYCLE_TEST02 ( SKUA_DATA * skua_dat )

5.35.2.20 int SKUA_Executioner ( SKUA_DATA * skua_dat )

5.35.2.21 int SKUA_LOW_TEST03 ( SKUA_DATA * skua_dat )

5.35.2.22 int SKUA_MID_TEST04 ( SKUA_DATA * skua_dat )

5.35.2.23 int SKUA_postprocesses ( SKUA_DATA * skua_dat )

5.35.2.24 int SKUA_preprocesses ( SKUA_DATA * skua_dat )

5.35.2.25 int SKUA_reset ( SKUA_DATA * skua_dat )

5.35.2.26 int SKUA_SCENARIOS ( const char * scene, const char * sorbent, const char * comp, const char * sorbate )

5.35.2.27 int SKUA_TESTS ( )

5.35.2.28 double theoretical_darken_Dc ( int i, int l, const void * data )

```

5.36 skua_opt.cpp File Reference

```
#include "skua_opt.h"
```

Functions

- [int SKUA_OPT_set_y \(SKUA_OPT_DATA *skua_opt\)](#)

- int [initial_guess_SKUA](#) ([SKUA_OPT_DATA](#) *[skua_opt](#))
- void [eval_SKUA_Uptake](#) (const double **par*, int *m_dat*, const void **data*, double **fvec*, int **info*)
- int [SKUA_OPTIMIZE](#) (const char **scene*, const char **sorbent*, const char **comp*, const char **sorbate*, const char **data*)

5.36.1 Function Documentation

5.36.1.1 void [eval_SKUA_Uptake](#) (const double * *par*, int *m_dat*, const void * *data*, double * *fvec*, int * *info*)

5.36.1.2 int [initial_guess_SKUA](#) ([SKUA_OPT_DATA](#) * *skua_opt*)

5.36.1.3 int [SKUA_OPT_set_y](#) ([SKUA_OPT_DATA](#) * *skua_opt*)

5.36.1.4 int [SKUA_OPTIMIZE](#) (const char * *scene*, const char * *sorbent*, const char * *comp*, const char * *sorbate*, const char * *data*)

5.37 skua_opt.h File Reference

```
#include "skua.h"
```

Classes

- struct [SKUA_OPT_DATA](#)

Functions

- int [SKUA_OPT_set_y](#) ([SKUA_OPT_DATA](#) *[skua_opt](#))
- int [initial_guess_SKUA](#) ([SKUA_OPT_DATA](#) *[skua_opt](#))
- void [eval_SKUA_Uptake](#) (const double **par*, int *m_dat*, const void **data*, double **fvec*, int **info*)
- int [SKUA_OPTIMIZE](#) (const char **scene*, const char **sorbent*, const char **comp*, const char **sorbate*, const char **data*)

5.37.1 Function Documentation

5.37.1.1 void [eval_SKUA_Uptake](#) (const double * *par*, int *m_dat*, const void * *data*, double * *fvec*, int * *info*)

5.37.1.2 int [initial_guess_SKUA](#) ([SKUA_OPT_DATA](#) * *skua_opt*)

5.37.1.3 int [SKUA_OPT_set_y](#) ([SKUA_OPT_DATA](#) * *skua_opt*)

5.37.1.4 int [SKUA_OPTIMIZE](#) (const char * *scene*, const char * *sorbent*, const char * *comp*, const char * *sorbate*, const char * *data*)

5.38 Trajectory.cpp File Reference

```
#include "Trajectory.h"
```

Functions

- double [Magnetic_R](#) (const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, int i, double b, double mu_0, double chi_p, double M, double H0, double a)
- double [Magnetic_T](#) (const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, int i, double b, double mu_0, double chi_p, double M, double H0, double a)
- double [Grav_R](#) (const [Matrix](#)< double > &dX, int i, double b, double rho_p, double rho_f)
- double [Grav_T](#) (const [Matrix](#)< double > &dX, int i, double b, double rho_p, double rho_f)
- double [Van_R](#) (const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, int i, double Hamaker, double b, double a)
- double [V_RAD](#) (const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, int i, double V0, double rho_f, double a, double eta)
- double [V_THETA](#) (const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, int i, double V0, double rho_f, double a, double eta)
- double [Brown_RAD](#) (double n_rand, double m_rand, double sigma_n, double sigma_m)
- double [Brown_THETA](#) (double s_rand, double t_rand, double sigma_n, double sigma_m)
- int [POLAR](#) ([Matrix](#)< double > &POL, const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, const void *data, int i)
- double [RADIAL_FORCE](#) (const [Matrix](#)< double > &POL, double eta, double b, double mp, double t, double a)
- double [TANGENTIAL_FORCE](#) (const [Matrix](#)< double > &POL, const [Matrix](#)< double > &dY, double eta, double b, double mp, double t, double a, int i)
- int [CARTESIAN](#) (const [Matrix](#)< double > &POL, [Matrix](#)< double > &H, const [Matrix](#)< double > &dY, double i, const void *data)
- int [DISPLACEMENT](#) ([Matrix](#)< double > &dX, [Matrix](#)< double > &dY, const [Matrix](#)< double > &H, int i)
- int [LOCATION](#) (const [Matrix](#)< double > &dY, const [Matrix](#)< double > &dX, [Matrix](#)< double > &X, [Matrix](#)< double > &Y, int i)
- double [Removal_Efficiency](#) (double Sum_Cap, const void *data)
- int [Trajectory_SetupConstants](#) ([TRAJECTORY_DATA](#) *dat)
- int [Number_Generator](#) ([TRAJECTORY_DATA](#) *dat)
- int [Run_Trajectory](#) ()

5.38.1 Function Documentation

5.38.1.1 double [Brown_RAD](#) (double *n_rand*, double *m_rand*, double *sigma_n*, double *sigma_m*)

5.38.1.2 double [Brown_THETA](#) (double *s_rand*, double *t_rand*, double *sigma_n*, double *sigma_m*)

5.38.1.3 int [CARTESIAN](#) (const [Matrix](#)< double > & *POL*, [Matrix](#)< double > & *H*, const [Matrix](#)< double > & *dY*, double *i*, const void * *data*)

5.38.1.4 int [DISPLACEMENT](#) ([Matrix](#)< double > & *dX*, [Matrix](#)< double > & *dY*, const [Matrix](#)< double > & *H*, int *i*)

5.38.1.5 double [Grav_R](#) (const [Matrix](#)< double > & *dX*, int *i*, double *b*, double *rho_p*, double *rho_f*)

5.38.1.6 double [Grav_T](#) (const [Matrix](#)< double > & *dX*, int *i*, double *b*, double *rho_p*, double *rho_f*)

5.38.1.7 int [LOCATION](#) (const [Matrix](#)< double > & *dY*, const [Matrix](#)< double > & *dX*, [Matrix](#)< double > & *X*, [Matrix](#)< double > & *Y*, int *i*)

5.38.1.8 double [Magnetic_R](#) (const [Matrix](#)< double > & *dX*, const [Matrix](#)< double > & *dY*, int *i*, double *b*, double *mu_0*, double *chi_p*, double *M*, double *H0*, double *a*)

5.38.1.9 double [Magnetic_T](#) (const [Matrix](#)< double > & *dX*, const [Matrix](#)< double > & *dY*, int *i*, double *b*, double *mu_0*, double *chi_p*, double *M*, double *H0*, double *a*)

- 5.38.1.10 `int Number_Generator (TRAJECTORY_DATA * dat)`
- 5.38.1.11 `int POLAR (Matrix< double > & POL, const Matrix< double > & dX, const Matrix< double > & dY, const void * data, int i)`
- 5.38.1.12 `double RADIAL_FORCE (const Matrix< double > & POL, double eta, double b, double mp, double t, double a)`
- 5.38.1.13 `double Removal_Efficiency (double Sum_Cap, const void * data)`
- 5.38.1.14 `int Run_Trajectory ()`
- 5.38.1.15 `double TANGENTIAL_FORCE (const Matrix< double > & POL, const Matrix< double > & dY, double eta, double b, double mp, double t, double a, int i)`
- 5.38.1.16 `int Trajectory_SetupConstants (TRAJECTORY_DATA * dat)`
- 5.38.1.17 `double V_RAD (const Matrix< double > & dX, const Matrix< double > & dY, int i, double V0, double rho_f, double a, double eta)`
- 5.38.1.18 `double V_THETA (const Matrix< double > & dX, const Matrix< double > & dY, int i, double V0, double rho_f, double a, double eta)`
- 5.38.1.19 `double Van_R (const Matrix< double > & dX, const Matrix< double > & dY, int i, double Hamaker, double b, double a)`

5.39 Trajectory.h File Reference

```
#include "macaw.h"
#include <random>
#include <chrono>
```

Classes

- struct [TRAJECTORY_DATA](#)

Functions

- double [Magnetic_R](#) (const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, int i, double b, double mu_0, double chi_p, double M, double H0, double a)
- double [Magnetic_T](#) (const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, int i, double b, double mu_0, double chi_p, double M, double H0, double a)
- double [Grav_R](#) (const [Matrix](#)< double > &dX, int i, double b, double rho_p, double rho_f)
- double [Grav_T](#) (const [Matrix](#)< double > &dX, int i, double b, double rho_p, double rho_f)
- double [Van_R](#) (const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, int i, double Hamaker, double b, double a)
- double [V_RAD](#) (const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, int i, double V0, double rho_f, double a, double eta)
- double [V_THETA](#) (const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, int i, double V0, double rho_f, double a, double eta)
- double [Brown_RAD](#) (double n_rand, double m_rand, double sigma_n, double sigma_m)
- double [Brown_THETA](#) (double s_rand, double t_rand, double sigma_n, double sigma_m)
- int [POLAR](#) ([Matrix](#)< double > &POL, const [Matrix](#)< double > &dX, const [Matrix](#)< double > &dY, const void *data, int i)

- double **RADIAL_FORCE** (const **Matrix**< double > &POL, double eta, double b, double mp, double t, double a)
- double **TANGENTIAL_FORCE** (const **Matrix**< double > &POL, const **Matrix**< double > &dY, double eta, double b, double mp, double t, double a, int i)
- int **CARTESIAN** (const **Matrix**< double > &POL, **Matrix**< double > &H, const **Matrix**< double > &dY, double i, const void *data)
- int **DISPLACEMENT** (**Matrix**< double > &dX, **Matrix**< double > &dY, const **Matrix**< double > &H, int i)
- int **LOCATION** (const **Matrix**< double > &dY, const **Matrix**< double > &dX, **Matrix**< double > &X, **Matrix**< double > &Y, int i)
- double **Removal_Efficiency** (double Sum_Cap, const void *data)
- int **Trajectory_SetupConstants** (**TRAJECTORY_DATA** *dat)
- int **Number_Generator** (**TRAJECTORY_DATA** *dat)
- int **Run_Trajectory** ()

5.39.1 Function Documentation

5.39.1.1 double **Brown_RAD** (double *n_rand*, double *m_rand*, double *sigma_n*, double *sigma_m*)

5.39.1.2 double **Brown_THETA** (double *s_rand*, double *t_rand*, double *sigma_n*, double *sigma_m*)

5.39.1.3 int **CARTESIAN** (const **Matrix**< double > &POL, **Matrix**< double > &H, const **Matrix**< double > &dY, double *i*, const void * data)

5.39.1.4 int **DISPLACEMENT** (**Matrix**< double > &dX, **Matrix**< double > &dY, const **Matrix**< double > &H, int *i*)

5.39.1.5 double **Grav_R** (const **Matrix**< double > &dX, int *i*, double *b*, double *rho_p*, double *rho_f*)

5.39.1.6 double **Grav_T** (const **Matrix**< double > &dX, int *i*, double *b*, double *rho_p*, double *rho_f*)

5.39.1.7 int **LOCATION** (const **Matrix**< double > &dY, const **Matrix**< double > &dX, **Matrix**< double > &X, **Matrix**< double > &Y, int *i*)

5.39.1.8 double **Magnetic_R** (const **Matrix**< double > &dX, const **Matrix**< double > &dY, int *i*, double *b*, double *mu_0*, double *chi_p*, double *M*, double *H0*, double *a*)

5.39.1.9 double **Magnetic_T** (const **Matrix**< double > &dX, const **Matrix**< double > &dY, int *i*, double *b*, double *mu_0*, double *chi_p*, double *M*, double *H0*, double *a*)

5.39.1.10 int **Number_Generator** (**TRAJECTORY_DATA** * dat)

5.39.1.11 int **POLAR** (**Matrix**< double > &POL, const **Matrix**< double > &dX, const **Matrix**< double > &dY, const void * data, int *i*)

5.39.1.12 double **RADIAL_FORCE** (const **Matrix**< double > &POL, double *eta*, double *b*, double *mp*, double *t*, double *a*)

5.39.1.13 double **Removal_Efficiency** (double *Sum_Cap*, const void * data)

5.39.1.14 int **Run_Trajectory** ()

5.39.1.15 double **TANGENTIAL_FORCE** (const **Matrix**< double > &POL, const **Matrix**< double > &dY, double *eta*, double *b*, double *mp*, double *t*, double *a*, int *i*)

5.39.1.16 int **Trajectory_SetupConstants** (**TRAJECTORY_DATA** * dat)

- 5.39.1.17 `double V_RAD (const Matrix< double > & dX, const Matrix< double > & dY, int i, double V0, double rho_f, double a, double eta)`
- 5.39.1.18 `double V_THETA (const Matrix< double > & dX, const Matrix< double > & dY, int i, double V0, double rho_f, double a, double eta)`
- 5.39.1.19 `double Van_R (const Matrix< double > & dX, const Matrix< double > & dY, int i, double Hamaker, double b, double a)`

5.40 ui.cpp File Reference

```
#include "ui.h"
```

Functions

- void [aui_help](#) ()
- void [bui_help](#) ()
- std::string [allLower](#) (const std::string &input)
- bool [exit](#) (const std::string &input)
- bool [help](#) (const std::string &input)
- bool [version](#) (const std::string &input)
- bool [test](#) (const std::string &input)
- bool [exec](#) (const std::string &input)
- bool [path](#) (const std::string &input)
- bool [input](#) (const std::string &input)
- bool [valid_test_string](#) (const std::string &input, UI_DATA *ui_dat)
- bool [valid_exec_string](#) (const std::string &input, UI_DATA *ui_dat)
- int [number_files](#) (UI_DATA *ui_dat)
- bool [valid_addon_options](#) (UI_DATA *ui_dat)
- void [display_help](#) (UI_DATA *ui_dat)
- void [display_version](#) (UI_DATA *ui_dat)
- int [invalid_input](#) (int count, int max)
- bool [valid_input_main](#) (UI_DATA *ui_dat)
- bool [valid_input_tests](#) (UI_DATA *ui_dat)
- bool [valid_input_execute](#) (UI_DATA *ui_dat)
- int [test_loop](#) (UI_DATA *ui_dat)
- int [exec_loop](#) (UI_DATA *ui_dat)
- int [run_test](#) (UI_DATA *ui_dat)
- int [run_exec](#) (UI_DATA *ui_dat)
- int [run_executable](#) (int argc, const char *argv[])

5.40.1 Function Documentation

- 5.40.1.1 `std::string allLower (const std::string & input)`
- 5.40.1.2 `void aui_help ()`
- 5.40.1.3 `void bui_help ()`
- 5.40.1.4 `void display_help (UI_DATA * ui_dat)`
- 5.40.1.5 `void display_version (UI_DATA * ui_dat)`

- 5.40.1.6 `bool exec (const std::string & input)`
- 5.40.1.7 `int exec_loop (UI_DATA * ui_dat)`
- 5.40.1.8 `bool exit (const std::string & input)`
- 5.40.1.9 `bool help (const std::string & input)`
- 5.40.1.10 `bool input (const std::string & input)`
- 5.40.1.11 `int invalid_input (int count, int max)`
- 5.40.1.12 `int number_files (UI_DATA * ui_dat)`
- 5.40.1.13 `bool path (const std::string & input)`
- 5.40.1.14 `int run_exec (UI_DATA * ui_dat)`
- 5.40.1.15 `int run_executable (int argc, const char * argv[])`
- 5.40.1.16 `int run_test (UI_DATA * ui_dat)`
- 5.40.1.17 `bool test (const std::string & input)`
- 5.40.1.18 `int test_loop (UI_DATA * ui_dat)`
- 5.40.1.19 `bool valid_addon_options (UI_DATA * ui_dat)`
- 5.40.1.20 `bool valid_exec_string (const std::string & input, UI_DATA * ui_dat)`
- 5.40.1.21 `bool valid_input_execute (UI_DATA * ui_dat)`
- 5.40.1.22 `bool valid_input_main (UI_DATA * ui_dat)`
- 5.40.1.23 `bool valid_input_tests (UI_DATA * ui_dat)`
- 5.40.1.24 `bool valid_test_string (const std::string & input, UI_DATA * ui_dat)`
- 5.40.1.25 `bool version (const std::string & input)`

5.41 ui.h File Reference

```
#include <fstream>
#include <string>
#include <iostream>
#include "error.h"
#include "yaml_wrapper.h"
#include "flock.h"
#include "school.h"
#include "sandbox.h"
#include "Trajectory.h"
```

Classes

- struct [UI_DATA](#)

Macros

- #define [UI_HPP_](#)
- #define [ECO_VERSION](#) "0.0 alpha"
- #define [ECO_EXECUTABLE](#) "eco0"

Enumerations

- enum [valid_options](#) {
 [TEST](#), [EXECUTE](#), [EXIT](#), [CONTINUE](#),
 [HELP](#), [dogfish](#), [eel](#), [egret](#),
 [finch](#), [lark](#), [macaw](#), [mola](#),
 [monkfish](#), [sandbox](#), [scopsowl](#), [shark](#),
 [skua](#), [gsta_opt](#), [magpie](#), [scops_opt](#),
 [skua_opt](#), [trajectory](#) }

Functions

- void [au_i_help](#) ()
- void [bui_help](#) ()
- std::string [allLower](#) (const std::string &[input](#))
- bool [exit](#) (const std::string &[input](#))
- bool [help](#) (const std::string &[input](#))
- bool [version](#) (const std::string &[input](#))
- bool [test](#) (const std::string &[input](#))
- bool [exec](#) (const std::string &[input](#))
- bool [path](#) (const std::string &[input](#))
- bool [input](#) (const std::string &[input](#))
- bool [valid_test_string](#) (const std::string &[input](#), [UI_DATA](#) *[ui_dat](#))
- bool [valid_exec_string](#) (const std::string &[input](#), [UI_DATA](#) *[ui_dat](#))
- int [number_files](#) ([UI_DATA](#) *[ui_dat](#))
- bool [valid_addon_options](#) ([UI_DATA](#) *[ui_dat](#))
- void [exec_option](#) ([UI_DATA](#) *[ui_dat](#))
- void [display_help](#) ([UI_DATA](#) *[ui_dat](#))
- void [display_version](#) ([UI_DATA](#) *[ui_dat](#))
- int [invalid_input](#) (int count, int [max](#))
- bool [valid_input_main](#) ([UI_DATA](#) *[ui_dat](#))
- bool [valid_input_tests](#) ([UI_DATA](#) *[ui_dat](#))
- bool [valid_input_execute](#) ([UI_DATA](#) *[ui_dat](#))
- int [test_loop](#) ([UI_DATA](#) *[ui_dat](#))
- int [exec_loop](#) ([UI_DATA](#) *[ui_dat](#))
- int [run_test](#) ([UI_DATA](#) *[ui_dat](#))
- int [run_exec](#) ([UI_DATA](#) *[ui_dat](#))
- int [run_executable](#) (int argc, const char *argv[])

5.41.1 Macro Definition Documentation

5.41.1.1 `#define ECO_EXECUTABLE "eco0"`

5.41.1.2 `#define ECO_VERSION "0.0 alpha"`

5.41.1.3 `#define UI_HPP_`

5.41.2 Enumeration Type Documentation

5.41.2.1 `enum valid_options`

Enumerator

TEST

EXECUTE

EXIT

CONTINUE

HELP

dogfish

eel

egret

finch

lark

macaw

mola

monkfish

sandbox

scopsowl

shark

skua

gsta_opt

magpie

scops_opt

skua_opt

trajectory

5.41.3 Function Documentation

5.41.3.1 `std::string allLower (const std::string & input)`

5.41.3.2 `void aui_help ()`

5.41.3.3 `void bui_help ()`

5.41.3.4 `void display_help (UI_DATA * ui_dat)`

5.41.3.5 `void display_version (UI_DATA * ui_dat)`

5.41.3.6 `bool exec (const std::string & input)`

5.41.3.7 int exec_loop (UI_DATA * *ui_dat*)

5.41.3.8 void exec_option (UI_DATA * *ui_dat*)

5.41.3.9 bool exit (const std::string & *input*)

5.41.3.10 bool help (const std::string & *input*)

5.41.3.11 bool input (const std::string & *input*)

5.41.3.12 int invalid_input (int *count*, int *max*)

5.41.3.13 int number_files (UI_DATA * *ui_dat*)

5.41.3.14 bool path (const std::string & *input*)

5.41.3.15 int run_exec (UI_DATA * *ui_dat*)

5.41.3.16 int run_executable (int *argc*, const char * *argv*[])

5.41.3.17 int run_test (UI_DATA * *ui_dat*)

5.41.3.18 bool test (const std::string & *input*)

5.41.3.19 int test_loop (UI_DATA * *ui_dat*)

5.41.3.20 bool valid_addon_options (UI_DATA * *ui_dat*)

5.41.3.21 bool valid_exec_string (const std::string & *input*, UI_DATA * *ui_dat*)

5.41.3.22 bool valid_input_execute (UI_DATA * *ui_dat*)

5.41.3.23 bool valid_input_main (UI_DATA * *ui_dat*)

5.41.3.24 bool valid_input_tests (UI_DATA * *ui_dat*)

5.41.3.25 bool valid_test_string (const std::string & *input*, UI_DATA * *ui_dat*)

5.41.3.26 bool version (const std::string & *input*)

5.42 yaml_wrapper.cpp File Reference

```
#include "yaml_wrapper.h"
```

Functions

- int [YAML_WRAPPER_TESTS](#) ()
- int [YAML_CPP_TEST](#) (const char **file*)

5.42.1 Function Documentation

5.42.1.1 int [YAML_CPP_TEST](#) (const char * *file*)

5.42.1.2 `int YAML_WRAPPER_TESTS ()`

5.43 `yaml_wrapper.h` File Reference

```
#include "yaml.h"
#include "error.h"
#include <map>
#include <string>
#include <iostream>
#include <utility>
#include <stdexcept>
```

Classes

- class [ValueTypePair](#)
- class [KeyValueMap](#)
- class [SubHeader](#)
- class [Header](#)
- class [Document](#)
- class [YamlWrapper](#)
- class [yaml_cpp_class](#)

Typedefs

- typedef enum [data_type](#) [data_type](#)
- typedef enum [header_state](#) [header_state](#)

Enumerations

- enum [data_type](#) {
 [STRING](#), [BOOLEAN](#), [DOUBLE](#), [INT](#),
 [UNKNOWN](#) }
- enum [header_state](#) { [ANCHOR](#), [ALIAS](#), [NONE](#) }

Functions

- int [YAML_WRAPPER_TESTS](#) ()
- int [YAML_CPP_TEST](#) (const char *file)

5.43.1 Typedef Documentation

5.43.1.1 typedef enum [data_type](#) [data_type](#)

5.43.1.2 typedef enum [header_state](#) [header_state](#)

5.43.2 Enumeration Type Documentation

5.43.2.1 enum [data_type](#)

Enumerator

STRING

BOOLEAN

DOUBLE

INT

UNKNOWN

5.43.2.2 enum header_state

Enumerator

ANCHOR

ALIAS

NONE

5.43.3 Function Documentation

5.43.3.1 int YAML_CPP_TEST (const char * *file*)

5.43.3.2 int YAML_WRAPPER_TESTS ()

Index

- ~Atom
 - Atom, [9](#)
- ~Document
 - Document, [16](#)
- ~Header
 - Header, [37](#)
- ~KeyValueMap
 - KeyValueMap, [39](#)
- ~MassBalance
 - MassBalance, [42](#)
- ~MasterSpeciesList
 - MasterSpeciesList, [43](#)
- ~Matrix
 - Matrix, [46](#)
- ~Molecule
 - Molecule, [51](#)
- ~PeriodicTable
 - PeriodicTable, [59](#)
- ~Reaction
 - Reaction, [65](#)
- ~SubHeader
 - SubHeader, [81](#)
- ~UnsteadyReaction
 - UnsteadyReaction, [89](#)
- ~ValueTypePair
 - ValueTypePair, [92](#)
- ~YamlWrapper
 - YamlWrapper, [95](#)
- ~yaml_cpp_class
 - yaml_cpp_class, [94](#)
- A
 - magpie.h, [122](#)
- a
 - TRAJECTORY_DATA, [85](#)
- ALIAS
 - yaml_wrapper.h, [153](#)
- ANCHOR
 - yaml_wrapper.h, [153](#)
- A_separator
 - TRAJECTORY_DATA, [85](#)
- A_wire
 - TRAJECTORY_DATA, [85](#)
- ARNOLDI_DATA, [7](#)
 - beta, [7](#)
 - e1, [7](#)
 - Hkp1, [7](#)
 - hp1, [7](#)
 - iter, [7](#)
 - k, [7](#)
 - Output, [7](#)
 - sum, [7](#)
 - v, [7](#)
 - Vk, [7](#)
 - w, [8](#)
 - yk, [8](#)
- abs_tol_bias
 - SCOPSOWL_OPT_DATA, [69](#)
 - SKUA_OPT_DATA, [77](#)
- act_choice
 - shark.cpp, [134](#)
 - shark.h, [137](#)
- act_fun
 - SHARK_DATA, [73](#)
- activation_energy
 - SCOPSOWL_PARAM_DATA, [71](#)
 - SKUA_PARAM, [79](#)
 - UnsteadyReaction, [91](#)
- activity_data
 - SHARK_DATA, [73](#)
- activity_new
 - SHARK_DATA, [73](#)
- activity_old
 - SHARK_DATA, [73](#)
- addDocKey
 - YamlWrapper, [95](#)
- addHeadKey
 - Document, [16](#)
- addKey
 - KeyValueMap, [39](#)
- addPair
 - Document, [16](#)
 - Header, [37](#)
 - KeyValueMap, [39](#)
 - SubHeader, [81](#)
- addSubKey
 - Header, [37](#)
- adjoint
 - Matrix, [46](#)
- adsorb_index
 - SCOPSOWL_OPT_DATA, [69](#)
 - SKUA_OPT_DATA, [77](#)
- Adsorbable
 - SCOPSOWL_PARAM_DATA, [71](#)
 - SKUA_PARAM, [79](#)
- affinity
 - SCOPSOWL_PARAM_DATA, [71](#)
 - SKUA_PARAM, [79](#)
- Ai

- OPTRANS_DATA, 57
- alias
 - SubHeader, 82
- alkalinity
 - MasterSpeciesList, 43
- all_pars
 - GSTA_OPT_DATA, 35
- allLower
 - ui.cpp, 147
 - ui.h, 150
- alpha
 - BACKTRACK_DATA, 11
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 13
 - GCR_DATA, 28
 - PCG_DATA, 58
- anchor_alias_dne
 - error.h, 104
- Ap
 - PCG_DATA, 58
- arg
 - GMRESR_DATA, 31
- arg_matrix_same
 - error.h, 104
- argc
 - UI_DATA, 87
- argv
 - UI_DATA, 87
- arnoldi
 - lark.cpp, 114
 - lark.h, 118
- arnoldi_dat
 - GMRESLP_DATA, 29
- As
 - SYSTEM_DATA, 83
- assertType
 - KeyValueMap, 39
 - ValueTypePair, 92
- Atom, 8
 - ~Atom, 9
 - Atom, 9
 - AtomCategory, 9
 - AtomName, 9
 - AtomState, 9
 - AtomSymbol, 9
 - atomic_number, 10
 - atomic_weight, 10
 - AtomicNumber, 9
 - AtomicWeight, 9
 - BondingElectrons, 9
 - Category, 10
 - DisplayInfo, 9
 - editAtomicWeight, 9
 - editElectrons, 9
 - editNeutrons, 9
 - editOxidationState, 9
 - editProtons, 9
 - editValence, 9
 - Electrons, 10
 - electrons, 10
 - Name, 10
 - NaturalState, 10
 - Neutrons, 10
 - neutrons, 10
 - oxidation_state, 10
 - OxidationState, 10
 - Protons, 10
 - protons, 10
 - Register, 10
 - removeElectron, 10
 - removeNeutron, 10
 - removeProton, 10
 - Symbol, 10
 - valence_e, 10
- AtomCategory
 - Atom, 9
- AtomName
 - Atom, 9
- AtomState
 - Atom, 9
- AtomSymbol
 - Atom, 9
- atomic_number
 - Atom, 10
- atomic_weight
 - Atom, 10
- AtomicNumber
 - Atom, 9
- AtomicWeight
 - Atom, 9
- atoms
 - Molecule, 52
- aii_help
 - ui.cpp, 147
 - ui.h, 150
- avg_fiber_density
 - MONKFISH_DATA, 54
- avg_norm
 - SYSTEM_DATA, 83
- avg_sorption
 - MONKFISH_PARAM, 55
- avg_sorption_old
 - MONKFISH_PARAM, 55
- avgDp
 - scopsowl.h, 130
- avgPar
 - gsta_opt.cpp, 111
 - gsta_opt.h, 112
- avgValue
 - gsta_opt.cpp, 111
 - gsta_opt.h, 112
- b
 - EX01_DATA, 20
 - EX02_DATA, 20
 - EX04_DATA, 20
 - EX15_DATA, 21

- TRAJECTORY_DATA, 85
- B0
 - TRAJECTORY_DATA, 85
- BOOLEAN
 - yaml_wrapper.h, 152
- BACKTRACK_DATA, 10
 - alpha, 11
 - constRho, 11
 - Fk, 11
 - lambdaMin, 11
 - normFkp1, 11
 - rho, 11
 - xk, 11
- backtrack_dat
 - PJFNK_DATA, 61
- backtrackLineSearch
 - lark.cpp, 114
 - lark.h, 118
- BasicUI
 - UI_DATA, 87
- begin
 - Document, 16
 - Header, 37
 - KeyValueMap, 40
 - YamlWrapper, 95
- best_par
 - GSTA_OPT_DATA, 35
- bestres
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 13
 - GCR_DATA, 28
 - GMRESLP_DATA, 29
 - GMRESRP_DATA, 32
 - PCG_DATA, 58
 - PICARD_DATA, 60
- bestx
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 13
 - GCR_DATA, 28
 - GMRESLP_DATA, 29
 - GMRESRP_DATA, 32
 - PCG_DATA, 58
 - PICARD_DATA, 60
 - PJFNK_DATA, 61
- beta
 - ARNOLDI_DATA, 7
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 13
 - FINCH_DATA, 24
 - GCR_DATA, 28
 - PCG_DATA, 58
 - TRAJECTORY_DATA, 85
- BiCGSTAB
 - lark.h, 117
- BiCGSTAB_DATA, 11
 - alpha, 12
 - bestres, 12
 - bestx, 12
 - beta, 12
 - breakdown, 12
 - iter, 12
 - maxit, 12
 - omega, 12
 - omega_old, 12
 - Output, 12
 - p, 12
 - r, 12
 - r0, 12
 - relres, 12
 - relres_base, 12
 - res, 12
 - rho, 12
 - rho_old, 12
 - s, 12
 - t, 12
 - tol_abs, 12
 - tol_rel, 12
 - v, 12
 - x, 13
 - y, 13
 - z, 13
- bicgstab
 - lark.cpp, 114
 - lark.h, 118
- bicgstab_dat
 - PJFNK_DATA, 61
- binary_diffusion
 - MIXED_GAS, 49
- binder_fraction
 - SCOPSOWL_DATA, 67
- binder_poresize
 - SCOPSOWL_DATA, 67
- binder_porosity
 - SCOPSOWL_DATA, 67
- BondingElectrons
 - Atom, 9
- Bounce
 - PJFNK_DATA, 61
- breakdown
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 13
 - GCR_DATA, 28
- Brown_RAD
 - Trajectory.cpp, 144
 - Trajectory.h, 146
- Brown_THETA
 - Trajectory.cpp, 144
 - Trajectory.h, 146
- buckley_leverett_ic
 - finch.cpp, 105
 - finch.h, 108
- buckley_leverett_params
 - finch.cpp, 105
 - finch.h, 108
- bui_help
 - ui.cpp, 147

- ui.h, 150
- burgers_bcs
 - finch.cpp, 105
 - finch.h, 108
- burgers_ic
 - finch.cpp, 105
 - finch.h, 108
- burgers_params
 - finch.cpp, 105
 - finch.h, 108
- C
 - Speciation_Test01_Data, 80
- c
 - CGS_DATA, 14
 - GCR_DATA, 28
- CGS
 - lark.h, 117
- CONTINUE
 - ui.h, 150
- c_temp
 - GCR_DATA, 28
- CARTESIAN
 - Trajectory.cpp, 144
 - Trajectory.h, 146
- CC_E
 - FINCH_DATA, 24
- CC_I
 - FINCH_DATA, 24
- CE3
 - egret.h, 101
- CGS_DATA, 13
 - alpha, 13
 - bestres, 13
 - bestx, 13
 - beta, 13
 - breakdown, 13
 - c, 14
 - iter, 14
 - maxit, 14
 - Output, 14
 - p, 14
 - r, 14
 - r0, 14
 - relres, 14
 - relres_base, 14
 - res, 14
 - rho, 14
 - sigma, 14
 - tol_abs, 14
 - tol_rel, 14
 - u, 14
 - v, 14
 - w, 14
 - x, 14
 - z, 14
- CL_E
 - FINCH_DATA, 24
- CL_I
 - FINCH_DATA, 24
- CN
 - FINCH_DATA, 24
- CR_E
 - FINCH_DATA, 24
- CR_I
 - FINCH_DATA, 24
- CT
 - Speciation_Test01_Data, 80
- CURVE_TEST03
 - scopsowl.cpp, 128
 - scopsowl.h, 131
- CURVE_TEST04
 - scopsowl.cpp, 128
 - scopsowl.h, 131
- CURVE_TEST05
 - scopsowl.cpp, 128
 - scopsowl.h, 131
- calculate_properties
 - egret.cpp, 100
 - egret.h, 102
- calculateAvgOxiState
 - Molecule, 51
- calculateEnergies
 - Reaction, 65
 - UnsteadyReaction, 89
- calculateEquilibrium
 - Reaction, 65
 - UnsteadyReaction, 89
- calculateRate
 - UnsteadyReaction, 89
- callroutine
 - FINCH_DATA, 24
- CanCalcG
 - Reaction, 65
- CanCalcHS
 - Reaction, 65
- Cap
 - TRAJECTORY_DATA, 85
- Carrier
 - SYSTEM_DATA, 83
- Cartesian
 - finch.h, 108
- Category
 - Atom, 10
- cgs
 - lark.cpp, 114
 - lark.h, 118
- cgs_dat
 - PJFNK_DATA, 61
- changeKey
 - Document, 16
 - Header, 37
 - YamlWrapper, 95
- char_length
 - MIXED_GAS, 49
- char_macro
 - SCOPSOWL_DATA, 67

- char_measure
 - SKUA_DATA, 76
- char_micro
 - SCOPSOWL_DATA, 67
- Charge
 - Molecule, 51
- charge
 - MasterSpeciesList, 43
 - Molecule, 52
- check_Mass
 - finch.cpp, 106
 - finch.h, 108
- CheckMass
 - FINCH_DATA, 24
- CheckMolefractions
 - MIXED_GAS, 49
- checkSpeciesEnergies
 - Reaction, 65
 - UnsteadyReaction, 89
- chi_p
 - TRAJECTORY_DATA, 85
- cleanup
 - yaml_cpp_class, 94
- clear
 - Document, 16
 - Header, 37
 - KeyValueMap, 40
 - SubHeader, 81
 - YamlWrapper, 95
- cofactor
 - Matrix, 46
- columnExtend
 - Matrix, 46
- columnExtract
 - Matrix, 46
- columnProjection
 - Matrix, 46
- columnReplace
 - Matrix, 46
- columnShrink
 - Matrix, 46
- columnVectorFill
 - Matrix, 46
- columns
 - Matrix, 46
- CompareFile
 - SCOPSOWL_OPT_DATA, 69
 - SKUA_OPT_DATA, 77
- Conc_new
 - SHARK_DATA, 73
- Conc_old
 - SHARK_DATA, 73
- Console_Output
 - SHARK_DATA, 73
- const_Dc
 - skua.cpp, 139
 - skua.h, 141
- const_filmMassTransfer
 - scopsowl.cpp, 128
 - scopsowl.h, 130
- const_kf
 - skua.cpp, 139
 - skua.h, 141
- const_pH
 - SHARK_DATA, 73
- const_pore_diffusion
 - scopsowl.cpp, 128
 - scopsowl.h, 130
- constRho
 - BACKTRACK_DATA, 11
- ConstantICFill
 - Matrix, 46
- Contains_pH
 - SHARK_DATA, 73
- Contains_pOH
 - SHARK_DATA, 73
- Converged
 - SHARK_DATA, 73
- Convert2Concentration
 - shark.cpp, 134
 - shark.h, 137
- Convert2LogConcentration
 - shark.cpp, 134
 - shark.h, 137
- coord
 - SKUA_DATA, 76
- coord_macro
 - SCOPSOWL_DATA, 67
- coord_micro
 - SCOPSOWL_DATA, 67
- copyAnchor2Alias
 - Document, 16
 - Header, 37
 - YamlWrapper, 96
- count
 - UI_DATA, 87
- crystal_radius
 - SCOPSOWL_DATA, 67
- Cstd
 - egret.h, 101
- current_equil
 - SCOPSOWL_OPT_DATA, 69
 - SKUA_OPT_DATA, 77
- current_points
 - SCOPSOWL_OPT_DATA, 69
 - SKUA_OPT_DATA, 77
- current_press
 - SCOPSOWL_OPT_DATA, 69
 - SKUA_OPT_DATA, 77
- current_temp
 - SCOPSOWL_OPT_DATA, 69
 - SKUA_OPT_DATA, 77
- current_token
 - yaml_cpp_class, 94
- Cylindrical
 - finch.h, 108

- d
 - FINCH_DATA, [24](#)
- DAVIES
 - shark.h, [137](#)
- DAVIES_LADSHAW
 - shark.h, [137](#)
- DEBYE_HUCKEL
 - shark.h, [137](#)
- DOUBLE
 - yaml_wrapper.h, [153](#)
- D_c
 - skua.h, [141](#)
- D_ii
 - egret.h, [101](#)
- D_ij
 - egret.h, [101](#)
- D_inf
 - skua.h, [141](#)
- D_o
 - skua.h, [141](#)
- DBL_EPSILON
 - magpie.h, [122](#)
- dHo
 - GSTA_DATA, [34](#)
- DIC
 - FINCH_DATA, [24](#)
- DISPLACEMENT
 - Trajectory.cpp, [144](#)
 - Trajectory.h, [146](#)
- DOGFISH
 - dogfish.cpp, [97](#)
 - dogfish.h, [99](#)
- DOGFISH_DATA, [17](#)
 - DirichletBC, [18](#)
 - end_time, [18](#)
 - eval_DI, [18](#)
 - eval_R, [18](#)
 - eval_kf, [18](#)
 - eval_qs, [18](#)
 - fiber_diameter, [18](#)
 - fiber_length, [18](#)
 - finch_dat, [18](#)
 - NonLinear, [18](#)
 - NumComp, [18](#)
 - OutputFile, [18](#)
 - param_dat, [18](#)
 - Print2Console, [18](#)
 - Print2File, [18](#)
 - t_counter, [18](#)
 - t_print, [18](#)
 - time, [18](#)
 - time_old, [18](#)
 - total_sorption, [18](#)
 - total_sorption_old, [19](#)
 - total_steps, [19](#)
 - user_data, [19](#)
- DOGFISH_Executioner
 - dogfish.cpp, [98](#)
 - dogfish.h, [99](#)
- DOGFISH_PARAM, [19](#)
 - film_transfer_coeff, [19](#)
 - initial_sorption, [19](#)
 - intraparticle_diffusion, [19](#)
 - sorbed_molefraction, [19](#)
 - species, [19](#)
 - surface_concentration, [19](#)
- DOGFISH_TESTS
 - dogfish.cpp, [98](#)
 - dogfish.h, [99](#)
- DOGFISH_postprocesses
 - dogfish.cpp, [98](#)
 - dogfish.h, [99](#)
- DOGFISH_preprocesses
 - dogfish.cpp, [98](#)
 - dogfish.h, [99](#)
- DOGFISH_reset
 - dogfish.cpp, [98](#)
 - dogfish.h, [99](#)
- dSo
 - GSTA_DATA, [34](#)
- dX
 - TRAJECTORY_DATA, [85](#)
- dY
 - TRAJECTORY_DATA, [85](#)
- Data
 - Matrix, [48](#)
- Data_Map
 - SubHeader, [82](#)
- data_type
 - yaml_wrapper.h, [152](#)
- Davies_equation
 - shark.cpp, [134](#)
 - shark.h, [137](#)
- DaviesLadshaw_equation
 - shark.cpp, [134](#)
 - shark.h, [137](#)
- DebyeHuckel_equation
 - shark.cpp, [134](#)
 - shark.h, [137](#)
- default_Dc
 - skua.cpp, [139](#)
 - skua.h, [141](#)
- default_FilmMTCoeff
 - dogfish.cpp, [97](#)
 - dogfish.h, [99](#)
- default_IntraDiffusion
 - dogfish.cpp, [97](#)
 - dogfish.h, [99](#)
- default_Retardation
 - dogfish.cpp, [97](#)
 - dogfish.h, [99](#)
- default_SurfaceConcentration
 - dogfish.cpp, [97](#)
 - dogfish.h, [99](#)
- default_adsorption
 - scopsowl.cpp, [128](#)

- scopsowl.h, 131
- default_bcs
 - finch.cpp, 106
 - finch.h, 108
- default_density
 - monkfish.cpp, 125
 - monkfish.h, 126
- default_effective_diffusion
 - scopsowl.cpp, 128
 - scopsowl.h, 131
- default_execution
 - finch.cpp, 106
 - finch.h, 108
- default_exterior_concentration
 - monkfish.cpp, 125
 - monkfish.h, 126
- default_film_transfer
 - monkfish.cpp, 125
 - monkfish.h, 126
- default_filmMassTransfer
 - scopsowl.cpp, 128
 - scopsowl.h, 131
- default_ic
 - finch.cpp, 106
 - finch.h, 109
- default_interparticle_diffusion
 - monkfish.cpp, 125
 - monkfish.h, 126
- default_kf
 - skua.cpp, 139
 - skua.h, 141
- default_monk_adsorption
 - monkfish.cpp, 125
 - monkfish.h, 126
- default_monk_equilibrium
 - monkfish.cpp, 125
 - monkfish.h, 126
- default_monkfish_retardation
 - monkfish.cpp, 125
 - monkfish.h, 126
- default_params
 - finch.cpp, 106
 - finch.h, 109
- default_pore_diffusion
 - scopsowl.cpp, 128
 - scopsowl.h, 131
- default_porosity
 - monkfish.cpp, 125
 - monkfish.h, 126
- default_postprocess
 - finch.cpp, 106
 - finch.h, 109
- default_precon
 - finch.cpp, 106
 - finch.h, 109
- default_preprocess
 - finch.cpp, 106
 - finch.h, 109
- default_res
 - finch.cpp, 106
 - finch.h, 109
- default_reset
 - finch.cpp, 106
 - finch.h, 109
- default_retardation
 - scopsowl.cpp, 128
 - scopsowl.h, 131
- default_solve
 - finch.cpp, 106
 - finch.h, 109
- default_surf_diffusion
 - scopsowl.cpp, 128
 - scopsowl.h, 131
- default_timestep
 - finch.cpp, 106
 - finch.h, 109
- Delta
 - MassBalance, 42
- density
 - PURE_GAS, 63
- determinate
 - Matrix, 46
- diagonalSolve
 - Matrix, 46
- dielectric_const
 - SHARK_DATA, 73
- diffusion_type
 - SCOPSOWL_OPT_DATA, 69
 - SKUA_OPT_DATA, 77
- dim_mis_match
 - error.h, 104
- Dirichlet
 - FINCH_DATA, 24
- DirichletBC
 - DOGFISH_DATA, 18
 - MONKFISH_DATA, 54
 - SCOPSOWL_DATA, 67
 - SKUA_DATA, 76
- dirichletBCFill
 - Matrix, 46
- discretize
 - FINCH_DATA, 24
- Display
 - Matrix, 46
- Display_Info
 - MassBalance, 42
 - Reaction, 65
 - UnsteadyReaction, 90
- display_help
 - ui.cpp, 147
 - ui.h, 150
- display_version
 - ui.cpp, 147
 - ui.h, 150
- DisplayAll
 - MasterSpeciesList, 43

- DisplayConcentrations
 - MasterSpeciesList, [43](#)
- DisplayContents
 - Document, [16](#)
 - Header, [37](#)
 - SubHeader, [82](#)
 - yaml_cpp_class, [94](#)
 - YamlWrapper, [96](#)
- DisplayInfo
 - Atom, [9](#)
 - MasterSpeciesList, [44](#)
 - Molecule, [51](#)
- DisplayMap
 - KeyValueMap, [40](#)
- DisplayPair
 - ValueTypePair, [93](#)
- DisplayTable
 - PeriodicTable, [59](#)
- Dk
 - scopsowl.h, [130](#)
- Dn
 - FINCH_DATA, [24](#)
- Dnp1
 - FINCH_DATA, [24](#)
- Do
 - FINCH_DATA, [24](#)
- Doc_Map
 - YamlWrapper, [96](#)
- Document, [14](#)
 - ~Document, [16](#)
 - addHeadKey, [16](#)
 - addPair, [16](#)
 - begin, [16](#)
 - changeKey, [16](#)
 - clear, [16](#)
 - copyAnchor2Alias, [16](#)
 - DisplayContents, [16](#)
 - Document, [16](#)
 - end, [16](#)
 - getAlias, [16](#)
 - getAnchoredHeader, [16](#)
 - getDataMap, [16](#)
 - getHeadFromSubAlias, [16](#)
 - getHeadMap, [16](#)
 - getHeader, [16](#)
 - getName, [16](#)
 - getState, [16](#)
 - Head_Map, [17](#)
 - isAlias, [17](#)
 - isAnchor, [17](#)
 - operator(), [17](#)
 - operator=, [17](#)
 - resetKeys, [17](#)
 - revalidateAllKeys, [17](#)
 - setAlias, [17](#)
 - setName, [17](#)
 - setNameAliasPair, [17](#)
 - setState, [17](#)
 - size, [17](#)
- dog_dat
 - MONKFISH_DATA, [54](#)
- dogfish
 - ui.h, [150](#)
- dogfish.cpp, [97](#)
 - DOGFISH, [97](#)
 - DOGFISH_Executioner, [98](#)
 - DOGFISH_TESTS, [98](#)
 - DOGFISH_postprocesses, [98](#)
 - DOGFISH_preprocesses, [98](#)
 - DOGFISH_reset, [98](#)
 - default_FilmMTCoeff, [97](#)
 - default_IntraDiffusion, [97](#)
 - default_Retardation, [97](#)
 - default_SurfaceConcentration, [97](#)
 - print2file_DOGFISH_header, [98](#)
 - print2file_DOGFISH_result_new, [98](#)
 - print2file_DOGFISH_result_old, [98](#)
 - print2file_species_header, [98](#)
 - set_DOGFISH_ICs, [98](#)
 - set_DOGFISH_params, [98](#)
 - set_DOGFISH_timestep, [98](#)
 - setup_DOGFISH_DATA, [98](#)
- dogfish.h, [98](#)
 - DOGFISH, [99](#)
 - DOGFISH_Executioner, [99](#)
 - DOGFISH_TESTS, [99](#)
 - DOGFISH_postprocesses, [99](#)
 - DOGFISH_preprocesses, [99](#)
 - DOGFISH_reset, [99](#)
 - default_FilmMTCoeff, [99](#)
 - default_IntraDiffusion, [99](#)
 - default_Retardation, [99](#)
 - default_SurfaceConcentration, [99](#)
 - print2file_DOGFISH_header, [99](#)
 - print2file_DOGFISH_result_new, [99](#)
 - print2file_DOGFISH_result_old, [99](#)
 - print2file_species_header, [99](#)
 - set_DOGFISH_ICs, [99](#)
 - set_DOGFISH_params, [99](#)
 - set_DOGFISH_timestep, [99](#)
 - setup_DOGFISH_DATA, [99](#)
- domain_diameter
 - MONKFISH_DATA, [54](#)
- Dp
 - scopsowl.h, [130](#)
- Dp_ij
 - egret.h, [102](#)
- dq_dc
 - SCOPSOWL_PARAM_DATA, [71](#)
- dq_dco
 - SCOPSOWL_PARAM_DATA, [71](#)
- dq_dp
 - magpie.cpp, [121](#)
 - magpie.h, [123](#)
- dt
 - FINCH_DATA, [24](#)

- SHARK_DATA, 73
- TRAJECTORY_DATA, 85
- dt_min
 - SHARK_DATA, 73
- dt_old
 - FINCH_DATA, 24
- duplicate_variable
 - error.h, 104
- dxj
 - NUM_JAC_DATA, 57
- dynamic_viscosity
 - PURE_GAS, 63
- dz
 - FINCH_DATA, 24
- e0
 - GMRESRP_DATA, 32
- e0_bar
 - GMRESRP_DATA, 32
- e1
 - ARNOLDI_DATA, 7
- EXECUTE
 - ui.h, 150
- EXIT
 - ui.h, 150
- e_norm
 - SCOPSOWL_OPT_DATA, 69
 - SKUA_OPT_DATA, 78
- e_norm_old
 - SCOPSOWL_OPT_DATA, 69
 - SKUA_OPT_DATA, 78
- ECO_EXECUTABLE
 - ui.h, 150
- ECO_VERSION
 - ui.h, 150
- EEL_TESTS
 - eel.cpp, 100
 - eel.h, 100
- EGRET_TESTS
 - egret.cpp, 100
 - egret.h, 102
- eMax
 - magpie.cpp, 121
 - magpie.h, 123
 - mSPD_DATA, 56
- EX01_DATA, 19
 - b, 20
 - M, 20
- EX02_DATA, 20
 - b, 20
 - M, 20
- EX04_DATA, 20
 - b, 20
 - M, 20
- EX09_DATA, 21
 - h, 21
 - k, 21
 - M, 21
 - N, 21
 - p, 21
 - s, 21
 - x, 21
- EX15_DATA, 21
 - b, 21
 - m, 21
 - N, 21
- edit
 - Matrix, 46
- editAllOxidationStates
 - Molecule, 51
- editAtomicWeight
 - Atom, 9
- editCharge
 - Molecule, 51
- editElectrons
 - Atom, 9
- editEnergy
 - Molecule, 51
- editEnthalpy
 - Molecule, 51
- editEntropy
 - Molecule, 51
- editHS
 - Molecule, 51
- editNeutrons
 - Atom, 9
- editOneOxidationState
 - Molecule, 51
- editOxidationState
 - Atom, 9
- editPair
 - ValueTypePair, 93
- editProtons
 - Atom, 9
- editValence
 - Atom, 9
- editValue
 - ValueTypePair, 93
- editValue4Key
 - KeyValueMap, 40
- eduGuess
 - gsta_opt.cpp, 111
 - gsta_opt.h, 112
- eel
 - ui.h, 150
- eel.cpp, 99
 - EEL_TESTS, 100
- eel.h, 100
 - EEL_TESTS, 100
- egret
 - ui.h, 150
- egret.cpp, 100
 - calculate_properties, 100
 - EGRET_TESTS, 100
 - initialize_data, 101
 - set_variables, 101
- egret.h, 101

- CE3, 101
- calculate_properties, 102
- Cstd, 101
- D_{ii}, 101
- D_{ij}, 101
- Dp_{ij}, 102
- EGRET_TESTS, 102
- FilmMTCoeff, 102
- initialize_data, 102
- Mu, 102
- Nu, 102
- PE3, 102
- PSI, 102
- Po, 102
- Pstd, 102
- RE3, 102
- ReNum, 102
- Rstd, 102
- ScNum, 102
- set_variables, 102
- Electrons
 - Atom, 10
- electrons
 - Atom, 10
- empirical_kf
 - skua.cpp, 139
 - skua.h, 141
- empty_matrix
 - error.h, 104
- end
 - Document, 16
 - Header, 37
 - KeyValueMap, 40
 - YamlWrapper, 96
- end_time
 - DOGFISH_DATA, 18
 - MONKFISH_DATA, 54
- Energy
 - Molecule, 51
- energy
 - Reaction, 66
- Enthalpy
 - Molecule, 51
- enthalpy
 - Reaction, 66
- Entropy
 - Molecule, 51
- entropy
 - Reaction, 66
- eps
 - NUM_JAC_DATA, 57
 - PJFNK_DATA, 61
- Equilibrium
 - Reaction, 66
- error
 - error.cpp, 102
 - error.h, 104
 - gsta_opt.h, 112
- error.h
 - anchor_alias_dne, 104
 - arg_matrix_same, 104
 - dim_mis_match, 104
 - duplicate_variable, 104
 - empty_matrix, 104
 - file_dne, 103
 - generic_error, 103
 - indexing_error, 103
 - initial_error, 104
 - invalid_atom, 104
 - invalid_boolean, 103
 - invalid_components, 103
 - invalid_console_input, 104
 - invalid_electron, 104
 - invalid_fraction, 104
 - invalid_gas_sum, 103
 - invalid_molefraction, 103
 - invalid_neutron, 104
 - invalid_norm, 104
 - invalid_proton, 104
 - invalid_size, 104
 - invalid_solid_sum, 103
 - invalid_species, 104
 - invalid_type, 104
 - invalid_valence, 104
 - key_not_found, 104
 - magpie_reverse_error, 103
 - matrix_too_small, 104
 - matvec_mis_match, 104
 - missing_information, 104
 - negative_mass, 104
 - negative_time, 104
 - no_diffusion, 104
 - non_real_edge, 104
 - non_square_matrix, 103
 - not_a_token, 104
 - nullptr_error, 104
 - nullptr_func, 104
 - opt_no_support, 104
 - ortho_check_fail, 104
 - out_of_bounds, 103
 - read_error, 104
 - rxn_rate_error, 104
 - scenario_fail, 103
 - simulation_fail, 103
 - singular_matrix, 104
 - string_parse_error, 104
 - tensor_out_of_bounds, 104
 - unregistered_name, 104
 - unstable_matrix, 104
 - vector_out_of_bounds, 104
 - zero_vector, 104
- error.cpp, 102
- error, 102
- error.h, 102
 - error, 104
 - error_type, 103

- mError, 103
- error_type
 - error.h, 103
- eta
 - mSPD_DATA, 56
 - TRAJECTORY_DATA, 85
- eval_Cex
 - MONKFISH_DATA, 54
- Eval_ChargeResidual
 - MasterSpeciesList, 44
- eval_DI
 - DOGFISH_DATA, 18
- eval_Dex
 - MONKFISH_DATA, 54
- eval_GPAST
 - magpie.cpp, 121
 - magpie.h, 123
- eval_GSTA
 - gsta_opt.cpp, 111
 - gsta_opt.h, 112
- Eval_IC_Residual
 - UnsteadyReaction, 90
- eval_R
 - DOGFISH_DATA, 18
- Eval_ReactionRate
 - UnsteadyReaction, 90
- Eval_Residual
 - MassBalance, 42
 - Reaction, 65
 - UnsteadyReaction, 90
- eval_Ret
 - MONKFISH_DATA, 54
- eval_SCOPSOWL_Uptake
 - scopsowl_opt.cpp, 132
 - scopsowl_opt.h, 132
- eval_SKUA_Uptake
 - skua_opt.cpp, 143
 - skua_opt.h, 143
- eval_ads
 - MONKFISH_DATA, 54
 - SCOPSOWL_DATA, 67
- eval_diff
 - SCOPSOWL_DATA, 67
 - SKUA_DATA, 76
- eval_eps
 - MONKFISH_DATA, 54
- eval_eta
 - magpie.cpp, 121
 - magpie.h, 123
- eval_kf
 - DOGFISH_DATA, 18
 - MONKFISH_DATA, 54
 - SCOPSOWL_DATA, 67
 - SKUA_DATA, 76
- eval_po
 - magpie.cpp, 121
 - magpie.h, 123
- eval_po_PI
 - magpie.cpp, 121
 - magpie.h, 123
- eval_po_qo
 - magpie.cpp, 121
 - magpie.h, 123
- eval_qs
 - DOGFISH_DATA, 18
- eval_retard
 - SCOPSOWL_DATA, 67
- eval_rho
 - MONKFISH_DATA, 54
- eval_surfDiff
 - SCOPSOWL_DATA, 67
- EvalActivity
 - SHARK_DATA, 74
- evalprecon
 - FINCH_DATA, 24
- evalres
 - FINCH_DATA, 24
- evaluation
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- evalx_ex09
 - lark.cpp, 114
 - lark.h, 118
- exec
 - ui.cpp, 147
 - ui.h, 150
- exec_loop
 - ui.cpp, 148
 - ui.h, 150
- exec_option
 - ui.h, 151
- executeYamlRead
 - yaml_cpp_class, 94
- exit
 - ui.cpp, 148
 - ui.h, 151
- Explicit_Eval
 - UnsteadyReaction, 90
- ExplicitFlux
 - FINCH_DATA, 24
- exterior_concentration
 - MONKFISH_PARAM, 55
- exterior_transfer_coeff
 - MONKFISH_PARAM, 55
- F
 - PJFNK_DATA, 61
- FOM
 - lark.h, 117
- f_bias
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- f_bias_old
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- fC_E
 - FINCH_DATA, 24

- fc_I
 - FINCH_DATA, 25
- FINCH_DATA, 22
 - beta, 24
 - CC_E, 24
 - CC_I, 24
 - CL_E, 24
 - CL_I, 24
 - CN, 24
 - CR_E, 24
 - CR_I, 24
 - callroutine, 24
 - CheckMass, 24
 - d, 24
 - DIC, 24
 - Dirichlet, 24
 - discretize, 24
 - Dn, 24
 - Dnp1, 24
 - Do, 24
 - dt, 24
 - dt_old, 24
 - dz, 24
 - evalprecon, 24
 - evalres, 24
 - ExplicitFlux, 24
 - fc_E, 24
 - fc_I, 25
 - fl_E, 25
 - fl_I, 25
 - fR_E, 25
 - fR_I, 25
 - Fn, 25
 - Fnp1, 25
 - gE, 25
 - gl, 25
 - Iterative, 25
 - kIC, 25
 - kfn, 25
 - kfnp1, 25
 - kn, 25
 - knp1, 25
 - ko, 25
 - L, 25
 - LN, 25
 - lambda_E, 25
 - lambda_I, 25
 - ME, 25
 - MI, 25
 - max_iter, 25
 - NE, 25
 - NI, 25
 - nl_method, 25
 - NormTrack, 25
 - OE, 25
 - OI, 26
 - param_data, 26
 - picard_dat, 26
 - pjfnk_dat, 26
 - pres, 26
 - RIC, 26
 - res, 26
 - resetime, 26
 - Rn, 26
 - Rnp1, 26
 - Ro, 26
 - s, 26
 - setbcs, 26
 - setic, 26
 - setparams, 26
 - setpostprocess, 26
 - setpreprocess, 26
 - settime, 26
 - Sn, 26
 - Snp1, 26
 - solve, 26
 - SteadyState, 26
 - T, 26
 - t, 26
 - t_old, 26
 - tol_abs, 26
 - tol_rel, 26
 - total_iter, 26
 - u_star, 27
 - uAvg, 27
 - uAvg_old, 27
 - uIC, 27
 - uT, 27
 - uT_old, 27
 - ubest, 27
 - un, 27
 - unm1, 27
 - unp1, 27
 - uo, 27
 - Update, 27
 - uz_I_E, 27
 - uz_I_I, 27
 - uz_lm1_E, 27
 - uz_lm1_I, 27
 - uz_lp1_E, 27
 - uz_lp1_I, 27
 - vIC, 27
 - vn, 27
 - vnp1, 27
 - vo, 27
- FINCH_Picard
 - finch.h, 108
- FINCH_TESTS
 - finch.cpp, 106
 - finch.h, 109
- fl_E
 - FINCH_DATA, 25
- fl_I
 - FINCH_DATA, 25
- fR_E
 - FINCH_DATA, 25

- fR_I
 - FINCH_DATA, 25
- fiber_diameter
 - DOGFISH_DATA, 18
- fiber_length
 - DOGFISH_DATA, 18
- file_dne
 - error.h, 103
- File_Output
 - SHARK_DATA, 74
- file_name
 - yaml_cpp_class, 94
- Files
 - UI_DATA, 87
- film_transfer
 - SCOPSOWL_PARAM_DATA, 71
 - SKUA_PARAM, 79
- film_transfer_coeff
 - DOGFISH_PARAM, 19
 - MONKFISH_PARAM, 55
- FilmMTCoeff
 - egret.h, 102
- finch
 - ui.h, 150
- finch.cpp, 104
 - buckley_leverett_ic, 105
 - buckley_leverett_params, 105
 - burgers_bcs, 105
 - burgers_ic, 105
 - burgers_params, 105
 - check_Mass, 106
 - default_bcs, 106
 - default_execution, 106
 - default_ic, 106
 - default_params, 106
 - default_postprocess, 106
 - default_precon, 106
 - default_preprocess, 106
 - default_res, 106
 - default_reset, 106
 - default_solve, 106
 - default_timestep, 106
 - FINCH_TESTS, 106
 - I_direct, 106
 - lark_picard_step, 106
 - max, 106
 - min, 106
 - minmod, 106
 - minmod_discretization, 106
 - nl_picard, 106
 - ospre_discretization, 106
 - print2file_dim_header, 106
 - print2file_newline, 106
 - print2file_result_new, 106
 - print2file_result_old, 106
 - print2file_tab, 106
 - print2file_time_header, 106
 - setup_FINCH_DATA, 106
 - uAverage, 107
 - uTotal, 107
 - vanAlbada_discretization, 107
- finch.h, 107
 - buckley_leverett_ic, 108
 - buckley_leverett_params, 108
 - burgers_bcs, 108
 - burgers_ic, 108
 - burgers_params, 108
 - Cartesian, 108
 - check_Mass, 108
 - Cylindrical, 108
 - default_bcs, 108
 - default_execution, 108
 - default_ic, 109
 - default_params, 109
 - default_postprocess, 109
 - default_precon, 109
 - default_preprocess, 109
 - default_res, 109
 - default_reset, 109
 - default_solve, 109
 - default_timestep, 109
 - FINCH_Picard, 108
 - FINCH_TESTS, 109
 - I_direct, 109
 - LARK_PJFNK, 108
 - LARK_Picard, 108
 - lark_picard_step, 109
 - max, 109
 - min, 109
 - minmod, 109
 - minmod_discretization, 109
 - nl_picard, 109
 - ospre_discretization, 109
 - print2file_dim_header, 109
 - print2file_newline, 109
 - print2file_result_new, 109
 - print2file_result_old, 109
 - print2file_tab, 109
 - print2file_time_header, 109
 - setup_FINCH_DATA, 109
 - Spherical, 108
 - uAverage, 110
 - uTotal, 110
 - vanAlbada_discretization, 110
- finch_dat
 - DOGFISH_DATA, 18
 - MONKFISH_DATA, 54
 - SCOPSOWL_DATA, 67
 - SKUA_DATA, 76
- findAllTypes
 - KeyValueMap, 40
- findType
 - KeyValueMap, 40
 - ValueTypePair, 93
- Fk
 - BACKTRACK_DATA, 11

flock.h, 110
Fn
 FINCH_DATA, 25
Fnp1
 FINCH_DATA, 25
Fobj
 GSTA_OPT_DATA, 35
fom
 lark.cpp, 114
 lark.h, 118
formation_energy
 Molecule, 52
formation_enthalpy
 Molecule, 52
formation_entropy
 Molecule, 52
Formula
 Molecule, 52
forward_rate
 UnsteadyReaction, 91
forward_ref_rate
 UnsteadyReaction, 91
funeval
 PJFNK_DATA, 61
funeval_ex09
 lark.cpp, 114
 lark.h, 118
funeval_ex10
 lark.cpp, 114
 lark.h, 118
Fv
 PJFNK_DATA, 61
Fx
 NUM_JAC_DATA, 57
Fxp
 NUM_JAC_DATA, 57

GCR
 lark.h, 117
GMRESLP
 lark.h, 117
GMRESR
 lark.h, 117
GMRESRP
 lark.h, 117
GCR_DATA, 27
 alpha, 28
 bestres, 28
 bestx, 28
 beta, 28
 breakdown, 28
 c, 28
 c_temp, 28
 iter_inner, 28
 iter_outer, 28
 maxit, 28
 Output, 28
 r, 28
 relres, 28
 relres_base, 28
 res, 28
 restart, 29
 tol_abs, 29
 tol_rel, 29
 total_iter, 29
 transpose_dat, 29
 u, 29
 u_temp, 29
 x, 29
GCR_Output
 GMRESR_DATA, 31
gE
 FINCH_DATA, 25
gl
 FINCH_DATA, 25
GMRES_Output
 GMRESR_DATA, 31
GMRESLP_DATA, 29
 arnoldi_dat, 29
 bestres, 29
 bestx, 29
 iter, 29
 maxit, 29
 Output, 30
 r, 30
 relres, 30
 relres_base, 30
 res, 30
 restart, 30
 steps, 30
 tol_abs, 30
 tol_rel, 30
 x, 30
GMRESR_DATA, 30
 arg, 31
 GCR_Output, 31
 GMRES_Output, 31
 gcr_abs_tol, 31
 gcr_dat, 31
 gcr_maxit, 31
 gcr_rel_tol, 31
 gcr_restart, 31
 gmres_dat, 31
 gmres_maxit, 31
 gmres_restart, 31
 gmres_tol, 31
 iter_inner, 31
 iter_outer, 31
 matvec, 31
 matvec_data, 31
 N, 31
 term_precon, 31
 terminal_precon, 31
 total_iter, 31
GMRESRP_DATA, 31
 bestres, 32
 bestx, 32

- e0, [32](#)
- e0_bar, [32](#)
- H, [32](#)
- H_bar, [32](#)
- iter_inner, [32](#)
- iter_outer, [32](#)
- iter_total, [32](#)
- maxit, [32](#)
- Output, [32](#)
- r, [32](#)
- relres, [32](#)
- relres_base, [32](#)
- res, [32](#)
- restart, [33](#)
- sum, [33](#)
- tol_abs, [33](#)
- tol_rel, [33](#)
- v, [33](#)
- Vk, [33](#)
- w, [33](#)
- x, [33](#)
- y, [33](#)
- GPAST_DATA, [33](#)
 - gama_inf, [33](#)
 - He, [33](#)
 - Plo, [33](#)
 - po, [33](#)
 - poi, [33](#)
 - present, [33](#)
 - q, [34](#)
 - qo, [34](#)
 - x, [34](#)
 - y, [34](#)
- GSTA_DATA, [34](#)
 - dHo, [34](#)
 - dSo, [34](#)
 - m, [34](#)
 - qmax, [34](#)
- GSTA_OPT_DATA, [34](#)
 - all_pars, [35](#)
 - best_par, [35](#)
 - Fobj, [35](#)
 - iso, [35](#)
 - Kno, [35](#)
 - n_par, [35](#)
 - norms, [35](#)
 - opt_qmax, [35](#)
 - P, [35](#)
 - q, [35](#)
 - qmax, [35](#)
 - total_eval, [35](#)
- gama
 - mSPD_DATA, [56](#)
- gama_inf
 - GPAST_DATA, [33](#)
- gas_dat
 - SCOPSOWL_DATA, [67](#)
 - SKUA_DATA, [76](#)
- gas_temperature
 - MIXED_GAS, [49](#)
 - SCOPSOWL_DATA, [67](#)
- gas_velocity
 - SCOPSOWL_DATA, [67](#)
 - SKUA_DATA, [76](#)
- gcr
 - lark.cpp, [114](#)
 - lark.h, [118](#)
- gcr_abs_tol
 - GMRESR_DATA, [31](#)
- gcr_dat
 - GMRESR_DATA, [31](#)
 - PJFNK_DATA, [62](#)
- gcr_maxit
 - GMRESR_DATA, [31](#)
- gcr_rel_tol
 - GMRESR_DATA, [31](#)
- gcr_restart
 - GMRESR_DATA, [31](#)
- generic_error
 - error.h, [103](#)
- Get_ActivationEnergy
 - UnsteadyReaction, [90](#)
- Get_Affinity
 - UnsteadyReaction, [90](#)
- Get_Delta
 - MassBalance, [42](#)
- Get_Energy
 - Reaction, [65](#)
 - UnsteadyReaction, [90](#)
- Get_Enthalpy
 - Reaction, [65](#)
 - UnsteadyReaction, [90](#)
- Get_Entropy
 - Reaction, [65](#)
 - UnsteadyReaction, [90](#)
- Get_Equilibrium
 - Reaction, [65](#)
 - UnsteadyReaction, [90](#)
- Get_Forward
 - UnsteadyReaction, [90](#)
- Get_ForwardRef
 - UnsteadyReaction, [90](#)
- Get_InitialValue
 - UnsteadyReaction, [90](#)
- Get_MaximumValue
 - UnsteadyReaction, [90](#)
- Get_Name
 - MassBalance, [42](#)
- Get_Reverse
 - UnsteadyReaction, [90](#)
- Get_ReverseRef
 - UnsteadyReaction, [90](#)
- Get_Species_Index
 - UnsteadyReaction, [90](#)
- Get_Stoichiometric
 - Reaction, [65](#)

- UnsteadyReaction, 90
- Get_TimeStep
 - UnsteadyReaction, 90
- Get_TotalConcentration
 - MassBalance, 42
- get_index
 - MasterSpeciesList, 44
- get_species
 - MasterSpeciesList, 44
- getAlias
 - Document, 16
 - Header, 37
 - SubHeader, 82
- getAnchoredDoc
 - YamlWrapper, 96
- getAnchoredHeader
 - Document, 16
- getAnchoredSub
 - Header, 37
- getBool
 - KeyValueMap, 40
 - ValueTypePair, 93
- getDataMap
 - Document, 16
 - Header, 37
- getDocFromHeadAlias
 - YamlWrapper, 96
- getDocFromSubAlias
 - YamlWrapper, 96
- getDocMap
 - YamlWrapper, 96
- getDocument
 - YamlWrapper, 96
- getDouble
 - KeyValueMap, 40
 - ValueTypePair, 93
- getHeadFromSubAlias
 - Document, 16
- getHeadMap
 - Document, 16
- getHeader
 - Document, 16
- getInt
 - KeyValueMap, 40
 - ValueTypePair, 93
- getMap
 - KeyValueMap, 40
 - SubHeader, 82
- getName
 - Document, 16
 - Header, 37
 - SubHeader, 82
- getPair
 - KeyValueMap, 40
 - ValueTypePair, 93
- getState
 - Document, 16
 - Header, 37
- SubHeader, 82
- getString
 - KeyValueMap, 40
 - ValueTypePair, 93
- getSubHeader
 - Header, 37
- getSubMap
 - Header, 37
- getType
 - KeyValueMap, 40
 - ValueTypePair, 93
- getValue
 - KeyValueMap, 40
 - ValueTypePair, 93
- getYamlWrapper
 - yaml_cpp_class, 94
- gmres_dat
 - GMRESR_DATA, 31
- gmres_maxit
 - GMRESR_DATA, 31
- gmres_restart
 - GMRESR_DATA, 31
- gmres_tol
 - GMRESR_DATA, 31
- gmresLeftPreconditioned
 - lark.cpp, 115
 - lark.h, 118
- gmresPreconditioner
 - lark.cpp, 115
 - lark.h, 118
- gmresRightPreconditioned
 - lark.cpp, 115
 - lark.h, 118
- gmreslp_dat
 - PJFNK_DATA, 62
- gmresr
 - lark.cpp, 115
 - lark.h, 118
- gmresr_dat
 - PJFNK_DATA, 62
- gmresrp_dat
 - PJFNK_DATA, 62
- gpast_dat
 - MAGPIE_DATA, 41
- grad_mSPD
 - magpie.cpp, 121
 - magpie.h, 123
- Grav_R
 - Trajectory.cpp, 144
 - Trajectory.h, 146
- Grav_T
 - Trajectory.cpp, 144
 - Trajectory.h, 146
- gsta_opt
 - ui.h, 150
- gsta_dat
 - MAGPIE_DATA, 41
- gsta_opt.cpp, 110

- avgPar, [111](#)
- avgValue, [111](#)
- eduGuess, [111](#)
- eval_GSTA, [111](#)
- gsta_optimize, [111](#)
- gstaFunc, [111](#)
- gstaObjFunc, [111](#)
- isSmooth, [111](#)
- minIndex, [111](#)
- minValue, [111](#)
- orderMag, [111](#)
- orthoLinReg, [111](#)
- rSq, [111](#)
- roundIt, [111](#)
- twoFifths, [111](#)
- weightedAvg, [111](#)
- gsta_opt.h, [111](#)
 - avgPar, [112](#)
 - avgValue, [112](#)
 - eduGuess, [112](#)
 - error, [112](#)
 - eval_GSTA, [112](#)
 - gsta_optimize, [112](#)
 - gstaFunc, [112](#)
 - gstaObjFunc, [112](#)
 - isSmooth, [112](#)
 - minIndex, [112](#)
 - minValue, [113](#)
 - Na, [112](#)
 - orderMag, [113](#)
 - orthoLinReg, [113](#)
 - Po, [112](#)
 - R, [112](#)
 - rSq, [113](#)
 - roundIt, [113](#)
 - twoFifths, [113](#)
 - weightedAvg, [113](#)
- gsta_optimize
 - gsta_opt.cpp, [111](#)
 - gsta_opt.h, [112](#)
- gstaFunc
 - gsta_opt.cpp, [111](#)
 - gsta_opt.h, [112](#)
- gstaObjFunc
 - gsta_opt.cpp, [111](#)
 - gsta_opt.h, [112](#)
- H
 - GMRESRP_DATA, [32](#)
 - TRAJECTORY_DATA, [85](#)
- h
 - EX09_DATA, [21](#)
- H0
 - TRAJECTORY_DATA, [85](#)
- HELP
 - ui.h, [150](#)
- H_bar
 - GMRESRP_DATA, [32](#)
- Hamaker
 - TRAJECTORY_DATA, [85](#)
- HaveEnergy
 - Molecule, [52](#)
- HaveEquil
 - Reaction, [66](#)
- haveEquilibrium
 - Reaction, [65](#)
 - UnsteadyReaction, [90](#)
- HaveForRef
 - UnsteadyReaction, [91](#)
- HaveForward
 - UnsteadyReaction, [91](#)
- HaveG
 - Reaction, [66](#)
- haveG
 - Molecule, [52](#)
- HaveHS
 - Molecule, [52](#)
 - Reaction, [66](#)
- haveHS
 - Molecule, [52](#)
- haveMinMax
 - MONKFISH_DATA, [54](#)
- haveRate
 - UnsteadyReaction, [90](#)
- HaveRevRef
 - UnsteadyReaction, [91](#)
- HaveReverse
 - UnsteadyReaction, [91](#)
- He
 - GPAST_DATA, [33](#)
 - magpie.h, [122](#)
- Head_Map
 - Document, [17](#)
- Header, [35](#)
 - ~Header, [37](#)
 - addPair, [37](#)
 - addSubKey, [37](#)
 - begin, [37](#)
 - changeKey, [37](#)
 - clear, [37](#)
 - copyAnchor2Alias, [37](#)
 - DisplayContents, [37](#)
 - end, [37](#)
 - getAlias, [37](#)
 - getAnchoredSub, [37](#)
 - getDataMap, [37](#)
 - getName, [37](#)
 - getState, [37](#)
 - getSubHeader, [37](#)
 - getSubMap, [37](#)
 - Header, [37](#)
 - isAlias, [37](#)
 - isAnchor, [38](#)
 - operator(), [38](#)
 - operator=, [38](#)
 - resetKeys, [38](#)
 - setAlias, [38](#)

- setName, 38
- setNameAliasPair, 38
- setState, 38
- size, 38
- Sub_Map, 38
- header_state
 - yaml_wrapper.h, 152, 153
- help
 - ui.cpp, 148
 - ui.h, 151
- Heterogeneous
 - SCOPSOWL_DATA, 68
- Hkp1
 - ARNOLDI_DATA, 7
- hp1
 - ARNOLDI_DATA, 7
- I
 - SYSTEM_DATA, 83
- IDEAL
 - shark.h, 137
- INT
 - yaml_wrapper.h, 153
- Ideal
 - SYSTEM_DATA, 83
- ideal_solution
 - shark.cpp, 134
 - shark.h, 137
- li
 - OPTRANS_DATA, 57
- indexing_error
 - error.h, 103
- initial_error
 - error.h, 104
- initial_guess_SCOPSOWL
 - scopsowl_opt.cpp, 132
 - scopsowl_opt.h, 132
- initial_guess_SKUA
 - skua_opt.cpp, 143
 - skua_opt.h, 143
- initial_sorption
 - DOGFISH_PARAM, 19
 - MONKFISH_PARAM, 55
- initial_value
 - UnsteadyReaction, 91
- initialGuess_mSPD
 - magpie.cpp, 121
 - magpie.h, 123
- Initialize_List
 - MassBalance, 42
 - Reaction, 65
 - UnsteadyReaction, 90
- initialize_data
 - egret.cpp, 101
 - egret.h, 102
- inner_product
 - Matrix, 46
- input
 - ui.cpp, 148
 - ui.h, 151
- input_file
 - yaml_cpp_class, 94
- input_files
 - UI_DATA, 87
- IntegralAvg
 - Matrix, 46
- IntegralTotal
 - Matrix, 46
- interparticle_diffusion
 - MONKFISH_PARAM, 55
- intraparticle_diffusion
 - DOGFISH_PARAM, 19
 - MONKFISH_PARAM, 55
- invalid_atom
 - error.h, 104
- invalid_boolean
 - error.h, 103
- invalid_components
 - error.h, 103
- invalid_console_input
 - error.h, 104
- invalid_electron
 - error.h, 104
- invalid_fraction
 - error.h, 104
- invalid_gas_sum
 - error.h, 103
- invalid_molefraction
 - error.h, 103
- invalid_neutron
 - error.h, 104
- invalid_norm
 - error.h, 104
- invalid_proton
 - error.h, 104
- invalid_size
 - error.h, 104
- invalid_solid_sum
 - error.h, 103
- invalid_species
 - error.h, 104
- invalid_type
 - error.h, 104
- invalid_valence
 - error.h, 104
- invalid_input
 - ui.cpp, 148
 - ui.h, 151
- inverse
 - Matrix, 46
- isAlias
 - Document, 17
 - Header, 37
 - SubHeader, 82
- isAnchor
 - Document, 17
 - Header, 38

- SubHeader, 82
- isRegistered
 - Molecule, 52
- isSmooth
 - gsta_opt.cpp, 111
 - gsta_opt.h, 112
- iso
 - GSTA_OPT_DATA, 35
- iter
 - ARNOLDI_DATA, 7
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 14
 - GMRESLP_DATA, 29
 - PCG_DATA, 58
 - PICARD_DATA, 60
- iter_inner
 - GCR_DATA, 28
 - GMRESR_DATA, 31
 - GMRESRP_DATA, 32
- iter_outer
 - GCR_DATA, 28
 - GMRESR_DATA, 31
 - GMRESRP_DATA, 32
- iter_total
 - GMRESRP_DATA, 32
- Iterative
 - FINCH_DATA, 25
- J
 - SYSTEM_DATA, 83
- Jacobian
 - Speciation_Test01_Data, 80
- jacvec
 - lark.cpp, 115
 - lark.h, 118
- K
 - SYSTEM_DATA, 83
- k
 - ARNOLDI_DATA, 7
 - EX09_DATA, 21
 - TRAJECTORY_DATA, 85
- kB
 - magpie.h, 122
- kIC
 - FINCH_DATA, 25
- key_not_found
 - error.h, 104
- Key_Value
 - KeyValueMap, 40
- KeyValueMap, 38
 - ~KeyValueMap, 39
 - addKey, 39
 - addPair, 39
 - assertType, 39
 - begin, 40
 - clear, 40
 - DisplayMap, 40
 - editValue4Key, 40
 - end, 40
 - findAllTypes, 40
 - findType, 40
 - getBool, 40
 - getDouble, 40
 - getInt, 40
 - getMap, 40
 - getPair, 40
 - getString, 40
 - getType, 40
 - getValue, 40
 - Key_Value, 40
 - KeyValueMap, 39
 - KeyValueMap, 39
 - operator=, 40
 - size, 40
- kfn
 - FINCH_DATA, 25
- kfnp1
 - FINCH_DATA, 25
- kinematic_viscosity
 - MIXED_GAS, 49
- kn
 - FINCH_DATA, 25
- Kno
 - GSTA_OPT_DATA, 35
- knp1
 - FINCH_DATA, 25
- ko
 - FINCH_DATA, 25
- krylov_method
 - lark.h, 117
- L
 - FINCH_DATA, 25
 - TRAJECTORY_DATA, 85
- L_Output
 - PJFNK_DATA, 62
- L_direct
 - finch.cpp, 106
 - finch.h, 109
- L_iter
 - PJFNK_DATA, 62
- L_wire
 - TRAJECTORY_DATA, 85
- LARGE_CYCLE_TEST01
 - scopsowl.cpp, 128
 - scopsowl.h, 131
- LARK_PJFNK
 - finch.h, 108
- LARK_Picard
 - finch.h, 108
- LARK_TESTS
 - lark.cpp, 115
 - lark.h, 118
- LN
 - FINCH_DATA, 25
- LOCATION
 - Trajectory.cpp, 144

- Trajectory.h, 146
- ladshawSolve
 - Matrix, 46
- lambda_E
 - FINCH_DATA, 25
- lambda_I
 - FINCH_DATA, 25
- lambdaMin
 - BACKTRACK_DATA, 11
- lark
 - ui.h, 150
- lark.h
 - BiCGSTAB, 117
 - CGS, 117
 - FOM, 117
 - GCR, 117
 - GMRESLP, 117
 - GMRESR, 117
 - GMRESRP, 117
 - PCG, 117
- lark.cpp, 113
 - arnoldi, 114
 - backtrackLineSearch, 114
 - bicgstab, 114
 - cgs, 114
 - evalx_ex09, 114
 - fom, 114
 - funeval_ex09, 114
 - funeval_ex10, 114
 - gcr, 114
 - gmresLeftPreconditioned, 115
 - gmresPreconditioner, 115
 - gmresRightPreconditioned, 115
 - gmresr, 115
 - jacvec, 115
 - LARK_TESTS, 115
 - matvec_ex01, 115
 - matvec_ex02, 115
 - matvec_ex04, 115
 - matvec_ex15, 115
 - NumericalJacobian, 115
 - operatorTranspose, 115
 - pcg, 115
 - picard, 115
 - pjfnk, 115
 - precon_ex01, 115
 - precon_ex04, 115
 - precon_ex10, 115
 - precon_ex15, 115
 - update_arnoldi_solution, 115
- lark.h, 116
 - arnoldi, 118
 - backtrackLineSearch, 118
 - bicgstab, 118
 - cgs, 118
 - evalx_ex09, 118
 - fom, 118
 - funeval_ex09, 118
 - funeval_ex10, 118
 - gcr, 118
 - gmresLeftPreconditioned, 118
 - gmresPreconditioner, 118
 - gmresRightPreconditioned, 118
 - gmresr, 118
 - jacvec, 118
 - krylov_method, 117
 - LARK_TESTS, 118
 - matvec_ex01, 118
 - matvec_ex02, 118
 - matvec_ex04, 118
 - matvec_ex15, 118
 - NumericalJacobian, 119
 - operatorTranspose, 119
 - pcg, 119
 - picard, 119
 - pjfnk, 119
 - precon_ex01, 119
 - precon_ex04, 119
 - precon_ex10, 119
 - precon_ex15, 119
 - update_arnoldi_solution, 119
- lark_picard_step
 - finch.cpp, 106
 - finch.h, 109
- level
 - MONKFISH_DATA, 54
 - SCOPSOWL_DATA, 68
- lin_precon
 - SHARK_DATA, 74
- lin_tol_abs
 - PJFNK_DATA, 62
- lin_tol_rel
 - PJFNK_DATA, 62
- LineSearch
 - PJFNK_DATA, 62
- linear_solver
 - PJFNK_DATA, 62
- linearsolve_choice
 - shark.cpp, 134
 - shark.h, 137
- linsearch_choice
 - shark.cpp, 134
 - shark.h, 137
- List
 - MassBalance, 42
 - Mechanism, 48
 - Reaction, 66
- list_size
 - MasterSpeciesList, 44
- InKo
 - magpie.h, 122
- Inact_mSPD
 - magpie.cpp, 121
 - magpie.h, 123
- logC
 - Speciation_Test01_Data, 80

- logKa1
 - Speciation_Test01_Data, 80
- logKa2
 - Speciation_Test01_Data, 80
- logKw
 - Speciation_Test01_Data, 80
- lowerHessenberg2Triangular
 - Matrix, 46
- lowerHessenbergSolve
 - Matrix, 46
- lowerTriangularSolve
 - Matrix, 47
- M
 - EX01_DATA, 20
 - EX02_DATA, 20
 - EX04_DATA, 20
 - EX09_DATA, 21
 - TRAJECTORY_DATA, 85
- m
 - EX15_DATA, 21
 - GSTA_DATA, 34
- M_PI
 - macaw.h, 120
- m_rand
 - TRAJECTORY_DATA, 85
- MACAW_TESTS
 - macaw.cpp, 119
 - macaw.h, 120
- MAGPIE
 - magpie.cpp, 121
 - magpie.h, 123
- MAGPIE_DATA, 41
 - gpast_dat, 41
 - gsta_dat, 41
 - mspd_dat, 41
 - sys_dat, 41
- MAGPIE_SCENARIOS
 - magpie.cpp, 121
 - magpie.h, 123
- ME
 - FINCH_DATA, 25
- mError
 - error.h, 103
- MI
 - FINCH_DATA, 25
- MIXED_GAS, 48
 - binary_diffusion, 49
 - char_length, 49
 - CheckMolefractions, 49
 - gas_temperature, 49
 - kinematic_viscosity, 49
 - molefraction, 49
 - N, 49
 - Reynolds, 49
 - species_dat, 49
 - total_density, 49
 - total_dyn_vis, 49
 - total_molecular_weight, 49
 - total_pressure, 49
 - total_specific_heat, 49
 - velocity, 49
- MOLA_TESTS
 - mola.cpp, 124
 - mola.h, 124
- MONKFISH_DATA, 53
 - avg_fiber_density, 54
 - DirichletBC, 54
 - dog_dat, 54
 - domain_diameter, 54
 - end_time, 54
 - eval_Cex, 54
 - eval_Dex, 54
 - eval_Ret, 54
 - eval_ads, 54
 - eval_eps, 54
 - eval_kf, 54
 - eval_rho, 54
 - finch_dat, 54
 - haveMinMax, 54
 - level, 54
 - max_fiber_density, 54
 - max_porosity, 54
 - min_fiber_density, 54
 - min_porosity, 54
 - MultiScale, 54
 - NonLinear, 54
 - NumComp, 54
 - Output, 54
 - param_dat, 54
 - Print2Console, 54
 - Print2File, 54
 - single_fiber_density, 54
 - t_counter, 54
 - t_print, 55
 - time, 55
 - time_old, 55
 - total_sorption, 55
 - total_sorption_old, 55
 - total_steps, 55
 - user_data, 55
- MONKFISH_PARAM, 55
 - avg_sorption, 55
 - avg_sorption_old, 55
 - exterior_concentration, 55
 - exterior_transfer_coeff, 55
 - film_transfer_coeff, 55
 - initial_sorption, 55
 - interparticle_diffusion, 55
 - intraparticle_diffusion, 55
 - sorbed_molefraction, 56
 - sorption_bc, 56
 - species, 56
- MONKFISH_TESTS
 - monkfish.cpp, 125
 - monkfish.h, 126
- mSPD_DATA, 56

- eMax, [56](#)
- eta, [56](#)
- gama, [56](#)
- s, [56](#)
- v, [56](#)
- macaw
 - ui.h, [150](#)
- macaw.cpp, [119](#)
 - MACAW_TESTS, [119](#)
- macaw.h, [119](#)
 - M_PI, [120](#)
 - MACAW_TESTS, [120](#)
- Magnetic_R
 - Trajectory.cpp, [144](#)
 - Trajectory.h, [146](#)
- Magnetic_T
 - Trajectory.cpp, [144](#)
 - Trajectory.h, [146](#)
- magpie
 - ui.h, [150](#)
- magpie.cpp, [120](#)
 - dq_dp, [121](#)
 - eMax, [121](#)
 - eval_GPAST, [121](#)
 - eval_eta, [121](#)
 - eval_po, [121](#)
 - eval_po_PI, [121](#)
 - eval_po_qo, [121](#)
 - grad_mSPD, [121](#)
 - initialGuess_mSPD, [121](#)
 - Inact_mSPD, [121](#)
 - MAGPIE, [121](#)
 - MAGPIE_SCENARIOS, [121](#)
 - PI, [121](#)
 - q_p, [121](#)
 - qT, [121](#)
 - qo, [121](#)
 - Qst, [121](#)
- magpie.h, [121](#)
 - A, [122](#)
 - DBL_EPSILON, [122](#)
 - dq_dp, [123](#)
 - eMax, [123](#)
 - eval_GPAST, [123](#)
 - eval_eta, [123](#)
 - eval_po, [123](#)
 - eval_po_PI, [123](#)
 - eval_po_qo, [123](#)
 - grad_mSPD, [123](#)
 - He, [122](#)
 - initialGuess_mSPD, [123](#)
 - kB, [122](#)
 - InKo, [122](#)
 - Inact_mSPD, [123](#)
 - MAGPIE, [123](#)
 - MAGPIE_SCENARIOS, [123](#)
 - Na, [122](#)
 - PI, [123](#)
 - Po, [123](#)
 - q_p, [123](#)
 - qT, [123](#)
 - qo, [123](#)
 - Qst, [123](#)
 - R, [123](#)
 - shapeFactor, [123](#)
 - V, [123](#)
 - Z, [123](#)
- magpie_reverse_error
 - error.h, [103](#)
- magpie_dat
 - SCOPSOWL_DATA, [68](#)
 - SKUA_DATA, [76](#)
- main
 - main.cpp, [124](#)
- main.cpp, [123](#)
 - main, [124](#)
- MassBalance, [41](#)
 - ~MassBalance, [42](#)
 - Delta, [42](#)
 - Display_Info, [42](#)
 - Eval_Residual, [42](#)
 - Get_Delta, [42](#)
 - Get_Name, [42](#)
 - Get_TotalConcentration, [42](#)
 - Initialize_List, [42](#)
 - List, [42](#)
 - MassBalance, [42](#)
 - MassBalance, [42](#)
 - Name, [42](#)
 - Set_Delta, [42](#)
 - Set_Name, [42](#)
 - Set_TotalConcentration, [42](#)
 - Sum_Delta, [42](#)
 - TotalConcentration, [42](#)
- MassBalanceList
 - SHARK_DATA, [74](#)
- MasterList
 - SHARK_DATA, [74](#)
- MasterSpeciesList, [42](#)
 - ~MasterSpeciesList, [43](#)
 - alkalinity, [43](#)
 - charge, [43](#)
 - DisplayAll, [43](#)
 - DisplayConcentrations, [43](#)
 - DisplayInfo, [44](#)
 - Eval_ChargeResidual, [44](#)
 - get_index, [44](#)
 - get_species, [44](#)
 - list_size, [44](#)
 - MasterSpeciesList, [43](#)
 - MasterSpeciesList, [43](#)
 - operator=, [44](#)
 - residual_alkalinity, [44](#)
 - set_alkalinity, [44](#)
 - set_list_size, [44](#)
 - set_species, [44](#)

- size, [44](#)
- species, [44](#)
- speciesName, [44](#)
- Matrix
 - ~Matrix, [46](#)
 - adjoint, [46](#)
 - cofactor, [46](#)
 - columnExtend, [46](#)
 - columnExtract, [46](#)
 - columnProjection, [46](#)
 - columnReplace, [46](#)
 - columnShrink, [46](#)
 - columnVectorFill, [46](#)
 - columns, [46](#)
 - ConstantICFill, [46](#)
 - Data, [48](#)
 - determinate, [46](#)
 - diagonalSolve, [46](#)
 - dirichletBCFill, [46](#)
 - Display, [46](#)
 - edit, [46](#)
 - inner_product, [46](#)
 - IntegralAvg, [46](#)
 - IntegralTotal, [46](#)
 - inverse, [46](#)
 - ladshawSolve, [46](#)
 - lowerHessenberg2Triangular, [46](#)
 - lowerHessenbergSolve, [46](#)
 - lowerTriangularSolve, [47](#)
 - Matrix, [46](#)
 - naturalLaplacian3D, [47](#)
 - norm, [47](#)
 - num_cols, [48](#)
 - num_rows, [48](#)
 - operator*, [47](#)
 - operator(), [47](#)
 - operator+, [47](#)
 - operator-, [47](#)
 - operator/, [47](#)
 - operator=, [47](#)
 - rowExtend, [47](#)
 - rowExtract, [47](#)
 - rowReplace, [47](#)
 - rowShrink, [47](#)
 - rows, [47](#)
 - set_size, [47](#)
 - SolnTransform, [47](#)
 - sphericalAvg, [47](#)
 - sphericalBCFill, [47](#)
 - sum, [47](#)
 - transpose, [47](#)
 - transpose_multiply, [47](#)
 - tridiagonalFill, [47](#)
 - tridiagonalSolve, [47](#)
 - tridiagonalVectorFill, [47](#)
 - upperHessenberg2Triangular, [48](#)
 - upperHessenbergSolve, [48](#)
 - upperTriangularSolve, [48](#)
 - zeros, [48](#)
- Matrix< T >, [44](#)
- matrix_too_small
 - error.h, [104](#)
- matvec
 - GMRESR_DATA, [31](#)
- matvec_mis_match
 - error.h, [104](#)
- matvec_data
 - GMRESR_DATA, [31](#)
- matvec_ex01
 - lark.cpp, [115](#)
 - lark.h, [118](#)
- matvec_ex02
 - lark.cpp, [115](#)
 - lark.h, [118](#)
- matvec_ex04
 - lark.cpp, [115](#)
 - lark.h, [118](#)
- matvec_ex15
 - lark.cpp, [115](#)
 - lark.h, [118](#)
- max
 - finch.cpp, [106](#)
 - finch.h, [109](#)
 - UI_DATA, [87](#)
- max_bias
 - SCOPSOWL_OPT_DATA, [70](#)
 - SKUA_OPT_DATA, [78](#)
- max_fiber_density
 - MONKFISH_DATA, [54](#)
- max_guess_iter
 - SCOPSOWL_OPT_DATA, [70](#)
 - SKUA_OPT_DATA, [78](#)
- max_iter
 - FINCH_DATA, [25](#)
- max_norm
 - SYSTEM_DATA, [83](#)
- max_porosity
 - MONKFISH_DATA, [54](#)
- max_value
 - UnsteadyReaction, [91](#)
- maxit
 - BiCGSTAB_DATA, [12](#)
 - CGS_DATA, [14](#)
 - GCR_DATA, [28](#)
 - GMRESLP_DATA, [29](#)
 - GMRESRP_DATA, [32](#)
 - PCG_DATA, [58](#)
 - PICARD_DATA, [60](#)
- Mechanism, [48](#)
 - List, [48](#)
 - reactions, [48](#)
 - species_index, [48](#)
 - weight, [48](#)
- min
 - finch.cpp, [106](#)
 - finch.h, [109](#)

- min_bias
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- min_fiber_density
 - MONKFISH_DATA, 54
- min_porosity
 - MONKFISH_DATA, 54
- minIndex
 - gsta_opt.cpp, 111
 - gsta_opt.h, 112
- minValue
 - gsta_opt.cpp, 111
 - gsta_opt.h, 113
- minmod
 - finch.cpp, 106
 - finch.h, 109
- minmod_discretization
 - finch.cpp, 106
 - finch.h, 109
- missing_information
 - error.h, 104
- MissingArg
 - UI_DATA, 87
- mola
 - ui.h, 150
- mola.cpp, 124
 - MOLA_TESTS, 124
- mola.h, 124
 - MOLA_TESTS, 124
- molar_weight
 - Molecule, 52
- MolarWeight
 - Molecule, 52
- molecular_diffusion
 - PURE_GAS, 63
- molecular_weight
 - PURE_GAS, 63
- MolecularFormula
 - Molecule, 52
- Molecule, 50
 - ~Molecule, 51
 - atoms, 52
 - calculateAvgOxiState, 51
 - Charge, 51
 - charge, 52
 - DisplayInfo, 51
 - editAllOxidationStates, 51
 - editCharge, 51
 - editEnergy, 51
 - editEnthalpy, 51
 - editEntropy, 51
 - editHS, 51
 - editOneOxidationState, 51
 - Energy, 51
 - Enthalpy, 51
 - Entropy, 51
 - formation_energy, 52
 - formation_enthalpy, 52
 - formation_entropy, 52
 - Formula, 52
 - HaveEnergy, 52
 - haveG, 52
 - HaveHS, 52
 - haveHS, 52
 - isRegistered, 52
 - molar_weight, 52
 - MolarWeight, 52
 - MolecularFormula, 52
 - Molecule, 51
 - MoleculeName, 52
 - MoleculePhase, 52
 - Name, 52
 - Phase, 52
 - recalculateMolarWeight, 52
 - Register, 52
 - registered, 52
 - removeAllAtoms, 52
 - removeOneAtom, 52
 - setFormula, 52
 - setMolarWeight, 52
- MoleculeName
 - Molecule, 52
- MoleculePhase
 - Molecule, 52
- molefraction
 - MIXED_GAS, 49
- molefractionCheck
 - skua.cpp, 139
 - skua.h, 141
- monkfish
 - ui.h, 150
- monkfish.cpp, 124
 - default_density, 125
 - default_exterior_concentration, 125
 - default_film_transfer, 125
 - default_interparticle_diffusion, 125
 - default_monk_adsorption, 125
 - default_monk_equilibrium, 125
 - default_monkfish_retardation, 125
 - default_porosity, 125
 - MONKFISH_TESTS, 125
- monkfish.h, 125
 - default_density, 126
 - default_exterior_concentration, 126
 - default_film_transfer, 126
 - default_interparticle_diffusion, 126
 - default_monk_adsorption, 126
 - default_monk_equilibrium, 126
 - default_monkfish_retardation, 126
 - default_porosity, 126
 - MONKFISH_TESTS, 126
 - setup_MONKFISH_DATA, 126
- mp
 - TRAJECTORY_DATA, 85
- Ms
 - TRAJECTORY_DATA, 85

mspd_dat
 MAGPIE_DATA, 41
 Mu
 egret.h, 102
 mu_0
 TRAJECTORY_DATA, 85
 MultiScale
 MONKFISH_DATA, 54
 N
 EX09_DATA, 21
 EX15_DATA, 21
 GMRESR_DATA, 31
 MIXED_GAS, 49
 Speciation_Test01_Data, 80
 SYSTEM_DATA, 83
 NONE
 yaml_wrapper.h, 153
 n_par
 GSTA_OPT_DATA, 35
 n_rand
 TRAJECTORY_DATA, 85
 NE
 FINCH_DATA, 25
 NI
 FINCH_DATA, 25
 NL_Output
 PJFNK_DATA, 62
 NUM_JAC_DATA, 56
 dxj, 57
 eps, 57
 Fx, 57
 Fxp, 57
 Na
 gsta_opt.h, 112
 magpie.h, 122
 NaT
 Speciation_Test01_Data, 80
 Name
 Atom, 10
 MassBalance, 42
 Molecule, 52
 name
 SubHeader, 82
 naturalLaplacian3D
 Matrix, 47
 NaturalState
 Atom, 10
 negative_mass
 error.h, 104
 negative_time
 error.h, 104
 Neutrons
 Atom, 10
 neutrons
 Atom, 10
 Newton_data
 SHARK_DATA, 74
 nl_bestres
 PJFNK_DATA, 62
 nl_iter
 PJFNK_DATA, 62
 nl_maxit
 PJFNK_DATA, 62
 nl_method
 FINCH_DATA, 25
 nl_picard
 finch.cpp, 106
 finch.h, 109
 nl_relres
 PJFNK_DATA, 62
 nl_res
 PJFNK_DATA, 62
 nl_res_base
 PJFNK_DATA, 62
 nl_tol_abs
 PJFNK_DATA, 62
 nl_tol_rel
 PJFNK_DATA, 62
 no_diffusion
 error.h, 104
 non_real_edge
 error.h, 104
 non_square_matrix
 error.h, 103
 NonLinear
 DOGFISH_DATA, 18
 MONKFISH_DATA, 54
 SCOPSOWL_DATA, 68
 SKUA_DATA, 76
 Norm
 SHARK_DATA, 74
 norm
 Matrix, 47
 normFkp1
 BACKTRACK_DATA, 11
 NormTrack
 FINCH_DATA, 25
 norms
 GSTA_OPT_DATA, 35
 not_a_token
 error.h, 104
 Nu
 egret.h, 102
 nullptr_error
 error.h, 104
 nullptr_func
 error.h, 104
 num_cols
 Matrix, 48
 num_curves
 SCOPSOWL_OPT_DATA, 70
 SKUA_OPT_DATA, 78
 num_mbe
 SHARK_DATA, 74
 num_other
 SHARK_DATA, 74

- num_params
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- num_rows
 - Matrix, 48
- num_ssr
 - SHARK_DATA, 74
- num_usr
 - SHARK_DATA, 74
- NumComp
 - DOGFISH_DATA, 18
 - MONKFISH_DATA, 54
- NumJac
 - Speciation_Test01_Data, 80
- Number_Generator
 - Trajectory.cpp, 144
 - Trajectory.h, 146
- number_elements
 - PeriodicTable, 59
- number_files
 - ui.cpp, 148
 - ui.h, 151
- NumericalJacobian
 - lark.cpp, 115
 - lark.h, 119
- numvar
 - SHARK_DATA, 74
- OE
 - FINCH_DATA, 25
- OI
 - FINCH_DATA, 26
- OPTRANS_DATA, 57
 - Ai, 57
 - li, 57
- omega
 - BiCGSTAB_DATA, 12
- omega_old
 - BiCGSTAB_DATA, 12
- operator*
 - Matrix, 47
- operator()
 - Document, 17
 - Header, 38
 - Matrix, 47
 - YamlWrapper, 96
- operator+
 - Matrix, 47
- operator-
 - Matrix, 47
- operator/
 - Matrix, 47
- operator=
 - Document, 17
 - Header, 38
 - KeyValueMap, 40
 - MasterSpeciesList, 44
 - Matrix, 47
 - SubHeader, 82
 - ValueTypePair, 93
 - YamlWrapper, 96
- operatorTranspose
 - lark.cpp, 115
 - lark.h, 119
- opt_no_support
 - error.h, 104
- opt_qmax
 - GSTA_OPT_DATA, 35
- Optimize
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- option
 - UI_DATA, 87
- orderMag
 - gsta_opt.cpp, 111
 - gsta_opt.h, 113
- ortho_check_fail
 - error.h, 104
- orthoLinReg
 - gsta_opt.cpp, 111
 - gsta_opt.h, 113
- ospre_discretization
 - finch.cpp, 106
 - finch.h, 109
- other_data
 - SHARK_DATA, 74
- OtherList
 - SHARK_DATA, 74
- out_of_bounds
 - error.h, 103
- Output
 - ARNOLDI_DATA, 7
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 14
 - GCR_DATA, 28
 - GMRESLP_DATA, 30
 - GMRESRP_DATA, 32
 - MONKFISH_DATA, 54
 - PCG_DATA, 58
 - PICARD_DATA, 60
 - SYSTEM_DATA, 83
- OutputFile
 - DOGFISH_DATA, 18
 - SCOPSOWL_DATA, 68
 - SHARK_DATA, 74
 - SKUA_DATA, 76
- owl_dat
 - SCOPSOWL_OPT_DATA, 70
- oxidation_state
 - Atom, 10
- OxidationState
 - Atom, 10
- P
 - GSTA_OPT_DATA, 35
- p
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 14

- EX09_DATA, 21
- PCG_DATA, 58
- PCG
 - lark.h, 117
- PITZER
 - shark.h, 137
- PCG_DATA, 57
 - alpha, 58
 - Ap, 58
 - bestres, 58
 - bestx, 58
 - beta, 58
 - iter, 58
 - maxit, 58
 - Output, 58
 - p, 58
 - r, 58
 - r_old, 58
 - relres, 58
 - relres_base, 58
 - res, 58
 - tol_abs, 58
 - tol_rel, 58
 - x, 58
 - z, 58
 - z_old, 58
- PE3
 - egret.h, 102
- pH
 - SHARK_DATA, 74
- pH_index
 - SHARK_DATA, 74
- PI
 - magpie.cpp, 121
 - magpie.h, 123
 - SYSTEM_DATA, 83
- PICARD_DATA, 60
 - bestres, 60
 - bestx, 60
 - iter, 60
 - maxit, 60
 - Output, 60
 - r, 60
 - relres, 60
 - relres_base, 60
 - res, 60
 - tol_abs, 60
 - tol_rel, 60
 - x0, 60
- Plo
 - GPAST_DATA, 33
- PJFNK_DATA, 60
 - backtrack_dat, 61
 - bestx, 61
 - bicgstab_dat, 61
 - Bounce, 61
 - cgs_dat, 61
 - eps, 61
- F, 61
- funeval, 61
- Fv, 61
- gcr_dat, 62
- gmreslp_dat, 62
- gmresr_dat, 62
- gmresrp_dat, 62
- L_Output, 62
- l_iter, 62
- lin_tol_abs, 62
- lin_tol_rel, 62
- LineSearch, 62
- linear_solver, 62
- NL_Output, 62
- nl_bestres, 62
- nl_iter, 62
- nl_maxit, 62
- nl_relres, 62
- nl_res, 62
- nl_res_base, 62
- nl_tol_abs, 62
- nl_tol_rel, 62
- pcg_dat, 62
- precon, 62
- precon_data, 62
- res_data, 62
- v, 62
- x, 62
- pOH_index
 - SHARK_DATA, 74
- POL
 - TRAJECTORY_DATA, 86
- POLAR
 - Trajectory.cpp, 145
 - Trajectory.h, 146
- PSI
 - egret.h, 102
- PT
 - SYSTEM_DATA, 83
- PURE_GAS, 63
 - density, 63
 - dynamic_viscosity, 63
 - molecular_diffusion, 63
 - molecular_weight, 63
 - Schmidt, 63
 - specific_heat, 63
 - Sutherland_Const, 63
 - Sutherland_Temp, 63
 - Sutherland_Viscosity, 64
- Par
 - SYSTEM_DATA, 83
- param_dat
 - DOGFISH_DATA, 18
 - MONKFISH_DATA, 54
 - SCOPSOWL_DATA, 68
 - SKUA_DATA, 76
- param_data
 - FINCH_DATA, 26

- param_guess
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- param_guess_old
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- ParamFile
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- Path
 - UI_DATA, 87
- path
 - ui.cpp, 148
 - ui.h, 151
 - UI_DATA, 87
- pcg
 - lark.cpp, 115
 - lark.h, 119
- pcg_dat
 - PJFNK_DATA, 62
- pellet_density
 - SCOPSOWL_DATA, 68
- pellet_radius
 - SCOPSOWL_DATA, 68
 - SKUA_DATA, 76
- PeriodicTable, 58
 - ~PeriodicTable, 59
 - DisplayTable, 59
 - number_elements, 59
 - PeriodicTable, 59
 - PeriodicTable, 59
 - Table, 59
- Phase
 - Molecule, 52
- pi
 - SYSTEM_DATA, 83
- picard
 - lark.cpp, 115
 - lark.h, 119
- picard_dat
 - FINCH_DATA, 26
- pjfnk
 - lark.cpp, 115
 - lark.h, 119
- pjfnk_dat
 - FINCH_DATA, 26
- Po
 - egret.h, 102
 - gsta_opt.h, 112
 - magpie.h, 123
- po
 - GPAST_DATA, 33
- poi
 - GPAST_DATA, 33
- pore_diffusion
 - SCOPSOWL_PARAM_DATA, 71
- porosity
 - TRAJECTORY_DATA, 86
- Precipitation, 63
- precon
 - PJFNK_DATA, 62
- precon_data
 - PJFNK_DATA, 62
 - SHARK_DATA, 74
- precon_ex01
 - lark.cpp, 115
 - lark.h, 119
- precon_ex04
 - lark.cpp, 115
 - lark.h, 119
- precon_ex10
 - lark.cpp, 115
 - lark.h, 119
- precon_ex15
 - lark.cpp, 115
 - lark.h, 119
- pres
 - FINCH_DATA, 26
- present
 - GPAST_DATA, 33
- previous_token
 - yaml_cpp_class, 94
- Print2Console
 - DOGFISH_DATA, 18
 - MONKFISH_DATA, 54
 - SCOPSOWL_DATA, 68
 - SKUA_DATA, 76
- Print2File
 - DOGFISH_DATA, 18
 - MONKFISH_DATA, 54
 - SCOPSOWL_DATA, 68
 - SKUA_DATA, 76
- print2file_DOGFISH_header
 - dogfish.cpp, 98
 - dogfish.h, 99
- print2file_DOGFISH_result_new
 - dogfish.cpp, 98
 - dogfish.h, 99
- print2file_DOGFISH_result_old
 - dogfish.cpp, 98
 - dogfish.h, 99
- print2file_SCOPSOWL_header
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- print2file_SCOPSOWL_result_new
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- print2file_SCOPSOWL_result_old
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- print2file_SCOPSOWL_time_header
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- print2file_SKUA_header
 - skua.cpp, 139
 - skua.h, 142

print2file_SKUA_results_new
 skua.cpp, 139
 skua.h, 142
 print2file_SKUA_results_old
 skua.cpp, 139
 skua.h, 142
 print2file_SKUA_time_header
 skua.cpp, 139
 skua.h, 142
 print2file_dim_header
 finch.cpp, 106
 finch.h, 109
 print2file_newline
 finch.cpp, 106
 finch.h, 109
 print2file_result_new
 finch.cpp, 106
 finch.h, 109
 print2file_result_old
 finch.cpp, 106
 finch.h, 109
 print2file_shark_header
 shark.cpp, 134
 shark.h, 137
 print2file_shark_info
 shark.cpp, 134
 shark.h, 137
 print2file_shark_results_new
 shark.cpp, 134
 shark.h, 137
 print2file_shark_results_old
 shark.cpp, 134
 shark.h, 137
 print2file_species_header
 dogfish.cpp, 98
 dogfish.h, 99
 scopsowl.cpp, 129
 scopsowl.h, 131
 skua.cpp, 139
 skua.h, 142
 print2file_tab
 finch.cpp, 106
 finch.h, 109
 print2file_time_header
 finch.cpp, 106
 finch.h, 109
 Protons
 Atom, 10
 protons
 Atom, 10
 Pstd
 egret.h, 102
 q
 GPAST_DATA, 34
 GSTA_OPT_DATA, 35
 q_bar
 TRAJECTORY_DATA, 86
 q_data
 SCOPSOWL_OPT_DATA, 70
 SKUA_OPT_DATA, 78
 Q_in
 TRAJECTORY_DATA, 86
 q_p
 magpie.cpp, 121
 magpie.h, 123
 q_sim
 SCOPSOWL_OPT_DATA, 70
 SKUA_OPT_DATA, 78
 qAvg
 SCOPSOWL_PARAM_DATA, 71
 qAvg_old
 SCOPSOWL_PARAM_DATA, 71
 qIntegralAvg
 SCOPSOWL_PARAM_DATA, 71
 qIntegralAvg_old
 SCOPSOWL_PARAM_DATA, 71
 qT
 magpie.cpp, 121
 magpie.h, 123
 SYSTEM_DATA, 83
 qTn
 SKUA_DATA, 76
 qTnp1
 SKUA_DATA, 76
 qmax
 GSTA_DATA, 34
 GSTA_OPT_DATA, 35
 qo
 GPAST_DATA, 34
 magpie.cpp, 121
 magpie.h, 123
 SCOPSOWL_PARAM_DATA, 71
 Qst
 magpie.cpp, 121
 magpie.h, 123
 SCOPSOWL_PARAM_DATA, 71
 Qst_old
 SCOPSOWL_PARAM_DATA, 71
 QstAvg
 SCOPSOWL_PARAM_DATA, 71
 QstAvg_old
 SCOPSOWL_PARAM_DATA, 71
 Qstn
 SKUA_PARAM, 79
 Qstnp1
 SKUA_PARAM, 79
 Qsto
 SCOPSOWL_PARAM_DATA, 72
 R
 gsta_opt.h, 112
 magpie.h, 123
 r
 BiCGSTAB_DATA, 12
 CGS_DATA, 14
 GCR_DATA, 28
 GMRESLP_DATA, 30

- GMRESRP_DATA, [32](#)
- PCG_DATA, [58](#)
- PICARD_DATA, [60](#)
- r0
 - BiCGSTAB_DATA, [12](#)
 - CGS_DATA, [14](#)
- r_old
 - PCG_DATA, [58](#)
- RADIAL_FORCE
 - Trajectory.cpp, [145](#)
 - Trajectory.h, [146](#)
- RE3
 - egret.h, [102](#)
- RIC
 - FINCH_DATA, [26](#)
- rSq
 - gsta_opt.cpp, [111](#)
 - gsta_opt.h, [113](#)
- RUN_SANDBOX
 - sandbox.cpp, [126](#)
 - sandbox.h, [127](#)
- ReNum
 - egret.h, [102](#)
- Reaction, [64](#)
 - ~Reaction, [65](#)
 - calculateEnergies, [65](#)
 - calculateEquilibrium, [65](#)
 - CanCalcG, [65](#)
 - CanCalcHS, [65](#)
 - checkSpeciesEnergies, [65](#)
 - Display_Info, [65](#)
 - energy, [66](#)
 - enthalpy, [66](#)
 - entropy, [66](#)
 - Equilibrium, [66](#)
 - Eval_Residual, [65](#)
 - Get_Energy, [65](#)
 - Get_Enthalpy, [65](#)
 - Get_Entropy, [65](#)
 - Get_Equilibrium, [65](#)
 - Get_Stoichiometric, [65](#)
 - HaveEquil, [66](#)
 - haveEquilibrium, [65](#)
 - HaveG, [66](#)
 - HaveHS, [66](#)
 - Initialize_List, [65](#)
 - List, [66](#)
 - Reaction, [65](#)
 - Set_Energy, [65](#)
 - Set_Enthalpy, [65](#)
 - Set_EnthalpyANDEntropy, [65](#)
 - Set_Entropy, [65](#)
 - Set_Equilibrium, [65](#)
 - Set_Stoichiometric, [65](#)
 - Stoichiometric, [66](#)
- ReactionList
 - SHARK_DATA, [74](#)
- reactions
 - Mechanism, [48](#)
- read_error
 - error.h, [104](#)
- read_equilrxn
 - shark.cpp, [134](#)
 - shark.h, [137](#)
- read_massbalance
 - shark.cpp, [134](#)
 - shark.h, [137](#)
- read_options
 - shark.cpp, [134](#)
 - shark.h, [137](#)
- read_scenario
 - shark.cpp, [134](#)
 - shark.h, [137](#)
- read_species
 - shark.cpp, [134](#)
 - shark.h, [138](#)
- read_unsteadyrxn
 - shark.cpp, [134](#)
 - shark.h, [138](#)
- readInputFile
 - yaml_cpp_class, [94](#)
- recalculateMolarWeight
 - Molecule, [52](#)
- Recover
 - SYSTEM_DATA, [83](#)
- ref_diffusion
 - SCOPSOWL_PARAM_DATA, [72](#)
 - SKUA_PARAM, [79](#)
- ref_pressure
 - SCOPSOWL_PARAM_DATA, [72](#)
 - SKUA_PARAM, [79](#)
- ref_temperature
 - SCOPSOWL_PARAM_DATA, [72](#)
 - SKUA_PARAM, [79](#)
- Register
 - Atom, [10](#)
 - Molecule, [52](#)
- registered
 - Molecule, [52](#)
- rel_tol_norm
 - SCOPSOWL_OPT_DATA, [70](#)
 - SKUA_OPT_DATA, [78](#)
- relres
 - BiCGSTAB_DATA, [12](#)
 - CGS_DATA, [14](#)
 - GCR_DATA, [28](#)
 - GMRESLP_DATA, [30](#)
 - GMRESRP_DATA, [32](#)
 - PCG_DATA, [58](#)
 - PICARD_DATA, [60](#)
- relres_base
 - BiCGSTAB_DATA, [12](#)
 - CGS_DATA, [14](#)
 - GCR_DATA, [28](#)
 - GMRESLP_DATA, [30](#)
 - GMRESRP_DATA, [32](#)

- PCG_DATA, 58
- PICARD_DATA, 60
- Removal_Efficiency
 - Trajectory.cpp, 145
 - Trajectory.h, 146
- removeAllAtoms
 - Molecule, 52
- removeElectron
 - Atom, 10
- removeNeutron
 - Atom, 10
- removeOneAtom
 - Molecule, 52
- removeProton
 - Atom, 10
- res
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 14
 - FINCH_DATA, 26
 - GCR_DATA, 28
 - GMRESLP_DATA, 30
 - GMRESRP_DATA, 32
 - PCG_DATA, 58
 - PICARD_DATA, 60
- res_data
 - PJFNK_DATA, 62
- resetKeys
 - Document, 17
 - Header, 38
 - YamlWrapper, 96
- resetime
 - FINCH_DATA, 26
- Residual
 - SHARK_DATA, 74
- residual_alkalinity
 - MasterSpeciesList, 44
- residual_data
 - SHARK_DATA, 74
- restart
 - GCR_DATA, 29
 - GMRESLP_DATA, 30
 - GMRESRP_DATA, 33
- revalidateAllKeys
 - Document, 17
 - YamlWrapper, 96
- reverse_rate
 - UnsteadyReaction, 91
- reverse_ref_rate
 - UnsteadyReaction, 91
- Reynolds
 - MIXED_GAS, 49
- rho
 - BACKTRACK_DATA, 11
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 14
- rho_f
 - TRAJECTORY_DATA, 86
- rho_old
 - BiCGSTAB_DATA, 12
- rho_p
 - TRAJECTORY_DATA, 86
- Rn
 - FINCH_DATA, 26
- Rnp1
 - FINCH_DATA, 26
- Ro
 - FINCH_DATA, 26
- Rough
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- roundIt
 - gsta_opt.cpp, 111
 - gsta_opt.h, 113
- rowExtend
 - Matrix, 47
- rowExtract
 - Matrix, 47
- rowReplace
 - Matrix, 47
- rowShrink
 - Matrix, 47
- rows
 - Matrix, 47
- Rs
 - TRAJECTORY_DATA, 86
- Rstd
 - egret.h, 102
 - shark.h, 137
- Run_Trajectory
 - Trajectory.cpp, 145
 - Trajectory.h, 146
- run_exec
 - ui.cpp, 148
 - ui.h, 151
- run_executable
 - ui.cpp, 148
 - ui.h, 151
- run_test
 - ui.cpp, 148
 - ui.h, 151
- rxn_rate_error
 - error.h, 104
- s
 - BiCGSTAB_DATA, 12
 - EX09_DATA, 21
 - FINCH_DATA, 26
 - mSPD_DATA, 56
- SIT
 - shark.h, 137
- STRING
 - yaml_wrapper.h, 152
- s_rand
 - TRAJECTORY_DATA, 86
- SCOPSOWL
 - scopsowl.cpp, 129
 - scopsowl.h, 131

- SCOPSOWL_DATA, 66
 - binder_fraction, 67
 - binder_poresize, 67
 - binder_porosity, 67
 - char_macro, 67
 - char_micro, 67
 - coord_macro, 67
 - coord_micro, 67
 - crystal_radius, 67
 - DirichletBC, 67
 - eval_ads, 67
 - eval_diff, 67
 - eval_kf, 67
 - eval_retard, 67
 - eval_surfDiff, 67
 - finch_dat, 67
 - gas_dat, 67
 - gas_temperature, 67
 - gas_velocity, 67
 - Heterogeneous, 68
 - level, 68
 - magpie_dat, 68
 - NonLinear, 68
 - OutputFile, 68
 - param_dat, 68
 - pellet_density, 68
 - pellet_radius, 68
 - Print2Console, 68
 - Print2File, 68
 - sim_time, 68
 - skua_dat, 68
 - SurfDiff, 68
 - t, 68
 - t_counter, 68
 - t_old, 68
 - t_print, 68
 - tempy, 68
 - total_pressure, 68
 - total_steps, 68
 - user_data, 68
 - y, 68
- SCOPSOWL_Executioner
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- SCOPSOWL_HPP_
 - scopsowl.h, 130
- SCOPSOWL_OPT_DATA, 68
 - adsorb_index, 69
 - CompareFile, 69
 - current_equil, 69
 - current_points, 69
 - current_press, 69
 - current_temp, 69
 - diffusion_type, 69
 - e_norm, 69
 - evaluation, 70
 - f_bias, 70
 - max_bias, 70
 - min_bias, 70
 - num_curves, 70
 - num_params, 70
 - Optimize, 70
 - owl_dat, 70
 - param_guess, 70
 - ParamFile, 70
 - q_data, 70
 - q_sim, 70
 - Rough, 70
 - simulation_equil, 70
 - t, 70
 - total_eval, 70
 - y_base, 70
- SCOPSOWL_OPT_set_y
 - scopsowl_opt.cpp, 132
 - scopsowl_opt.h, 132
- SCOPSOWL_OPTIMIZE
 - scopsowl_opt.cpp, 132
 - scopsowl_opt.h, 132
- SCOPSOWL_PARAM_DATA, 70
 - Adsorbable, 71
 - affinity, 71
 - qAvg, 71
 - qo, 71
 - Qst, 71
 - QstAvg, 71
 - Qsto, 72
 - speciesName, 72
- SCOPSOWL_SCENARIOS
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- SCOPSOWL_TESTS
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- SCOPSOWL_postprocesses
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- SCOPSOWL_preprocesses
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- SCOPSOWL_reset
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- SHARK
 - shark.cpp, 134
 - shark.h, 138
- SHARK_DATA, 72
 - act_fun, 73
 - activity_data, 73
 - activity_new, 73
 - activity_old, 73
 - Conc_new, 73
 - Conc_old, 73
 - Console_Output, 73
 - const_pH, 73
 - Contains_pH, 73
 - Contains_pOH, 73

- Converged, [73](#)
- dielectric_const, [73](#)
- dt, [73](#)
- dt_min, [73](#)
- EvalActivity, [74](#)
- File_Output, [74](#)
- lin_precon, [74](#)
- MassBalanceList, [74](#)
- MasterList, [74](#)
- Newton_data, [74](#)
- Norm, [74](#)
- num_mbe, [74](#)
- num_other, [74](#)
- num_ssr, [74](#)
- num_usr, [74](#)
- numvar, [74](#)
- other_data, [74](#)
- OtherList, [74](#)
- OutputFile, [74](#)
- pH, [74](#)
- pH_index, [74](#)
- pOH_index, [74](#)
- precon_data, [74](#)
- ReactionList, [74](#)
- Residual, [74](#)
- residual_data, [74](#)
- shark.h, [137](#)
- simulationtime, [74](#)
- SpeciationCurve, [74](#)
- steadystate, [74](#)
- t_count, [74](#)
- t_out, [74](#)
- temperature, [74](#)
- time, [75](#)
- time_old, [75](#)
- TimeAdaptivity, [75](#)
- timesteps, [75](#)
- totalsteps, [75](#)
- UnsteadyList, [75](#)
- X_new, [75](#)
- X_old, [75](#)
- yaml_object, [75](#)
- SHARK_SCENARIO
 - shark.cpp, [135](#)
 - shark.h, [138](#)
- SHARK_TESTS
 - shark.cpp, [135](#)
 - shark.h, [138](#)
- SKUA
 - skua.cpp, [140](#)
 - skua.h, [142](#)
- SKUA_CYCLE_TEST01
 - skua.cpp, [140](#)
 - skua.h, [142](#)
- SKUA_CYCLE_TEST02
 - skua.cpp, [140](#)
 - skua.h, [142](#)
- SKUA_DATA, [75](#)
- char_measure, [76](#)
- coord, [76](#)
- DirichletBC, [76](#)
- eval_diff, [76](#)
- eval_kf, [76](#)
- finch_dat, [76](#)
- gas_dat, [76](#)
- gas_velocity, [76](#)
- maggie_dat, [76](#)
- NonLinear, [76](#)
- OutputFile, [76](#)
- param_dat, [76](#)
- pellet_radius, [76](#)
- Print2Console, [76](#)
- Print2File, [76](#)
- qTn, [76](#)
- qTnp1, [76](#)
- sim_time, [76](#)
- t, [76](#)
- t_counter, [76](#)
- t_old, [76](#)
- t_print, [76](#)
- total_steps, [76](#)
- user_data, [76](#)
- y, [76](#)
- SKUA_Executioner
 - skua.cpp, [140](#)
 - skua.h, [142](#)
- SKUA_HPP_
 - skua.h, [141](#)
- SKUA_LOW_TEST03
 - skua.cpp, [140](#)
 - skua.h, [142](#)
- SKUA_MID_TEST04
 - skua.cpp, [140](#)
 - skua.h, [142](#)
- SKUA_OPT_DATA, [77](#)
 - abs_tol_bias, [77](#)
 - adsorb_index, [77](#)
 - CompareFile, [77](#)
 - current_equil, [77](#)
 - current_points, [77](#)
 - current_press, [77](#)
 - current_temp, [77](#)
 - diffusion_type, [77](#)
 - e_norm, [78](#)
 - e_norm_old, [78](#)
 - evaluation, [78](#)
 - f_bias, [78](#)
 - f_bias_old, [78](#)
 - max_bias, [78](#)
 - max_guess_iter, [78](#)
 - min_bias, [78](#)
 - num_curves, [78](#)
 - num_params, [78](#)
 - Optimize, [78](#)
 - param_guess, [78](#)
 - param_guess_old, [78](#)

- ParamFile, 78
- q_data, 78
- q_sim, 78
- rel_tol_norm, 78
- Rough, 78
- simulation_equil, 78
- skua_dat, 78
- t, 78
- total_eval, 78
- y_base, 78
- SKUA_OPT_set_y
 - skua_opt.cpp, 143
 - skua_opt.h, 143
- SKUA_OPTIMIZE
 - skua_opt.cpp, 143
 - skua_opt.h, 143
- SKUA_PARAM, 78
 - activation_energy, 79
 - Adsorbable, 79
 - affinity, 79
 - film_transfer, 79
 - Qstn, 79
 - Qstnp1, 79
 - ref_diffusion, 79
 - ref_pressure, 79
 - ref_temperature, 79
 - speciesName, 79
 - xiC, 79
 - xn, 79
 - xnp1, 79
 - y_eff, 79
- SKUA_SCENARIOS
 - skua.cpp, 140
 - skua.h, 142
- SKUA_TESTS
 - skua.cpp, 140
 - skua.h, 142
- SKUA_postprocesses
 - skua.cpp, 140
 - skua.h, 142
- SKUA_preprocesses
 - skua.cpp, 140
 - skua.h, 142
- SKUA_reset
 - skua.cpp, 140
 - skua.h, 142
- SMALL_CYCLE_TEST02
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- SYSTEM_DATA, 82
 - As, 83
 - avg_norm, 83
 - Carrier, 83
 - I, 83
 - Ideal, 83
 - J, 83
 - K, 83
 - max_norm, 83
 - N, 83
 - Output, 83
 - PI, 83
 - PT, 83
 - Par, 83
 - pi, 83
 - qT, 83
 - Recover, 83
 - Sys, 83
 - T, 83
 - total_eval, 84
- sandbox
 - ui.h, 150
- sandbox.cpp, 126
 - RUN_SANDBOX, 126
 - Speciation_Test01_Function, 126
 - Speciation_Test01_Guess, 126
 - Speciation_Test01_Jacobian, 126
 - Speciation_Test01_MatVec, 126
- sandbox.h, 127
 - RUN_SANDBOX, 127
 - Speciation_Test01_Function, 127
 - Speciation_Test01_Guess, 127
 - Speciation_Test01_Jacobian, 127
 - Speciation_Test01_MatVec, 127
- ScNum
 - egret.h, 102
- scenario_fail
 - error.h, 103
- Schmidt
 - PURE_GAS, 63
- school.h, 127
- scops_opt
 - ui.h, 150
- scopsowl
 - ui.h, 150
- scopsowl.cpp, 127
 - CURVE_TEST03, 128
 - CURVE_TEST04, 128
 - CURVE_TEST05, 128
 - const_filmMassTransfer, 128
 - const_pore_diffusion, 128
 - default_adsorption, 128
 - default_effective_diffusion, 128
 - default_filmMassTransfer, 128
 - default_pore_diffusion, 128
 - default_retardation, 128
 - default_surf_diffusion, 128
 - LARGE_CYCLE_TEST01, 128
 - print2file_SCOPSOWL_header, 129
 - print2file_SCOPSOWL_result_new, 129
 - print2file_SCOPSOWL_result_old, 129
 - print2file_SCOPSOWL_time_header, 129
 - print2file_species_header, 129
 - SCOPSOWL, 129
 - SCOPSOWL_Executioner, 129
 - SCOPSOWL_SCENARIOS, 129
 - SCOPSOWL_TESTS, 129

- SCOPSOWL_postprocesses, 129
- SCOPSOWL_preprocesses, 129
- SCOPSOWL_reset, 129
- SMALL_CYCLE_TEST02, 129
- set_SCOPSOWL_ICs, 129
- set_SCOPSOWL_params, 129
- set_SCOPSOWL_timestep, 129
- setup_SCOPSOWL_DATA, 129
- scopsowl.h, 129
 - avgDp, 130
 - CURVE_TEST03, 131
 - CURVE_TEST04, 131
 - CURVE_TEST05, 131
 - const_filmMassTransfer, 130
 - const_pore_diffusion, 130
 - default_adsorption, 131
 - default_effective_diffusion, 131
 - default_filmMassTransfer, 131
 - default_pore_diffusion, 131
 - default_retardation, 131
 - default_surf_diffusion, 131
 - Dk, 130
 - Dp, 130
 - LARGE_CYCLE_TEST01, 131
 - print2file_SCOPSOWL_header, 131
 - print2file_SCOPSOWL_result_new, 131
 - print2file_SCOPSOWL_result_old, 131
 - print2file_SCOPSOWL_time_header, 131
 - print2file_species_header, 131
 - SCOPSOWL, 131
 - SCOPSOWL_Executioner, 131
 - SCOPSOWL_HPP_, 130
 - SCOPSOWL_SCENARIOS, 131
 - SCOPSOWL_TESTS, 131
 - SCOPSOWL_postprocesses, 131
 - SCOPSOWL_preprocesses, 131
 - SCOPSOWL_reset, 131
 - SMALL_CYCLE_TEST02, 131
 - set_SCOPSOWL_ICs, 131
 - set_SCOPSOWL_params, 131
 - set_SCOPSOWL_timestep, 131
 - setup_SCOPSOWL_DATA, 131
- scopsowl_opt.cpp, 132
 - eval_SCOPSOWL_Uptake, 132
 - initial_guess_SCOPSOWL, 132
 - SCOPSOWL_OPT_set_y, 132
 - SCOPSOWL_OPTIMIZE, 132
- scopsowl_opt.h, 132
 - eval_SCOPSOWL_Uptake, 132
 - initial_guess_SCOPSOWL, 132
 - SCOPSOWL_OPT_set_y, 132
 - SCOPSOWL_OPTIMIZE, 132
- Set_ActivationEnergy
 - UnsteadyReaction, 90
- Set_Affinity
 - UnsteadyReaction, 90
- set_DOGFISH_ICs
 - dogfish.cpp, 98
- dogfish.h, 99
- set_DOGFISH_params
 - dogfish.cpp, 98
 - dogfish.h, 99
- set_DOGFISH_timestep
 - dogfish.cpp, 98
 - dogfish.h, 99
- Set_Delta
 - MassBalance, 42
- Set_Energy
 - Reaction, 65
 - UnsteadyReaction, 90
- Set_Enthalpy
 - Reaction, 65
 - UnsteadyReaction, 90
- Set_EnthalpyANDEntropy
 - Reaction, 65
 - UnsteadyReaction, 91
- Set_Entropy
 - Reaction, 65
 - UnsteadyReaction, 91
- Set_Equilibrium
 - Reaction, 65
 - UnsteadyReaction, 91
- Set_Forward
 - UnsteadyReaction, 91
- Set_ForwardRef
 - UnsteadyReaction, 91
- Set_InitialValue
 - UnsteadyReaction, 91
- Set_MaximumValue
 - UnsteadyReaction, 91
- Set_Name
 - MassBalance, 42
- Set_Reverse
 - UnsteadyReaction, 91
- Set_ReverseRef
 - UnsteadyReaction, 91
- set_SCOPSOWL_ICs
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- set_SCOPSOWL_params
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- set_SCOPSOWL_timestep
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- set_SKUA_ICs
 - skua.cpp, 140
 - skua.h, 142
- set_SKUA_params
 - skua.cpp, 140
 - skua.h, 142
- set_SKUA_timestep
 - skua.cpp, 140
 - skua.h, 142
- Set_Species_Index
 - UnsteadyReaction, 91

- Set_Stoichiometric
 - Reaction, 65
 - UnsteadyReaction, 91
- Set_TimeStep
 - UnsteadyReaction, 91
- Set_TotalConcentration
 - MassBalance, 42
- set_alkalinity
 - MasterSpeciesList, 44
- set_list_size
 - MasterSpeciesList, 44
- set_size
 - Matrix, 47
- set_species
 - MasterSpeciesList, 44
- set_variables
 - egret.cpp, 101
 - egret.h, 102
- setAlias
 - Document, 17
 - Header, 38
 - SubHeader, 82
- setFormula
 - Molecule, 52
- setInputFile
 - yaml_cpp_class, 94
- setMolarWeigth
 - Molecule, 52
- setName
 - Document, 17
 - Header, 38
 - SubHeader, 82
- setNameAliasPair
 - Document, 17
 - Header, 38
 - SubHeader, 82
- setState
 - Document, 17
 - Header, 38
 - SubHeader, 82
- setbcs
 - FINCH_DATA, 26
- setic
 - FINCH_DATA, 26
- setparams
 - FINCH_DATA, 26
- setpostprocess
 - FINCH_DATA, 26
- setpreprocess
 - FINCH_DATA, 26
- settime
 - FINCH_DATA, 26
- setup_DOGFISH_DATA
 - dogfish.cpp, 98
 - dogfish.h, 99
- setup_FINCH_DATA
 - finch.cpp, 106
 - finch.h, 109
- setup_MONKFISH_DATA
 - monkfish.h, 126
- setup_SCOPSOWL_DATA
 - scopsowl.cpp, 129
 - scopsowl.h, 131
- setup_SHARK_DATA
 - shark.cpp, 134
 - shark.h, 138
- setup_SKUA_DATA
 - skua.cpp, 140
 - skua.h, 142
- shapeFactor
 - magpie.h, 123
- shark
 - ui.h, 150
- shark.h
 - DAVIES, 137
 - DAVIES_LADSHAW, 137
 - DEBYE_HUCKEL, 137
 - IDEAL, 137
 - PITZER, 137
 - SIT, 137
- shark.cpp, 133
 - act_choice, 134
 - Convert2Concentration, 134
 - Convert2LogConcentration, 134
 - Davies_equation, 134
 - DaviesLadshaw_equation, 134
 - DebyeHuckel_equation, 134
 - ideal_solution, 134
 - linearsolve_choice, 134
 - linesearch_choice, 134
 - print2file_shark_header, 134
 - print2file_shark_info, 134
 - print2file_shark_results_new, 134
 - print2file_shark_results_old, 134
 - read_equilrxn, 134
 - read_massbalance, 134
 - read_options, 134
 - read_scenario, 134
 - read_species, 134
 - read_unsteadyrxn, 134
 - SHARK, 134
 - SHARK_SCENARIO, 135
 - SHARK_TESTS, 135
 - setup_SHARK_DATA, 134
 - shark_add_customResidual, 134
 - shark_energy_calculations, 134
 - shark_executioner, 134
 - shark_guess, 134
 - shark_initial_conditions, 134
 - shark_pH_finder, 135
 - shark_parameter_check, 135
 - shark_postprocesses, 135
 - shark_preprocesses, 135
 - shark_reset, 135
 - shark_residual, 135
 - shark_solver, 135

- shark_temperature_calculations, 135
- shark_timestep_adapt, 135
- shark_timestep_const, 135
- shark.h, 135
 - act_choice, 137
 - Convert2Concentration, 137
 - Convert2LogConcentration, 137
 - Davies_equation, 137
 - DaviesLadshaw_equation, 137
 - DebyeHuckel_equation, 137
 - ideal_solution, 137
 - linearsolve_choice, 137
 - linsearch_choice, 137
 - print2file_shark_header, 137
 - print2file_shark_info, 137
 - print2file_shark_results_new, 137
 - print2file_shark_results_old, 137
 - read_equilrxn, 137
 - read_massbalance, 137
 - read_options, 137
 - read_scenario, 137
 - read_species, 138
 - read_unsteadyrxn, 138
 - Rstd, 137
 - SHARK, 138
 - SHARK_DATA, 137
 - SHARK_SCENARIO, 138
 - SHARK_TESTS, 138
 - setup_SHARK_DATA, 138
 - shark_add_customResidual, 138
 - shark_energy_calculations, 138
 - shark_executioner, 138
 - shark_guess, 138
 - shark_initial_conditions, 138
 - shark_pH_finder, 138
 - shark_parameter_check, 138
 - shark_postprocesses, 138
 - shark_preprocesses, 138
 - shark_reset, 138
 - shark_residual, 138
 - shark_solver, 138
 - shark_temperature_calculations, 138
 - shark_timestep_adapt, 138
 - shark_timestep_const, 138
 - valid_act, 137
- shark_add_customResidual
 - shark.cpp, 134
 - shark.h, 138
- shark_energy_calculations
 - shark.cpp, 134
 - shark.h, 138
- shark_executioner
 - shark.cpp, 134
 - shark.h, 138
- shark_guess
 - shark.cpp, 134
 - shark.h, 138
- shark_initial_conditions
 - shark.cpp, 134
 - shark.h, 138
- shark.cpp, 134
- shark.h, 138
- shark_pH_finder
 - shark.cpp, 135
 - shark.h, 138
- shark_parameter_check
 - shark.cpp, 135
 - shark.h, 138
- shark_postprocesses
 - shark.cpp, 135
 - shark.h, 138
- shark_preprocesses
 - shark.cpp, 135
 - shark.h, 138
- shark_reset
 - shark.cpp, 135
 - shark.h, 138
- shark_residual
 - shark.cpp, 135
 - shark.h, 138
- shark_solver
 - shark.cpp, 135
 - shark.h, 138
- shark_temperature_calculations
 - shark.cpp, 135
 - shark.h, 138
- shark_timestep_adapt
 - shark.cpp, 135
 - shark.h, 138
- shark_timestep_const
 - shark.cpp, 135
 - shark.h, 138
- sigma
 - CGS_DATA, 14
- sigma_m
 - TRAJECTORY_DATA, 86
- sigma_n
 - TRAJECTORY_DATA, 86
- sigma_v
 - TRAJECTORY_DATA, 86
- sigma_vz
 - TRAJECTORY_DATA, 86
- sigma_z
 - TRAJECTORY_DATA, 86
- sim_time
 - SCOPSOWL_DATA, 68
 - SKUA_DATA, 76
- simple_darken_Dc
 - skua.cpp, 140
 - skua.h, 142
- simulation_fail
 - error.h, 103
- simulation_equil
 - SCOPSOWL_OPT_DATA, 70
 - SKUA_OPT_DATA, 78
- simulationtime
 - SHARK_DATA, 74
- single_fiber_density

- MONKFISH_DATA, 54
- singular_matrix
 - error.h, 104
- size
 - Document, 17
 - Header, 38
 - KeyValueMap, 40
 - MasterSpeciesList, 44
 - YamlWrapper, 96
- skua
 - ui.h, 150
- skua.cpp, 138
 - const_Dc, 139
 - const_kf, 139
 - default_Dc, 139
 - default_kf, 139
 - empirical_kf, 139
 - molefractionCheck, 139
 - print2file_SKUA_header, 139
 - print2file_SKUA_results_new, 139
 - print2file_SKUA_results_old, 139
 - print2file_SKUA_time_header, 139
 - print2file_species_header, 139
 - SKUA, 140
 - SKUA_CYCLE_TEST01, 140
 - SKUA_CYCLE_TEST02, 140
 - SKUA_Executioner, 140
 - SKUA_LOW_TEST03, 140
 - SKUA_MID_TEST04, 140
 - SKUA_SCENARIOS, 140
 - SKUA_TESTS, 140
 - SKUA_postprocesses, 140
 - SKUA_preprocesses, 140
 - SKUA_reset, 140
 - set_SKUA_ICs, 140
 - set_SKUA_params, 140
 - set_SKUA_timestep, 140
 - setup_SKUA_DATA, 140
 - simple_darken_Dc, 140
 - theoretical_darken_Dc, 140
- skua.h, 140
 - const_Dc, 141
 - const_kf, 141
 - D_c, 141
 - D_inf, 141
 - D_o, 141
 - default_Dc, 141
 - default_kf, 141
 - empirical_kf, 141
 - molefractionCheck, 141
 - print2file_SKUA_header, 142
 - print2file_SKUA_results_new, 142
 - print2file_SKUA_results_old, 142
 - print2file_SKUA_time_header, 142
 - print2file_species_header, 142
 - SKUA, 142
 - SKUA_CYCLE_TEST01, 142
 - SKUA_CYCLE_TEST02, 142
 - SKUA_Executioner, 142
 - SKUA_HPP_, 141
 - SKUA_LOW_TEST03, 142
 - SKUA_MID_TEST04, 142
 - SKUA_SCENARIOS, 142
 - SKUA_TESTS, 142
 - SKUA_postprocesses, 142
 - SKUA_preprocesses, 142
 - SKUA_reset, 142
 - set_SKUA_ICs, 142
 - set_SKUA_params, 142
 - set_SKUA_timestep, 142
 - setup_SKUA_DATA, 142
 - simple_darken_Dc, 142
 - theoretical_darken_Dc, 142
- skua_opt
 - ui.h, 150
- skua_dat
 - SCOPSOWL_DATA, 68
 - SKUA_OPT_DATA, 78
- skua_opt.cpp, 142
 - eval_SKUA_Uptake, 143
 - initial_guess_SKUA, 143
 - SKUA_OPT_set_y, 143
 - SKUA_OPTIMIZE, 143
- skua_opt.h, 143
 - eval_SKUA_Uptake, 143
 - initial_guess_SKUA, 143
 - SKUA_OPT_set_y, 143
 - SKUA_OPTIMIZE, 143
- Sn
 - FINCH_DATA, 26
- Snpl
 - FINCH_DATA, 26
- SolnTransform
 - Matrix, 47
- solve
 - FINCH_DATA, 26
- sorbed_molefraction
 - DOGFISH_PARAM, 19
 - MONKFISH_PARAM, 56
- sorption_bc
 - MONKFISH_PARAM, 56
- Speciation_Test01_Data, 79
 - C, 80
 - CT, 80
 - Jacobian, 80
 - logC, 80
 - logKa1, 80
 - logKa2, 80
 - logKw, 80
 - N, 80
 - NaT, 80
 - NumJac, 80
 - x, 80
- Speciation_Test01_Function
 - sandbox.cpp, 126
 - sandbox.h, 127

- Speciation_Test01_Guess
 - sandbox.cpp, [126](#)
 - sandbox.h, [127](#)
- Speciation_Test01_Jacobian
 - sandbox.cpp, [126](#)
 - sandbox.h, [127](#)
- Speciation_Test01_MatVec
 - sandbox.cpp, [126](#)
 - sandbox.h, [127](#)
- SpeciationCurve
 - SHARK_DATA, [74](#)
- species
 - DOGFISH_PARAM, [19](#)
 - MasterSpeciesList, [44](#)
 - MONKFISH_PARAM, [56](#)
- species_dat
 - MIXED_GAS, [49](#)
- species_index
 - Mechanism, [48](#)
 - UnsteadyReaction, [91](#)
- speciesName
 - MasterSpeciesList, [44](#)
 - SCOPSOWL_PARAM_DATA, [72](#)
 - SKUA_PARAM, [79](#)
- specific_heat
 - PURE_GAS, [63](#)
- Spherical
 - finch.h, [108](#)
- sphericalAvg
 - Matrix, [47](#)
- sphericalBCFill
 - Matrix, [47](#)
- state
 - SubHeader, [82](#)
- SteadyState
 - FINCH_DATA, [26](#)
- steadystate
 - SHARK_DATA, [74](#)
- steps
 - GMRESLP_DATA, [30](#)
- Stoichiometric
 - Reaction, [66](#)
- string_parse_error
 - error.h, [104](#)
- Sub_Map
 - Header, [38](#)
- SubHeader, [80](#)
 - ~SubHeader, [81](#)
 - addPair, [81](#)
 - alias, [82](#)
 - clear, [81](#)
 - Data_Map, [82](#)
 - DisplayContents, [82](#)
 - getAlias, [82](#)
 - getMap, [82](#)
 - getName, [82](#)
 - getState, [82](#)
 - isAlias, [82](#)
 - isAnchor, [82](#)
 - name, [82](#)
 - operator=, [82](#)
 - setAlias, [82](#)
 - setName, [82](#)
 - setNameAliasPair, [82](#)
 - setState, [82](#)
 - state, [82](#)
 - SubHeader, [81](#)
 - SubHeader, [81](#)
- sum
 - ARNOLDI_DATA, [7](#)
 - GMRESRP_DATA, [33](#)
 - Matrix, [47](#)
- Sum_Delta
 - MassBalance, [42](#)
- SurfDiff
 - SCOPSOWL_DATA, [68](#)
- surface_concentration
 - DOGFISH_PARAM, [19](#)
- Sutherland_Const
 - PURE_GAS, [63](#)
- Sutherland_Temp
 - PURE_GAS, [63](#)
- Sutherland_Viscosity
 - PURE_GAS, [64](#)
- Symbol
 - Atom, [10](#)
- Sys
 - SYSTEM_DATA, [83](#)
- sys_dat
 - MAGPIE_DATA, [41](#)
- T
 - FINCH_DATA, [26](#)
 - SYSTEM_DATA, [83](#)
- t
 - BiCGSTAB_DATA, [12](#)
 - FINCH_DATA, [26](#)
 - SCOPSOWL_DATA, [68](#)
 - SCOPSOWL_OPT_DATA, [70](#)
 - SKUA_DATA, [76](#)
 - SKUA_OPT_DATA, [78](#)
- TEST
 - ui.h, [150](#)
- t_count
 - SHARK_DATA, [74](#)
- t_counter
 - DOGFISH_DATA, [18](#)
 - MONKFISH_DATA, [54](#)
 - SCOPSOWL_DATA, [68](#)
 - SKUA_DATA, [76](#)
- t_old
 - FINCH_DATA, [26](#)
 - SCOPSOWL_DATA, [68](#)
 - SKUA_DATA, [76](#)
- t_out
 - SHARK_DATA, [74](#)
- t_print

- DOGFISH_DATA, [18](#)
- MONKFISH_DATA, [55](#)
- SCOPSOWL_DATA, [68](#)
- SKUA_DATA, [76](#)
- t_rand
 - TRAJECTORY_DATA, [86](#)
- TANGENTIAL_FORCE
 - Trajectory.cpp, [145](#)
 - Trajectory.h, [146](#)
- TRAJECTORY_DATA, [84](#)
 - a, [85](#)
 - A_separator, [85](#)
 - A_wire, [85](#)
 - b, [85](#)
 - B0, [85](#)
 - beta, [85](#)
 - Cap, [85](#)
 - chi_p, [85](#)
 - dX, [85](#)
 - dY, [85](#)
 - dt, [85](#)
 - eta, [85](#)
 - H, [85](#)
 - H0, [85](#)
 - Hamaker, [85](#)
 - k, [85](#)
 - L, [85](#)
 - L_wire, [85](#)
 - M, [85](#)
 - m_rand, [85](#)
 - mp, [85](#)
 - Ms, [85](#)
 - mu_0, [85](#)
 - n_rand, [85](#)
 - POL, [86](#)
 - porosity, [86](#)
 - q_bar, [86](#)
 - Q_in, [86](#)
 - rho_f, [86](#)
 - rho_p, [86](#)
 - Rs, [86](#)
 - s_rand, [86](#)
 - sigma_m, [86](#)
 - sigma_n, [86](#)
 - sigma_v, [86](#)
 - sigma_vz, [86](#)
 - sigma_z, [86](#)
 - t_rand, [86](#)
 - Temp, [86](#)
 - V0, [86](#)
 - V_separator, [86](#)
 - V_wire, [86](#)
 - X, [86](#)
 - Y, [86](#)
 - Y_initial, [86](#)
- Table
 - PeriodicTable, [59](#)
- Temp
 - TRAJECTORY_DATA, [86](#)
- temperature
 - SHARK_DATA, [74](#)
- temperature_affinity
 - UnsteadyReaction, [91](#)
- tempy
 - SCOPSOWL_DATA, [68](#)
- tensor_out_of_bounds
 - error.h, [104](#)
- term_precon
 - GMRESR_DATA, [31](#)
- terminal_precon
 - GMRESR_DATA, [31](#)
- test
 - ui.cpp, [148](#)
 - ui.h, [151](#)
- test_loop
 - ui.cpp, [148](#)
 - ui.h, [151](#)
- theoretical_darken_Dc
 - skua.cpp, [140](#)
 - skua.h, [142](#)
- time
 - DOGFISH_DATA, [18](#)
 - MONKFISH_DATA, [55](#)
 - SHARK_DATA, [75](#)
- time_old
 - DOGFISH_DATA, [18](#)
 - MONKFISH_DATA, [55](#)
 - SHARK_DATA, [75](#)
- time_step
 - UnsteadyReaction, [91](#)
- TimeAdaptivity
 - SHARK_DATA, [75](#)
- timesteps
 - SHARK_DATA, [75](#)
- token_parser
 - yaml_cpp_class, [94](#)
- tol_abs
 - BiCGSTAB_DATA, [12](#)
 - CGS_DATA, [14](#)
 - FINCH_DATA, [26](#)
 - GCR_DATA, [29](#)
 - GMRESLP_DATA, [30](#)
 - GMRESRP_DATA, [33](#)
 - PCG_DATA, [58](#)
 - PICARD_DATA, [60](#)
- tol_rel
 - BiCGSTAB_DATA, [12](#)
 - CGS_DATA, [14](#)
 - FINCH_DATA, [26](#)
 - GCR_DATA, [29](#)
 - GMRESLP_DATA, [30](#)
 - GMRESRP_DATA, [33](#)
 - PCG_DATA, [58](#)
 - PICARD_DATA, [60](#)
- total_density
 - MIXED_GAS, [49](#)

- total_dyn_vis
 - MIXED_GAS, [49](#)
- total_eval
 - GSTA_OPT_DATA, [35](#)
 - SCOPSOWL_OPT_DATA, [70](#)
 - SKUA_OPT_DATA, [78](#)
 - SYSTEM_DATA, [84](#)
- total_iter
 - FINCH_DATA, [26](#)
 - GCR_DATA, [29](#)
 - GMRESR_DATA, [31](#)
- total_molecular_weight
 - MIXED_GAS, [49](#)
- total_pressure
 - MIXED_GAS, [49](#)
 - SCOPSOWL_DATA, [68](#)
- total_sorption
 - DOGFISH_DATA, [18](#)
 - MONKFISH_DATA, [55](#)
- total_sorption_old
 - DOGFISH_DATA, [19](#)
 - MONKFISH_DATA, [55](#)
- total_specific_heat
 - MIXED_GAS, [49](#)
- total_steps
 - DOGFISH_DATA, [19](#)
 - MONKFISH_DATA, [55](#)
 - SCOPSOWL_DATA, [68](#)
 - SKUA_DATA, [76](#)
- TotalConcentration
 - MassBalance, [42](#)
- totalsteps
 - SHARK_DATA, [75](#)
- trajectory
 - ui.h, [150](#)
- Trajectory.cpp, [143](#)
 - Brown_RAD, [144](#)
 - Brown_THETA, [144](#)
 - CARTESIAN, [144](#)
 - DISPLACEMENT, [144](#)
 - Grav_R, [144](#)
 - Grav_T, [144](#)
 - LOCATION, [144](#)
 - Magnetic_R, [144](#)
 - Magnetic_T, [144](#)
 - Number_Generator, [144](#)
 - POLAR, [145](#)
 - RADIAL_FORCE, [145](#)
 - Removal_Efficiency, [145](#)
 - Run_Trajectory, [145](#)
 - TANGENTIAL_FORCE, [145](#)
 - Trajectory_SetupConstants, [145](#)
 - V_RAD, [145](#)
 - V_THETA, [145](#)
 - Van_R, [145](#)
- Trajectory.h, [145](#)
 - Brown_RAD, [146](#)
 - Brown_THETA, [146](#)
- CARTESIAN, [146](#)
- DISPLACEMENT, [146](#)
- Grav_R, [146](#)
- Grav_T, [146](#)
- LOCATION, [146](#)
- Magnetic_R, [146](#)
- Magnetic_T, [146](#)
- Number_Generator, [146](#)
- POLAR, [146](#)
- RADIAL_FORCE, [146](#)
- Removal_Efficiency, [146](#)
- Run_Trajectory, [146](#)
- TANGENTIAL_FORCE, [146](#)
- Trajectory_SetupConstants, [146](#)
- V_RAD, [146](#)
- V_THETA, [147](#)
- Van_R, [147](#)
- Trajectory_SetupConstants
 - Trajectory.cpp, [145](#)
 - Trajectory.h, [146](#)
- transpose
 - Matrix, [47](#)
- transpose_dat
 - GCR_DATA, [29](#)
- transpose_multiply
 - Matrix, [47](#)
- tridiagonalFill
 - Matrix, [47](#)
- tridiagonalSolve
 - Matrix, [47](#)
- tridiagonalVectorFill
 - Matrix, [47](#)
- twoFifths
 - gsta_opt.cpp, [111](#)
 - gsta_opt.h, [113](#)
- type
 - ValueTypePair, [93](#)
- u
 - CGS_DATA, [14](#)
 - GCR_DATA, [29](#)
- UNKNOWN
 - yaml_wrapper.h, [153](#)
- u_star
 - FINCH_DATA, [27](#)
- u_temp
 - GCR_DATA, [29](#)
- uAverage
 - finch.cpp, [107](#)
 - finch.h, [110](#)
- uAvg
 - FINCH_DATA, [27](#)
- uAvg_old
 - FINCH_DATA, [27](#)
- UI_DATA, [86](#)
 - argc, [87](#)
 - argv, [87](#)
 - BasicUI, [87](#)
 - count, [87](#)

- Files, 87
- input_files, 87
- max, 87
- MissingArg, 87
- option, 87
- Path, 87
- path, 87
- user_input, 87
- value_type, 87
- UI_HPP_
 - ui.h, 150
- uIC
 - FINCH_DATA, 27
- uT
 - FINCH_DATA, 27
- uT_old
 - FINCH_DATA, 27
- uTotal
 - finch.cpp, 107
 - finch.h, 110
- ubest
 - FINCH_DATA, 27
- ui.h
 - CONTINUE, 150
 - dogfish, 150
 - EXECUTE, 150
 - EXIT, 150
 - eel, 150
 - egret, 150
 - finch, 150
 - gsta_opt, 150
 - HELP, 150
 - lark, 150
 - macaw, 150
 - magpie, 150
 - mola, 150
 - monkfish, 150
 - sandbox, 150
 - scops_opt, 150
 - scopsowl, 150
 - shark, 150
 - skua, 150
 - skua_opt, 150
 - TEST, 150
 - trajectory, 150
- ui.cpp, 147
 - allLower, 147
 - ai_help, 147
 - bui_help, 147
 - display_help, 147
 - display_version, 147
 - exec, 147
 - exec_loop, 148
 - exit, 148
 - help, 148
 - input, 148
 - invalid_input, 148
 - number_files, 148
 - path, 148
 - run_exec, 148
 - run_executable, 148
 - run_test, 148
 - test, 148
 - test_loop, 148
 - valid_addon_options, 148
 - valid_exec_string, 148
 - valid_input_execute, 148
 - valid_input_main, 148
 - valid_input_tests, 148
 - valid_test_string, 148
 - version, 148
- ui.h, 148
 - allLower, 150
 - ai_help, 150
 - bui_help, 150
 - display_help, 150
 - display_version, 150
 - ECO_EXECUTABLE, 150
 - ECO_VERSION, 150
 - exec, 150
 - exec_loop, 150
 - exec_option, 151
 - exit, 151
 - help, 151
 - input, 151
 - invalid_input, 151
 - number_files, 151
 - path, 151
 - run_exec, 151
 - run_executable, 151
 - run_test, 151
 - test, 151
 - test_loop, 151
 - UI_HPP_, 150
 - valid_addon_options, 151
 - valid_exec_string, 151
 - valid_input_execute, 151
 - valid_input_main, 151
 - valid_input_tests, 151
 - valid_options, 150
 - valid_test_string, 151
 - version, 151
- un
 - FINCH_DATA, 27
- unm1
 - FINCH_DATA, 27
- unp1
 - FINCH_DATA, 27
- unregistered_name
 - error.h, 104
- unstable_matrix
 - error.h, 104
- UnsteadyList
 - SHARK_DATA, 75
- UnsteadyPrecipitation, 87
- UnsteadyReaction, 88

- ~UnsteadyReaction, 89
- activation_energy, 91
- calculateEnergies, 89
- calculateEquilibrium, 89
- calculateRate, 89
- checkSpeciesEnergies, 89
- Display_Info, 90
- Eval_IC_Residual, 90
- Eval_ReactionRate, 90
- Eval_Residual, 90
- Explicit_Eval, 90
- forward_rate, 91
- forward_ref_rate, 91
- Get_ActivationEnergy, 90
- Get_Affinity, 90
- Get_Energy, 90
- Get_Enthalpy, 90
- Get_Entropy, 90
- Get_Equilibrium, 90
- Get_Forward, 90
- Get_ForwardRef, 90
- Get_InitialValue, 90
- Get_MaximumValue, 90
- Get_Reverse, 90
- Get_ReverseRef, 90
- Get_Species_Index, 90
- Get_Stoichiometric, 90
- Get_TimeStep, 90
- haveEquilibrium, 90
- HaveForRef, 91
- HaveForward, 91
- haveRate, 90
- HaveRevRef, 91
- HaveReverse, 91
- initial_value, 91
- Initialize_List, 90
- max_value, 91
- reverse_rate, 91
- reverse_ref_rate, 91
- Set_ActivationEnergy, 90
- Set_Affinity, 90
- Set_Energy, 90
- Set_Enthalpy, 90
- Set_EnthalpyANDEntropy, 91
- Set_Entropy, 91
- Set_Equilibrium, 91
- Set_Forward, 91
- Set_ForwardRef, 91
- Set_InitialValue, 91
- Set_MaximumValue, 91
- Set_Reverse, 91
- Set_ReverseRef, 91
- Set_Species_Index, 91
- Set_Stoichiometric, 91
- Set_TimeStep, 91
- species_index, 91
- temperature_affinity, 91
- time_step, 91
- UnsteadyReaction, 89
- UnsteadyReaction, 89
- uo
 - FINCH_DATA, 27
- Update
 - FINCH_DATA, 27
- update_arnoldi_solution
 - lark.cpp, 115
 - lark.h, 119
- upperHessenberg2Triangular
 - Matrix, 48
- upperHessenbergSolve
 - Matrix, 48
- upperTriangularSolve
 - Matrix, 48
- user_data
 - DOGFISH_DATA, 19
 - MONKFISH_DATA, 55
 - SCOPSOWL_DATA, 68
 - SKUA_DATA, 76
- user_input
 - UI_DATA, 87
- uz_I_E
 - FINCH_DATA, 27
- uz_I_I
 - FINCH_DATA, 27
- uz_lm1_E
 - FINCH_DATA, 27
- uz_lm1_I
 - FINCH_DATA, 27
- uz_lp1_E
 - FINCH_DATA, 27
- uz_lp1_I
 - FINCH_DATA, 27
- V
 - maggie.h, 123
- v
 - ARNOLDI_DATA, 7
 - BiCGSTAB_DATA, 12
 - CGS_DATA, 14
 - GMRESRP_DATA, 33
 - mSPD_DATA, 56
 - PJFNK_DATA, 62
- V0
 - TRAJECTORY_DATA, 86
- V_RAD
 - Trajectory.cpp, 145
 - Trajectory.h, 146
- V_THETA
 - Trajectory.cpp, 145
 - Trajectory.h, 147
- V_separator
 - TRAJECTORY_DATA, 86
- V_wire
 - TRAJECTORY_DATA, 86
- vIC
 - FINCH_DATA, 27
- valence_e

- Atom, [10](#)
- valid_act
 - shark.h, [137](#)
- valid_addon_options
 - ui.cpp, [148](#)
 - ui.h, [151](#)
- valid_exec_string
 - ui.cpp, [148](#)
 - ui.h, [151](#)
- valid_input_execute
 - ui.cpp, [148](#)
 - ui.h, [151](#)
- valid_input_main
 - ui.cpp, [148](#)
 - ui.h, [151](#)
- valid_input_tests
 - ui.cpp, [148](#)
 - ui.h, [151](#)
- valid_options
 - ui.h, [150](#)
- valid_test_string
 - ui.cpp, [148](#)
 - ui.h, [151](#)
- Value_Type
 - ValueTypePair, [93](#)
- value_type
 - UI_DATA, [87](#)
- ValueTypePair, [92](#)
 - ~ValueTypePair, [92](#)
 - assertType, [92](#)
 - DisplayPair, [93](#)
 - editPair, [93](#)
 - editValue, [93](#)
 - findType, [93](#)
 - getBool, [93](#)
 - getDouble, [93](#)
 - getInt, [93](#)
 - getPair, [93](#)
 - getString, [93](#)
 - getType, [93](#)
 - getValue, [93](#)
 - operator=, [93](#)
 - type, [93](#)
 - Value_Type, [93](#)
 - ValueTypePair, [92](#)
 - ValueTypePair, [92](#)
- Van_R
 - Trajectory.cpp, [145](#)
 - Trajectory.h, [147](#)
- vanAlbada_discretization
 - finch.cpp, [107](#)
 - finch.h, [110](#)
- vector_out_of_bounds
 - error.h, [104](#)
- velocity
 - MIXED_GAS, [49](#)
- version
 - ui.cpp, [148](#)
- ui.h, [151](#)
- Vk
 - ARNOLDI_DATA, [7](#)
 - GMRESRP_DATA, [33](#)
- vn
 - FINCH_DATA, [27](#)
- vnp1
 - FINCH_DATA, [27](#)
- vo
 - FINCH_DATA, [27](#)
- w
 - ARNOLDI_DATA, [8](#)
 - CGS_DATA, [14](#)
 - GMRESRP_DATA, [33](#)
- weight
 - Mechanism, [48](#)
- weightedAvg
 - gsta_opt.cpp, [111](#)
 - gsta_opt.h, [113](#)
- X
 - TRAJECTORY_DATA, [86](#)
- x
 - BiCGSTAB_DATA, [13](#)
 - CGS_DATA, [14](#)
 - EX09_DATA, [21](#)
 - GCR_DATA, [29](#)
 - GMRESLP_DATA, [30](#)
 - GMRESRP_DATA, [33](#)
 - GPAST_DATA, [34](#)
 - PCG_DATA, [58](#)
 - PJFNK_DATA, [62](#)
 - Speciation_Test01_Data, [80](#)
- x0
 - PICARD_DATA, [60](#)
- X_new
 - SHARK_DATA, [75](#)
- X_old
 - SHARK_DATA, [75](#)
- xiC
 - SCOPSOWL_PARAM_DATA, [72](#)
 - SKUA_PARAM, [79](#)
- xk
 - BACKTRACK_DATA, [11](#)
- xn
 - SKUA_PARAM, [79](#)
- xnp1
 - SKUA_PARAM, [79](#)
- Y
 - TRAJECTORY_DATA, [86](#)
- y
 - BiCGSTAB_DATA, [13](#)
 - GMRESRP_DATA, [33](#)
 - GPAST_DATA, [34](#)
 - SCOPSOWL_DATA, [68](#)
 - SKUA_DATA, [76](#)
- y_base

- SCOPSOWL_OPT_DATA, 70
- SKUA_OPT_DATA, 78
- y_eff
 - SKUA_PARAM, 79
- Y_initial
 - TRAJECTORY_DATA, 86
- YAML_CPP_TEST
 - yaml_wrapper.cpp, 151
 - yaml_wrapper.h, 153
- YAML_WRAPPER_TESTS
 - yaml_wrapper.cpp, 151
 - yaml_wrapper.h, 153
- yaml_wrapper.h
 - ALIAS, 153
 - ANCHOR, 153
 - BOOLEAN, 152
 - DOUBLE, 153
 - INT, 153
 - NONE, 153
 - STRING, 152
 - UNKNOWN, 153
- yaml_cpp_class, 93
 - ~yaml_cpp_class, 94
 - cleanup, 94
 - current_token, 94
 - DisplayContents, 94
 - executeYamlRead, 94
 - file_name, 94
 - getYamlWrapper, 94
 - input_file, 94
 - previous_token, 94
 - readInputFile, 94
 - setInputFile, 94
 - token_parser, 94
 - yaml_cpp_class, 94
 - yaml_wrapper, 94
 - yaml_cpp_class, 94
- yaml_object
 - SHARK_DATA, 75
- yaml_wrapper
 - yaml_cpp_class, 94
- yaml_wrapper.cpp, 151
 - YAML_CPP_TEST, 151
- yaml_wrapper.h, 152
 - data_type, 152
 - header_state, 152, 153
 - YAML_CPP_TEST, 153
- YamlWrapper, 94
 - ~YamlWrapper, 95
 - addDocKey, 95
 - begin, 95
 - changeKey, 95
 - clear, 95
 - copyAnchor2Alias, 96
 - DisplayContents, 96
 - Doc_Map, 96
 - end, 96
 - getAnchoredDoc, 96
 - getDocFromHeadAlias, 96
 - getDocFromSubAlias, 96
 - getDocMap, 96
 - getDocument, 96
 - operator(), 96
 - operator=, 96
 - resetKeys, 96
 - revalidateAllKeys, 96
 - size, 96
 - YamlWrapper, 95
 - YamlWrapper, 95
- yk
 - ARNOLDI_DATA, 8
- Z
 - magpie.h, 123
- z
 - BiCGSTAB_DATA, 13
 - CGS_DATA, 14
 - PCG_DATA, 58
- z_old
 - PCG_DATA, 58
- zero_vector
 - error.h, 104
- zeros
 - Matrix, 48