

LUMA

1.1.1-alpha

Generated by Doxygen 1.8.11

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	BFLBody Class Reference	7
4.1.1	Constructor & Destructor Documentation	8
4.1.1.1	BFLBody(void)	8
4.1.1.2	~BFLBody(void)	8
4.1.1.3	BFLBody(PCpts *_PCpts, GridObj *g_hierarchy)	8
4.1.2	Member Function Documentation	8
4.1.2.1	computeQ(int i, int j, int k, int N_lim, int M_lim, int K_lim, GridObj *g)	8
4.1.2.2	computeQ(int i, int j, int N_lim, int M_lim, GridObj *g)	8
4.1.3	Friends And Related Function Documentation	8
4.1.3.1	GridObj	8
4.1.4	Member Data Documentation	8
4.1.4.1	Q	8
4.2	BFLMarker Class Reference	8
4.2.1	Constructor & Destructor Documentation	9
4.2.1.1	BFLMarker(void)	9

4.2.1.2	~BFLMarker(void)	9
4.2.1.3	BFLMarker(double x, double y, double z)	9
4.2.2	Friends And Related Function Documentation	9
4.2.2.1	BFLBody	9
4.3	Body< MarkerType > Class Template Reference	9
4.3.1	Constructor & Destructor Documentation	10
4.3.1.1	Body(void)	10
4.3.1.2	~Body(void)	10
4.3.1.3	Body(GridObj *g)	10
4.3.2	Member Function Documentation	10
4.3.2.1	addMarker(double x, double y, double z)	10
4.3.2.2	getMarkerData(double x, double y, double z)	10
4.3.2.3	isInVoxel(double x, double y, double z, int curr_mark)	10
4.3.2.4	isVoxelMarkerVoxel(double x, double y, double z)	10
4.3.2.5	markerAdder(double x, double y, double z, int &curr_mark, std::vector< int > &counter)	10
4.3.3	Member Data Documentation	10
4.3.3.1	_Owner	10
4.3.3.2	closed_surface	10
4.3.3.3	markers	10
4.3.3.4	spacing	10
4.4	MpiManager::buffer_struct Struct Reference	10
4.4.1	Member Data Documentation	11
4.4.1.1	level	11
4.4.1.2	region	11
4.4.1.3	size	11
4.5	GridObj Class Reference	11
4.5.1	Constructor & Destructor Documentation	13
4.5.1.1	GridObj()	13
4.5.1.2	GridObj(int level)	13
4.5.1.3	GridObj(int RegionNumber, GridObj &pGrid)	13

4.5.1.4	<code>GridObj(int level, std::vector< int > local_size, std::vector< std::vector< int > > GlobalLimsInd, std::vector< std::vector< double > > GlobalLimsPos)</code>	13
4.5.1.5	<code>~GridObj()</code>	13
4.5.2	Member Function Documentation	13
4.5.2.1	<code>bc_applyBfl(int i, int j, int k)</code>	13
4.5.2.2	<code>bc_applyBounceBack(int label, int i, int j, int k, int N_lim, int M_lim, int K_lim)</code> . . .	13
4.5.2.3	<code>bc_applyExtrapolation(int label, int i, int j, int k, int M_lim, int K_lim)</code>	13
4.5.2.4	<code>bc_applyNrbc(int i, int j, int k)</code>	13
4.5.2.5	<code>bc_applyRegularised(int label, int i, int j, int k, int N_lim, int M_lim, int K_lim)</code> . . .	13
4.5.2.6	<code>bc_applySpecReflect(int label, int i, int j, int k, int N_lim, int M_lim, int K_lim)</code> . . .	14
4.5.2.7	<code>bc_applyZouHe(int label, int i, int j, int k, int M_lim, int K_lim)</code>	14
4.5.2.8	<code>bc_getWallDensityForRBC(std::vector< double > &ftmp, int normal, int i, int j, int k, int M_lim, int K_lim)</code>	14
4.5.2.9	<code>bc_solidSiteReset()</code>	14
4.5.2.10	<code>io_hdf5(double tval)</code>	14
4.5.2.11	<code>io_lite(double tval, std::string Tag)</code>	14
4.5.2.12	<code>io_probeOutput()</code>	14
4.5.2.13	<code>io_restart(bool IO_flag)</code>	14
4.5.2.14	<code>io_textout(std::string output_tag)</code>	14
4.5.2.15	<code>LBM_addSubGrid(int RegionNumber)</code>	14
4.5.2.16	<code>LBM_boundary(int bc_type_flag)</code>	14
4.5.2.17	<code>LBM_coalesce(int RegionNumber)</code>	14
4.5.2.18	<code>LBM_collide()</code>	14
4.5.2.19	<code>LBM_collide(int i, int j, int k, int v, int M_lim, int K_lim)</code>	14
4.5.2.20	<code>LBM_explode(int RegionNumber)</code>	14
4.5.2.21	<code>LBM_forcegrid(bool reset_flag)</code>	14
4.5.2.22	<code>LBM_init_getInletProfile()</code>	14
4.5.2.23	<code>LBM_initBoundLab()</code>	14
4.5.2.24	<code>LBM_initGrid()</code>	14
4.5.2.25	<code>LBM_initGrid(std::vector< int > local_size, std::vector< std::vector< int > > GlobalLimsInd, std::vector< std::vector< double > > GlobalLimsPos)</code>	14
4.5.2.26	<code>LBM_initRefinedLab(GridObj &pGrid)</code>	14

4.5.2.27	LBM_initRho()	14
4.5.2.28	LBM_initSolidLab()	15
4.5.2.29	LBM_initSubGrid(GridObj &pGrid)	15
4.5.2.30	LBM_initVelocity()	15
4.5.2.31	LBM_kbcCollide(int i, int j, int k, int M_lim, int K_lim, IVector< double > &f_new)	15
4.5.2.32	LBM_macro()	15
4.5.2.33	LBM_macro(int i, int j, int k)	15
4.5.2.34	LBM_multi(bool IBM_flag)	15
4.5.2.35	LBM_stream()	15
4.5.3	Friends And Related Function Documentation	15
4.5.3.1	GridUtils	15
4.5.3.2	MpiManager	15
4.5.3.3	ObjectManager	15
4.5.4	Member Data Documentation	15
4.5.4.1	dt	15
4.5.4.2	LatTyp	15
4.5.4.3	level	15
4.5.4.4	nu	15
4.5.4.5	omega	15
4.5.4.6	t	15
4.5.4.7	timeav_mpi_overhead	15
4.5.4.8	timeav_timestep	15
4.5.4.9	XInd	15
4.5.4.10	XPos	15
4.5.4.11	YInd	15
4.5.4.12	YPos	15
4.5.4.13	ZInd	15
4.5.4.14	ZPos	15
4.6	GridUtils Class Reference	16
4.6.1	Detailed Description	17

4.6.2	Member Function Documentation	17
4.6.2.1	add(std::vector< double > a, std::vector< double > b)	17
4.6.2.2	createOutputDirectory(std::string path_str)	17
4.6.2.3	crossprod(std::vector< double > vec1, std::vector< double > vec2)	17
4.6.2.4	dotprod(std::vector< double > vec1, std::vector< double > vec2)	17
4.6.2.5	downToLimit(NumType x, NumType limit)	17
4.6.2.6	factorial(NumType n)	17
4.6.2.7	getCoarseIndices(int fine_i, int x_start, int fine_j, int y_start, int fine_k, int z_start)	17
4.6.2.8	getFineIndices(int coarse_i, int x_start, int coarse_j, int y_start, int coarse_k, int z_start)	17
4.6.2.9	getGrid(GridObj *&Grids, int level, int region, GridObj *&ptr)	17
4.6.2.10	getOpposite(int direction)	17
4.6.2.11	global_to_local(int i, int j, int k, GridObj *g, std::vector< NumType > &locals)	17
4.6.2.12	hasThisSubGrid(const GridObj &pGrid, int RegNum)	18
4.6.2.13	indexToPosition(int index, double dx)	18
4.6.2.14	isOffGrid(int i, int j, int k, int N_lim, int M_lim, int K_lim, GridObj &g)	18
4.6.2.15	isOnRecvLayer(double pos_x, double pos_y, double pos_z)	18
4.6.2.16	isOnRecvLayer(double site_position, enum eCartesianDirection xyz, enum eMinMax minmax)	18
4.6.2.17	isOnSenderLayer(double pos_x, double pos_y, double pos_z)	18
4.6.2.18	isOnSenderLayer(double site_position, enum eCartesianDirection xyz, enum eMinMax minmax)	18
4.6.2.19	isOnThisRank(int gi, int gj, int gk, const GridObj &pGrid)	18
4.6.2.20	isOnThisRank(int gl, enum eCartesianDirection xyz, const GridObj &pGrid)	18
4.6.2.21	isOverlapPeriodic(int i, int j, int k, const GridObj &pGrid)	18
4.6.2.22	linspace(double min, double max, int n)	18
4.6.2.23	local_to_global(int i, int j, int k, GridObj *g, std::vector< NumType > &globals)	18
4.6.2.24	matrix_multiply(const std::vector< std::vector< double > > &A, const std::vector< double > &x)	18
4.6.2.25	onespace(int min, int max)	18
4.6.2.26	stridedCopy(NumType *dest, NumType *src, size_t block, size_t offset, size_t stride, size_t count, size_t buf_offset=0)	18
4.6.2.27	subtract(std::vector< double > a, std::vector< double > b)	18

4.6.2.28	<code>upToZero(NumType x)</code>	18
4.6.2.29	<code>vecmultiply(double scalar, std::vector< double > vec)</code>	18
4.6.2.30	<code>vecnorm(double vec[])</code>	18
4.6.2.31	<code>vecnorm(double val1, double val2)</code>	18
4.6.2.32	<code>vecnorm(double val1, double val2, double val3)</code>	18
4.6.2.33	<code>vecnorm(std::vector< double > vec)</code>	19
4.6.2.34	<code>vecnorm(NumType a1, NumType a2, NumType a3)</code>	19
4.6.2.35	<code>vecnorm(NumType a1, NumType a2)</code>	19
4.6.3	Member Data Documentation	19
4.6.3.1	<code>dir_reflect</code>	19
4.6.3.2	<code>logfile</code>	19
4.6.3.3	<code>path_str</code>	19
4.7	IBody Class Reference	19
4.7.1	Detailed Description	20
4.7.2	Constructor & Destructor Documentation	20
4.7.2.1	<code>IBody(void)</code>	20
4.7.2.2	<code>~IBody(void)</code>	20
4.7.2.3	<code>IBody(GridObj *g)</code>	20
4.7.3	Member Function Documentation	20
4.7.3.1	<code>addMarker(double x, double y, double z, bool flex_rigid)</code>	20
4.7.3.2	<code>makeBody(double radius, std::vector< double > centre, bool flex_rigid, bool moving, int group)</code>	21
4.7.3.3	<code>makeBody(std::vector< double > width_length_depth, std::vector< double > angles, std::vector< double > centre, bool flex_rigid, bool deform, int group)</code>	21
4.7.3.4	<code>makeBody(int numbermarkers, std::vector< double > start_point, double fil_← length, std::vector< double > angles, std::vector< int > BCs, bool flex_rigid, bool deform, int group)</code>	21
4.7.3.5	<code>makeBody(std::vector< double > width_length, double angle, std::vector< dou- ble > centre, bool flex_rigid, bool deform, int group, bool plate)</code>	21
4.7.3.6	<code>makeBody(PCpts *_PCpts)</code>	21
4.7.4	Friends And Related Function Documentation	21
4.7.4.1	<code>ObjectManager</code>	21
4.7.5	Member Data Documentation	21

4.7.5.1	BCs	21
4.7.5.2	deformable	21
4.7.5.3	delta_rho	21
4.7.5.4	flex_rigid	21
4.7.5.5	flexural_rigidity	21
4.7.5.6	groupID	21
4.7.5.7	tension	21
4.8	IBMarker Class Reference	21
4.8.1	Constructor & Destructor Documentation	22
4.8.1.1	IBMarker(void)	22
4.8.1.2	~IBMarker(void)	22
4.8.1.3	IBMarker(double xPos, double yPos, double zPos, bool flex_rigid=false)	22
4.8.2	Friends And Related Function Documentation	22
4.8.2.1	IBBody	22
4.8.2.2	ObjectManager	22
4.8.3	Member Data Documentation	22
4.8.3.1	deltaval	22
4.8.3.2	desired_vel	22
4.8.3.3	dilation	22
4.8.3.4	epsilon	23
4.8.3.5	flex_rigid	23
4.8.3.6	fluid_vel	23
4.8.3.7	force_xyz	23
4.8.3.8	local_area	23
4.8.3.9	position_old	23
4.9	IVector< GenTyp > Class Template Reference	23
4.9.1	Constructor & Destructor Documentation	24
4.9.1.1	IVector()	24
4.9.1.2	~IVector()	24
4.9.1.3	IVector(size_t size, GenTyp val)	24

4.9.2	Member Function Documentation	24
4.9.2.1	operator()(size_t i, size_t j, size_t k, size_t v, size_t j_max, size_t k_max, size_t v_max)	24
4.9.2.2	operator()(size_t i, size_t j, size_t k, size_t j_max, size_t k_max)	24
4.9.2.3	operator()(size_t i, size_t j, size_t j_max)	24
4.10	MpiManager::layer_edges Struct Reference	24
4.10.1	Member Data Documentation	24
4.10.1.1	X	24
4.10.1.2	Y	24
4.10.1.3	Z	24
4.11	Marker Class Reference	25
4.11.1	Detailed Description	25
4.11.2	Constructor & Destructor Documentation	25
4.11.2.1	Marker(void)	25
4.11.2.2	~Marker(void)	25
4.11.2.3	Marker(double x, double y, double z)	25
4.11.3	Member Data Documentation	25
4.11.3.1	position	25
4.11.3.2	supp_i	25
4.11.3.3	supp_j	25
4.11.3.4	supp_k	25
4.11.3.5	support_rank	25
4.12	MarkerData Class Reference	26
4.12.1	Constructor & Destructor Documentation	26
4.12.1.1	MarkerData(int i, int j, int k, double x, double y, double z, int ID)	26
4.12.1.2	MarkerData(void)	26
4.12.1.3	~MarkerData(void)	26
4.12.2	Member Data Documentation	26
4.12.2.1	i	26
4.12.2.2	ID	26
4.12.2.3	j	26

4.12.2.4	k	26
4.12.2.5	x	26
4.12.2.6	y	26
4.12.2.7	z	26
4.13	MpiManager Class Reference	27
4.13.1	Member Function Documentation	28
4.13.1.1	destroyInstance()	28
4.13.1.2	getInstance()	28
4.13.1.3	mpi_buffer_pack(int dir, GridObj *g)	28
4.13.1.4	mpi_buffer_size()	28
4.13.1.5	mpi_buffer_size_recv(GridObj *g)	28
4.13.1.6	mpi_buffer_size_send(GridObj *g)	28
4.13.1.7	mpi_buffer_unpack(int dir, GridObj *g)	28
4.13.1.8	mpi_buildCommunicators()	28
4.13.1.9	mpi_communicate(int level, int regnum)	28
4.13.1.10	mpi_getOpposite(int direction)	28
4.13.1.11	mpi_gridbuild()	28
4.13.1.12	mpi_init()	28
4.13.1.13	mpi_writeout_buf(std::string filename, int dir)	28
4.13.2	Member Data Documentation	28
4.13.2.1	buffer_recv_info	28
4.13.2.2	buffer_send_info	28
4.13.2.3	f_buffer_recv	28
4.13.2.4	f_buffer_send	28
4.13.2.5	global_dims	29
4.13.2.6	global_edge_ind	29
4.13.2.7	global_edge_pos	29
4.13.2.8	Grids	29
4.13.2.9	local_size	29
4.13.2.10	logout	29

4.13.2.11 MPI_cartlab	29
4.13.2.12 MPI_coords	29
4.13.2.13 MPI_dims	29
4.13.2.14 my_rank	29
4.13.2.15 neighbour_coords	29
4.13.2.16 neighbour_rank	29
4.13.2.17 num_ranks	29
4.13.2.18 p_data	29
4.13.2.19 recv_layer_pos	29
4.13.2.20 recv_stat	29
4.13.2.21 send_requests	29
4.13.2.22 send_stat	29
4.13.2.23 sender_layer_pos	29
4.13.2.24 subGrid_comm	29
4.13.2.25 world_comm	29
4.14 ObjectManager Class Reference	30
4.14.1 Member Function Documentation	31
4.14.1.1 bfl_build_body(int body_type)	31
4.14.1.2 bfl_build_body(PCpts *_PCpts)	31
4.14.1.3 destroyInstance()	31
4.14.1.4 getInstance()	31
4.14.1.5 getInstance(GridObj *g)	31
4.14.1.6 getVoxInd(double x, double y, double z)	31
4.14.1.7 getVoxInd(double p)	31
4.14.1.8 ibm_apply(GridObj &g)	31
4.14.1.9 ibm_banbks(double **a, long n, int m1, int m2, double **al, unsigned long indx[], double b[])	31
4.14.1.10 ibm_bandec(double **a, long n, int m1, int m2, double **al, unsigned long indx[], double *d)	31
4.14.1.11 ibm_bicgstab(std::vector< std::vector< double > > &Amatrix, std::vector< double > &bVector, std::vector< double > &epsilon, double tolerance, int maxiterations)	31
4.14.1.12 ibm_build_body(int body_type)	31

4.14.1.13	<code>ibm_build_body(PCpts *_PCpts, GridObj *owner)</code>	31
4.14.1.14	<code>ibm_computeforce(int ib, GridObj &g)</code>	31
4.14.1.15	<code>ibm_deltakernel(double rad, double dilation)</code>	31
4.14.1.16	<code>ibm_findepsilon(int ib, GridObj &g)</code>	31
4.14.1.17	<code>ibm_findsupport(int ib, int m, GridObj &g)</code>	31
4.14.1.18	<code>ibm_initialise(GridObj &g)</code>	31
4.14.1.19	<code>ibm_interpol(int ib, GridObj &g)</code>	31
4.14.1.20	<code>ibm_jacowire(int ib, GridObj &g)</code>	32
4.14.1.21	<code>ibm_move_bodies(GridObj &g)</code>	32
4.14.1.22	<code>ibm_position_update(int ib, GridObj &g)</code>	32
4.14.1.23	<code>ibm_position_update_grp(int group, GridObj &g)</code>	32
4.14.1.24	<code>ibm_spread(int ib, GridObj &g)</code>	32
4.14.1.25	<code>io_readInCloud(PCpts *_PCpts, eObjectType objtype)</code>	32
4.14.1.26	<code>io_restart(bool IO_flag, int level)</code>	32
4.14.1.27	<code>io_vtk_IBwriter(double tval)</code>	32
4.14.1.28	<code>io_write_body_pos(int timestep)</code>	32
4.14.1.29	<code>io_write_lift_drag(int timestep)</code>	32
4.14.2	Friends And Related Function Documentation	32
4.14.2.1	<code>GridObj</code>	32
4.15	<code>PCpts</code> Class Reference	32
4.15.1	Constructor & Destructor Documentation	33
4.15.1.1	<code>PCpts(void)</code>	33
4.15.1.2	<code>~PCpts(void)</code>	33
4.15.2	Member Data Documentation	33
4.15.2.1	<code>x</code>	33
4.15.2.2	<code>y</code>	33
4.15.2.3	<code>z</code>	33
4.16	<code>MpiManager::phdf5_struct</code> Struct Reference	33
4.16.1	Member Data Documentation	34
4.16.1.1	<code>halo_max</code>	34
4.16.1.2	<code>halo_min</code>	34
4.16.1.3	<code>i_end</code>	34
4.16.1.4	<code>i_start</code>	34
4.16.1.5	<code>j_end</code>	34
4.16.1.6	<code>j_start</code>	34
4.16.1.7	<code>k_end</code>	34
4.16.1.8	<code>k_start</code>	34
4.16.1.9	<code>level</code>	34
4.16.1.10	<code>region</code>	34
4.16.1.11	<code>writable_data_count</code>	34

5 File Documentation	35
5.1 BFLBody.cpp File Reference	35
5.2 BFLBody.h File Reference	35
5.3 BFLMarker.cpp File Reference	35
5.4 BFLMarker.h File Reference	35
5.5 Body.h File Reference	36
5.6 definitions.h File Reference	36
5.6.1 Macro Definition Documentation	38
5.6.1.1 L_a_x	38
5.6.1.2 L_a_y	38
5.6.1.3 L_a_z	38
5.6.1.4 L_b_x	38
5.6.1.5 L_b_y	38
5.6.1.6 L_b_z	38
5.6.1.7 L_bfl_length	38
5.6.1.8 L_bfl_length_ref	38
5.6.1.9 L_bfl_on_grid_lev	39
5.6.1.10 L_bfl_on_grid_reg	39
5.6.1.11 L_bfl_scale_direction	39
5.6.1.12 L_block_on_grid_lev	39
5.6.1.13 L_block_on_grid_reg	39
5.6.1.14 L_block_x_max	39
5.6.1.15 L_block_x_min	39
5.6.1.16 L_block_y_max	39
5.6.1.17 L_block_y_min	39
5.6.1.18 L_block_z_max	39
5.6.1.19 L_block_z_min	39
5.6.1.20 L_BUILD_FOR_MPI	39
5.6.1.21 L_centre_bfl_z	39
5.6.1.22 L_centre_ibb_z	39

5.6.1.23	L_centre_object_z	39
5.6.1.24	L_CHEAP_NEAREST_NODE_DETECTION	39
5.6.1.25	L_dims	39
5.6.1.26	L_end_BC	39
5.6.1.27	L_FREESTREAM_TUNNEL	39
5.6.1.28	L_grav_direction	39
5.6.1.29	L_grav_force	39
5.6.1.30	L_HDF5_OUTPUT	39
5.6.1.31	L_ibt_angle_horz	39
5.6.1.32	L_ibt_angle_vert	40
5.6.1.33	L_ibt_d	40
5.6.1.34	L_ibt_deform	40
5.6.1.35	L_ibt_delta_rho	40
5.6.1.36	L_ibt_EI	40
5.6.1.37	L_ibt_filament_length	40
5.6.1.38	L_ibt_filament_start_x	40
5.6.1.39	L_ibt_filament_start_y	40
5.6.1.40	L_ibt_filament_start_z	40
5.6.1.41	L_ibt_flex_rigid	40
5.6.1.42	L_ibt_l	40
5.6.1.43	L_ibt_length	40
5.6.1.44	L_ibt_length_ref	40
5.6.1.45	L_ibt_on_grid_lev	40
5.6.1.46	L_ibt_on_grid_reg	40
5.6.1.47	L_ibt_r	40
5.6.1.48	L_ibt_scale_direction	40
5.6.1.49	L_ibt_w	40
5.6.1.50	L_ibt_x	40
5.6.1.51	L_ibt_y	40
5.6.1.52	L_ibt_z	40

5.6.1.53	L_INLET_ON	40
5.6.1.54	L_K	40
5.6.1.55	L_M	41
5.6.1.56	L_MPI_dir	41
5.6.1.57	L_N	41
5.6.1.58	L_num_markers	41
5.6.1.59	L_NumLev	41
5.6.1.60	L_NumReg	41
5.6.1.61	L_nVels	41
5.6.1.62	L_object_length	41
5.6.1.63	L_object_length_ref	41
5.6.1.64	L_object_on_grid_lev	41
5.6.1.65	L_object_on_grid_reg	41
5.6.1.66	L_object_scale_direction	41
5.6.1.67	L_out_every	41
5.6.1.68	L_out_every_probe	41
5.6.1.69	L_OUTLET_ON	41
5.6.1.70	L_output_precision	41
5.6.1.71	L_PI	41
5.6.1.72	L_Re	41
5.6.1.73	L_restart_out_every	41
5.6.1.74	L_rho_in	41
5.6.1.75	L_SOLID_BLOCK_ON	41
5.6.1.76	L_start_BC	41
5.6.1.77	L_start_bfl_x	41
5.6.1.78	L_start_bfl_y	42
5.6.1.79	L_start_ibt_x	42
5.6.1.80	L_start_ibt_y	42
5.6.1.81	L_start_object_x	42
5.6.1.82	L_start_object_y	42

5.6.1.83	L_Timesteps	42
5.6.1.84	L_u_0x	42
5.6.1.85	L_u_0y	42
5.6.1.86	L_u_0z	42
5.6.1.87	L_u_max	42
5.6.1.88	L_u_ref	42
5.6.1.89	L_USE_KBC_COLLISION	42
5.6.1.90	L_wall_thickness	42
5.6.1.91	L_Xcores	42
5.6.1.92	L_Ycores	42
5.6.1.93	L_Zcores	42
5.6.1.94	LUMA_VERSION	42
5.6.2	Variable Documentation	42
5.6.2.1	nProbes	42
5.6.2.2	RefXend	42
5.6.2.3	RefXstart	42
5.6.2.4	RefYend	42
5.6.2.5	RefYstart	42
5.6.2.6	RefZend	43
5.6.2.7	RefZstart	43
5.6.2.8	xProbeLims	43
5.6.2.9	yProbeLims	43
5.6.2.10	zProbeLims	43
5.7	Gen_init_globalvars.cpp File Reference	43
5.7.1	Variable Documentation	43
5.7.1.1	c	43
5.7.1.2	cs	43
5.7.1.3	w	43
5.8	globalvars.h File Reference	44
5.8.1	Variable Documentation	44

5.8.1.1	c	44
5.8.1.2	cs	44
5.8.1.3	w	44
5.9	GridObj.cpp File Reference	44
5.10	GridObj.h File Reference	44
5.10.1	Enumeration Type Documentation	45
5.10.1.1	eBCType	45
5.10.1.2	eType	45
5.11	GridObj_init_grids.cpp File Reference	46
5.12	GridObj_ops_boundary.cpp File Reference	46
5.13	GridObj_ops_io.cpp File Reference	46
5.14	GridObj_ops_lbm.cpp File Reference	46
5.15	GridUtils.cpp File Reference	47
5.16	GridUtils.h File Reference	47
5.16.1	Enumeration Type Documentation	47
5.16.1.1	eCartesianDirection	47
5.16.1.2	eMinMax	47
5.17	hdf5luma.h File Reference	48
5.17.1	Macro Definition Documentation	48
5.17.1.1	H5_BUILT_AS_DYNAMIC_LIB	48
5.17.1.2	HDF5_EXT_SZIP	48
5.17.1.3	HDF5_EXT_ZLIB	48
5.17.2	Enumeration Type Documentation	48
5.17.2.1	eHdf5SlabType	48
5.17.3	Function Documentation	49
5.17.3.1	hdf5_writeDataSet(hid_t &memspace, hid_t &filespace, hid_t &dataset_id, e↔ Hdf5SlabType slab_type, int N_lim, int M_lim, int K_lim, int N_mod, int M_mod, int K_mod, GridObj *g, T *data, hid_t hdf_datatype, int TL_thickness, MPIManager↔ ::phdf5_struct hdf_data)	49
5.18	IBBody.cpp File Reference	49
5.19	IBBody.h File Reference	49
5.20	IBMarker.cpp File Reference	49

5.21	IBMarker.h File Reference	49
5.22	IVector.h File Reference	50
5.23	main_ibm.cpp File Reference	50
5.23.1	Function Documentation	50
5.23.1.1	main(int argc, char *argv[])	50
5.24	Marker.h File Reference	50
5.25	Mpi_buffer_pack.cpp File Reference	50
5.26	Mpi_buffer_size_rcv.cpp File Reference	51
5.27	Mpi_buffer_size_send.cpp File Reference	51
5.28	Mpi_buffer_unpk.cpp File Reference	51
5.29	MpiManager.cpp File Reference	51
5.30	MpiManager.h File Reference	51
5.30.1	Macro Definition Documentation	52
5.30.1.1	range_i_left	52
5.30.1.2	range_i_right	52
5.30.1.3	range_j_down	52
5.30.1.4	range_j_up	52
5.30.1.5	range_k_back	52
5.30.1.6	range_k_front	52
5.31	ObjectManager.cpp File Reference	52
5.32	ObjectManager.h File Reference	52
5.32.1	Enumeration Type Documentation	53
5.32.1.1	eObjectType	53
5.33	ObjectManager_init_bflbody.cpp File Reference	53
5.34	ObjectManager_init_ibmbody.cpp File Reference	53
5.35	ObjectManager_ops_ibm.cpp File Reference	53
5.36	ObjectManager_ops_ibmflex.cpp File Reference	53
5.36.1	Macro Definition Documentation	54
5.36.1.1	SWAP	54
5.36.1.2	SWAP	54
5.36.1.3	TINY	54
5.37	ObjectManager_ops_io.cpp File Reference	54
5.38	PCpts.h File Reference	54
5.39	stdafx.cpp File Reference	54
5.40	stdafx.h File Reference	54
5.40.1	Macro Definition Documentation	55
5.40.1.1	LUMA_FAILED	55
5.40.2	Function Documentation	55
5.40.2.1	is_nan(NumType n)	55
5.41	targetver.h File Reference	55

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Body< MarkerType >	9
Body< BFLMarker >	9
BFLBody	7
Body< IBMarker >	9
IBBody	19
MpiManager::buffer_struct	10
GridObj	11
GridUtils	16
MpiManager::layer_edges	24
Marker	25
BFLMarker	8
IBMarker	21
MarkerData	26
MpiManager	27
ObjectManager	30
PCpts	32
MpiManager::phdf5_struct	33
vector	
IVector< GenTyp >	23
IVector< double >	23
IVector< eType >	23

Chapter 2

Class Index

2.1 Class List

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

BFLBody	7
BFLMarker	8
Body< MarkerType >	9
MpiManager::buffer_struct	10
GridObj	11
GridUtils	16
IBBody	19
IBMarker	21
IVector< GenTyp >	23
MpiManager::layer_edges	24
Marker	25
MarkerData	26
MpiManager	27
ObjectManager	30
PCpts	32
MpiManager::phdf5_struct	33

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

BFLBody.cpp	35
BFLBody.h	35
BFLMarker.cpp	35
BFLMarker.h	35
Body.h	36
definitions.h	36
Gen_init_globalvars.cpp	43
globalvars.h	44
GridObj.cpp	44
GridObj.h	44
GridObj_init_grids.cpp	46
GridObj_ops_boundary.cpp	46
GridObj_ops_io.cpp	46
GridObj_ops_ibm.cpp	46
GridUtils.cpp	47
GridUtils.h	47
hdf5luma.h	48
IBBody.cpp	49
IBBody.h	49
IBMarker.cpp	49
IBMarker.h	49
IVector.h	50
main_ibm.cpp	50
Marker.h	50
Mpi_buffer_pack.cpp	50
Mpi_buffer_size_recv.cpp	51
Mpi_buffer_size_send.cpp	51
Mpi_buffer_unpk.cpp	51
MpiManager.cpp	51
MpiManager.h	51
ObjectManager.cpp	52
ObjectManager.h	52
ObjectManager_init_bflbody.cpp	53
ObjectManager_init_ibmbody.cpp	53
ObjectManager_ops_ibm.cpp	53

ObjectManager_ops_ibmflex.cpp	53
ObjectManager_ops_io.cpp	54
PCpts.h	54
stdafx.cpp	54
stdafx.h	54
targetver.h	55

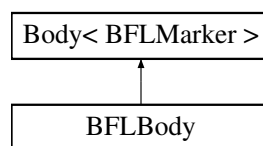
Chapter 4

Class Documentation

4.1 BFLBody Class Reference

```
#include <BFLBody.h>
```

Inheritance diagram for BFLBody:



Public Member Functions

- [BFLBody](#) (void)
- [~BFLBody](#) (void)
- [BFLBody](#) ([PCpts](#) *_PCpts, [GridObj](#) *g_hierarchy)

Protected Member Functions

- void [computeQ](#) (int i, int j, int k, int N_lim, int M_lim, int K_lim, [GridObj](#) *g)
- void [computeQ](#) (int i, int j, int N_lim, int M_lim, [GridObj](#) *g)

Protected Attributes

- `std::vector< std::vector< double > >` [Q](#)

Friends

- class [GridObj](#)

4.1.1 Constructor & Destructor Documentation

4.1.1.1 `BFLBody::BFLBody (void)`

4.1.1.2 `BFLBody::~~BFLBody (void)`

4.1.1.3 `BFLBody::BFLBody (PCpts * _PCpts, GridObj * g_hierarchy)`

4.1.2 Member Function Documentation

4.1.2.1 `void BFLBody::computeQ (int i, int j, int k, int N_lim, int M_lim, int K_lim, GridObj * g)` `[protected]`

4.1.2.2 `void BFLBody::computeQ (int i, int j, int N_lim, int M_lim, GridObj * g)` `[protected]`

4.1.3 Friends And Related Function Documentation

4.1.3.1 `friend class GridObj` `[friend]`

4.1.4 Member Data Documentation

4.1.4.1 `std::vector< std::vector<double> > BFLBody::Q` `[protected]`

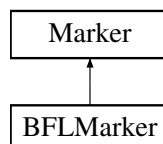
The documentation for this class was generated from the following files:

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

4.2 BFLMarker Class Reference

```
#include <BFLMarker.h>
```

Inheritance diagram for BFLMarker:



Public Member Functions

- [BFLMarker](#) (void)
- [~BFLMarker](#) (void)
- [BFLMarker](#) (double x, double y, double z)

Friends

- class [BFLBody](#)

Additional Inherited Members

4.2.1 Constructor & Destructor Documentation

4.2.1.1 [BFLMarker::BFLMarker](#) (void)

4.2.1.2 [BFLMarker::~~BFLMarker](#) (void)

4.2.1.3 [BFLMarker::BFLMarker](#) (double x, double y, double z)

4.2.2 Friends And Related Function Documentation

4.2.2.1 [friend class BFLBody](#) [[friend](#)]

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

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

4.3 Body< MarkerType > Class Template Reference

```
#include <Body.h>
```

Public Member Functions

- [Body](#) (void)
- [~Body](#) (void)
- [Body](#) (GridObj *g)

Protected Member Functions

- void [addMarker](#) (double x, double y, double z)
- [MarkerData](#) * [getMarkerData](#) (double x, double y, double z)
- void [markerAdder](#) (double x, double y, double z, int &curr_mark, std::vector< int > &counter)
- bool [isInVoxel](#) (double x, double y, double z, int curr_mark)
- bool [isVoxelMarkerVoxel](#) (double x, double y, double z)

Protected Attributes

- double [spacing](#)
- std::vector< MarkerType > [markers](#)
- bool [closed_surface](#)
- GridObj * [_Owner](#)

4.3.1 Constructor & Destructor Documentation

4.3.1.1 `template<typename MarkerType> Body< MarkerType >::Body (void)` `[inline]`

4.3.1.2 `template<typename MarkerType> Body< MarkerType >::~~Body (void)` `[inline]`

4.3.1.3 `template<typename MarkerType> Body< MarkerType >::Body (GridObj * g)` `[inline]`

4.3.2 Member Function Documentation

4.3.2.1 `template<typename MarkerType> void Body< MarkerType >::addMarker (double x, double y, double z)`
`[inline], [protected]`

4.3.2.2 `template<typename MarkerType> MarkerData* Body< MarkerType >::getMarkerData (double x, double y, double z)` `[inline], [protected]`

4.3.2.3 `template<typename MarkerType> bool Body< MarkerType >::isInVoxel (double x, double y, double z, int curr_mark)` `[inline], [protected]`

4.3.2.4 `template<typename MarkerType> bool Body< MarkerType >::isVoxelMarkerVoxel (double x, double y, double z)`
`[inline], [protected]`

4.3.2.5 `template<typename MarkerType> void Body< MarkerType >::markerAdder (double x, double y, double z, int & curr_mark, std::vector< int > & counter)` `[inline], [protected]`

4.3.3 Member Data Documentation

4.3.3.1 `template<typename MarkerType> GridObj* Body< MarkerType >::_Owner` `[protected]`

4.3.3.2 `template<typename MarkerType> bool Body< MarkerType >::closed_surface` `[protected]`

4.3.3.3 `template<typename MarkerType> std::vector<MarkerType> Body< MarkerType >::markers` `[protected]`

4.3.3.4 `template<typename MarkerType> double Body< MarkerType >::spacing` `[protected]`

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

- [Body.h](#)

4.4 MpiManager::buffer_struct Struct Reference

```
#include <MpiManager.h>
```

Public Attributes

- int [size](#) [[L_MPI_dir](#)]
- int [level](#)
- int [region](#)

4.4.1 Member Data Documentation

4.4.1.1 int `MpiManager::buffer_struct::level`

4.4.1.2 int `MpiManager::buffer_struct::region`

4.4.1.3 int `MpiManager::buffer_struct::size[L_MPI_dir]`

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

- [MpiManager.h](#)

4.5 GridObj Class Reference

```
#include <GridObj.h>
```

Public Member Functions

- [GridObj](#) ()
- [GridObj](#) (int [level](#))
- [GridObj](#) (int [RegionNumber](#), [GridObj](#) &pGrid)
- [GridObj](#) (int [level](#), std::vector< int > [local_size](#), std::vector< std::vector< int > > [GlobalLimsInd](#), std::vector< std::vector< double > > [GlobalLimsPos](#))
- [~GridObj](#) ()
- void [LBM_initVelocity](#) ()
- void [LBM_initRho](#) ()
- void [LBM_initGrid](#) ()
- void [LBM_initGrid](#) (std::vector< int > [local_size](#), std::vector< std::vector< int > > [GlobalLimsInd](#), std::vector< std::vector< double > > [GlobalLimsPos](#))
- void [LBM_initSubGrid](#) ([GridObj](#) &pGrid)
- void [LBM_initBoundLab](#) ()
- void [LBM_initSolidLab](#) ()
- void [LBM_initRefinedLab](#) ([GridObj](#) &pGrid)
- void [LBM_init_getInletProfile](#) ()
- void [LBM_multi](#) (bool [IBM_flag](#))
- void [LBM_collide](#) ()
- double [LBM_collide](#) (int i, int j, int k, int v, int [M_lim](#), int [K_lim](#))
- void [LBM_kbcCollide](#) (int i, int j, int k, int [M_lim](#), int [K_lim](#), [IVector](#)< double > &[f_new](#))
- void [LBM_stream](#) ()
- void [LBM_macro](#) ()
- void [LBM_macro](#) (int i, int j, int k)
- void [LBM_boundary](#) (int [bc_type_flag](#))
- void [LBM_forcegrid](#) (bool [reset_flag](#))

- void [bc_applyBounceBack](#) (int label, int i, int j, int k, int N_lim, int M_lim, int K_lim)
- void [bc_applySpecReflect](#) (int label, int i, int j, int k, int N_lim, int M_lim, int K_lim)
- void [bc_applyZouHe](#) (int label, int i, int j, int k, int M_lim, int K_lim)
- void [bc_applyRegularised](#) (int label, int i, int j, int k, int N_lim, int M_lim, int K_lim)
- void [bc_applyExtrapolation](#) (int label, int i, int j, int k, int M_lim, int K_lim)
- void [bc_applyBfl](#) (int i, int j, int k)
- void [bc_applyNrbc](#) (int i, int j, int k)
- void [bc_solidSiteReset](#) ()
- double [bc_getWallDensityForRBC](#) (std::vector< double > &ftmp, int normal, int i, int j, int k, int M_lim, int K_lim)
- void [LBM_explode](#) (int RegionNumber)
- void [LBM_coalesce](#) (int RegionNumber)
- void [LBM_addSubGrid](#) (int RegionNumber)
- void [io_textout](#) (std::string output_tag)
- void [io_restart](#) (bool IO_flag)
- void [io_probeOutput](#) ()
- void [io_lite](#) (double tval, std::string Tag)
- int [io_hdf5](#) (double tval)

Public Attributes

- std::vector< int > [XInd](#)
- std::vector< int > [YInd](#)
- std::vector< int > [ZInd](#)
- std::vector< double > [XPos](#)
- std::vector< double > [YPos](#)
- std::vector< double > [ZPos](#)
- [IVector](#)< [eType](#) > [LatTyp](#)
- int [level](#)
- double [dt](#)
- int [t](#)
- double [nu](#)
- double [omega](#)
- double [timeav_mpi_overhead](#)
- double [timeav_timestep](#)

Friends

- class [MpiManager](#)
- class [ObjectManager](#)
- class [GridUtils](#)

4.5.1 Constructor & Destructor Documentation

4.5.1.1 GridObj::GridObj (void)

4.5.1.2 GridObj::GridObj (int *level*)

4.5.1.3 GridObj::GridObj (int *RegionNumber*, GridObj & *pGrid*)

4.5.1.4 GridObj::GridObj (int *level*, std::vector< int > *local_size*, std::vector< std::vector< int > > *GlobalLimsInd*, std::vector< std::vector< double > > *GlobalLimsPos*)

4.5.1.5 GridObj::~GridObj (void)

4.5.2 Member Function Documentation

4.5.2.1 void GridObj::bc_applyBfl (int *i*, int *j*, int *k*)

Apply BC in pairs – BC 1

Apply BC in pairs – BC 2

4.5.2.2 void GridObj::bc_applyBounceBack (int *label*, int *i*, int *j*, int *k*, int *N_lim*, int *M_lim*, int *K_lim*)

4.5.2.3 void GridObj::bc_applyExtrapolation (int *label*, int *i*, int *j*, int *k*, int *M_lim*, int *K_lim*)

4.5.2.4 void GridObj::bc_applyNrbc (int *i*, int *j*, int *k*)

4.5.2.5 void GridObj::bc_applyRegularised (int *label*, int *i*, int *j*, int *k*, int *N_lim*, int *M_lim*, int *K_lim*)

To allow a generalised application of the regularised BC we need to implement the following:

1. Check normal directions to find orientation of normal wall;
2. Store the unknown and known directions in two vectors
3. Apply the boundary condition. Note: for the corners, the unknown distributions are too numerous to evaluate the density so must use extrapolation for the 4 corners (2D) or 8 corners (3D) from the bulk flow to find the density. Hence, we handle this case after the normal cases.

If it reaches here and has gone through all the directions then the wall orientation cannot be found based on the fact that it cannot find an adjacent site within the centre of the domain on which to base the BC. In this case it must be a buffered layer of inlet sites (the "second row" of sites you get when embedding a sub-grid in the inlet) so just set it to default values as it doesn't affect the domain anyway.

- 4.5.2.6 void GridObj::bc_applySpecReflect (int *label*, int *i*, int *j*, int *k*, int *N_lim*, int *M_lim*, int *K_lim*)
- 4.5.2.7 void GridObj::bc_applyZouHe (int *label*, int *i*, int *j*, int *k*, int *M_lim*, int *K_lim*)
- 4.5.2.8 double GridObj::bc_getWallDensityForRBC (std::vector< double > & *ftmp*, int *normal*, int *i*, int *j*, int *k*, int *M_lim*, int *K_lim*)
- 4.5.2.9 void GridObj::bc_solidSiteReset ()
- 4.5.2.10 int GridObj::io_hdf5 (double *tval*)
- 4.5.2.11 void GridObj::io_lite (double *tval*, std::string *Tag*)
- 4.5.2.12 void GridObj::io_probeOutput ()
- 4.5.2.13 void GridObj::io_restart (bool *IO_flag*)
- 4.5.2.14 void GridObj::io_textout (std::string *output_tag*)
- 4.5.2.15 void GridObj::LBM_addSubGrid (int *RegionNumber*)
- 4.5.2.16 void GridObj::LBM_boundary (int *bc_type_flag*)
- 4.5.2.17 void GridObj::LBM_coalesce (int *RegionNumber*)
- 4.5.2.18 void GridObj::LBM_collide ()
- 4.5.2.19 double GridObj::LBM_collide (int *i*, int *j*, int *k*, int *v*, int *M_lim*, int *K_lim*)
- 4.5.2.20 void GridObj::LBM_explode (int *RegionNumber*)
- 4.5.2.21 void GridObj::LBM_forcegrid (bool *reset_flag*)
- 4.5.2.22 void GridObj::LBM_init_getInletProfile ()
- 4.5.2.23 void GridObj::LBM_initBoundLab ()
- 4.5.2.24 void GridObj::LBM_initGrid ()
- 4.5.2.25 void GridObj::LBM_initGrid (std::vector< int > *local_size*, std::vector< std::vector< int > > *GlobalLimsInd*, std::vector< std::vector< double > > *GlobalLimsPos*)
- 4.5.2.26 void GridObj::LBM_initRefinedLab (GridObj & *pGrid*)
- 4.5.2.27 void GridObj::LBM_initRho ()

- 4.5.2.28 void GridObj::LBM_initSolidLab ()
- 4.5.2.29 void GridObj::LBM_initSubGrid (GridObj & pGrid)
- 4.5.2.30 void GridObj::LBM_initVelocity ()
- 4.5.2.31 void GridObj::LBM_kbcCollide (int *i*, int *j*, int *k*, int *M_lim*, int *K_lim*, IVector< double > & *f_new*)
- 4.5.2.32 void GridObj::LBM_macro ()
- 4.5.2.33 void GridObj::LBM_macro (int *i*, int *j*, int *k*)
- 4.5.2.34 void GridObj::LBM_multi (bool *IBM_flag*)
- 4.5.2.35 void GridObj::LBM_stream ()

4.5.3 Friends And Related Function Documentation

- 4.5.3.1 friend class GridUtils [friend]
- 4.5.3.2 friend class MpiManager [friend]
- 4.5.3.3 friend class ObjectManager [friend]

4.5.4 Member Data Documentation

- 4.5.4.1 double GridObj::dt
- 4.5.4.2 IVector<eType> GridObj::LatTyp
- 4.5.4.3 int GridObj::level
- 4.5.4.4 double GridObj::nu
- 4.5.4.5 double GridObj::omega
- 4.5.4.6 int GridObj::t
- 4.5.4.7 double GridObj::timeav_mpi_overhead
- 4.5.4.8 double GridObj::timeav_timestep
- 4.5.4.9 std::vector<int> GridObj::XInd
- 4.5.4.10 std::vector<double> GridObj::XPos
- 4.5.4.11 std::vector<int> GridObj::YInd
- 4.5.4.12 std::vector<double> GridObj::YPos
- 4.5.4.13 std::vector<int> GridObj::ZInd
- 4.5.4.14 std::vector<double> GridObj::ZPos

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

- [GridObj.h](#)
- [GridObj.cpp](#)
- [GridObj_init_grids.cpp](#)
- [GridObj_ops_boundary.cpp](#)
- [GridObj_ops_io.cpp](#)
- [GridObj_ops_lbm.cpp](#)

4.6 GridUtils Class Reference

```
#include <GridUtils.h>
```

Static Public Member Functions

- static int [createOutputDirectory](#) (std::string [path_str](#))
- static std::vector< int > [onespace](#) (int min, int max)
- static std::vector< double > [linspace](#) (double min, double max, int n)
- static double [vecnorm](#) (double vec[])
- static double [vecnorm](#) (double val1, double val2)
- static double [vecnorm](#) (double val1, double val2, double val3)
- static double [vecnorm](#) (std::vector< double > vec)
- static std::vector< int > [getFineIndices](#) (int coarse_i, int x_start, int coarse_j, int y_start, int coarse_k, int z_start)
- static std::vector< int > [getCoarseIndices](#) (int fine_i, int x_start, int fine_j, int y_start, int fine_k, int z_start)
- static double [indexToPosition](#) (int index, double dx)
- static double [dotprod](#) (std::vector< double > vec1, std::vector< double > vec2)
- static std::vector< double > [subtract](#) (std::vector< double > a, std::vector< double > b)
- static std::vector< double > [add](#) (std::vector< double > a, std::vector< double > b)
- static std::vector< double > [vecmultiply](#) (double scalar, std::vector< double > vec)
- static std::vector< double > [crossprod](#) (std::vector< double > vec1, std::vector< double > vec2)
- static std::vector< double > [matrix_multiply](#) (const std::vector< std::vector< double > > &A, const std::vector< double > &x)
- static int [getOpposite](#) (int direction)
- static void [getGrid](#) (GridObj *&Grids, int level, int region, GridObj *&ptr)
- static bool [isOverlapPeriodic](#) (int i, int j, int k, const GridObj &pGrid)
- static bool [isOnThisRank](#) (int gi, int gj, int gk, const GridObj &pGrid)
- static bool [isOnThisRank](#) (int gl, enum [eCartesianDirection](#) xyz, const GridObj &pGrid)
- static bool [hasThisSubGrid](#) (const GridObj &pGrid, int RegNum)
- static bool [isOnSenderLayer](#) (double pos_x, double pos_y, double pos_z)
- static bool [isOnRecvLayer](#) (double pos_x, double pos_y, double pos_z)
- static bool [isOnSenderLayer](#) (double site_position, enum [eCartesianDirection](#) xyz, enum [eMinMax](#) minmax)
- static bool [isOnRecvLayer](#) (double site_position, enum [eCartesianDirection](#) xyz, enum [eMinMax](#) minmax)
- static bool [isOffGrid](#) (int i, int j, int k, int N_lim, int M_lim, int K_lim, GridObj &g)
- template<typename NumType >
static NumType [vecnorm](#) (NumType a1, NumType a2, NumType a3)
- template<typename NumType >
static NumType [vecnorm](#) (NumType a1, NumType a2)
- template<typename NumType >
static NumType [upToZero](#) (NumType x)
- template<typename NumType >
static NumType [downToLimit](#) (NumType x, NumType limit)
- template<typename NumType >
static NumType [factorial](#) (NumType n)

- `template<typename NumType >`
`static void stridedCopy (NumType *dest, NumType *src, size_t block, size_t offset, size_t stride, size_t count, size_t buf_offset=0)`
- `template<typename NumType >`
`static void global_to_local (int i, int j, int k, GridObj *g, std::vector< NumType > &locals)`
- `template<typename NumType >`
`static void local_to_global (int i, int j, int k, GridObj *g, std::vector< NumType > &globals)`

Static Public Attributes

- `static std::ofstream * logfile`
- `static std::string path_str`
- `static const int dir_reflect [L_dims *2][L_nVels]`

4.6.1 Detailed Description

[GridUtils](#) Class is a utility class to hold all the general methods used by the [GridObj](#) and others. Everything about this is static as no need to instantiate it for every grid on a process.

4.6.2 Member Function Documentation

4.6.2.1 `std::vector< double > GridUtils::add (std::vector< double > a, std::vector< double > b)` `[static]`

4.6.2.2 `int GridUtils::createOutputDirectory (std::string path_str)` `[static]`

4.6.2.3 `std::vector< double > GridUtils::crossprod (std::vector< double > vec1, std::vector< double > vec2)`
`[static]`

4.6.2.4 `double GridUtils::dotprod (std::vector< double > vec1, std::vector< double > vec2)` `[static]`

4.6.2.5 `template<typename NumType > static NumType GridUtils::downToLimit (NumType x, NumType limit)`
`[inline], [static]`

4.6.2.6 `template<typename NumType > static NumType GridUtils::factorial (NumType n)` `[inline], [static]`

4.6.2.7 `std::vector< int > GridUtils::getCoarseIndices (int fine_i, int x_start, int fine_j, int y_start, int fine_k, int z_start)`
`[static]`

4.6.2.8 `std::vector< int > GridUtils::getFineIndices (int coarse_i, int x_start, int coarse_j, int y_start, int coarse_k, int z_start)` `[static]`

4.6.2.9 `void GridUtils::getGrid (GridObj *& Grids, int level, int region, GridObj *& ptr)` `[static]`

4.6.2.10 `int GridUtils::getOpposite (int direction)` `[static]`

4.6.2.11 `template<typename NumType > static void GridUtils::global_to_local (int i, int j, int k, GridObj * g, std::vector< NumType > & locals)` `[inline], [static]`

- 4.6.2.12 `bool GridUtils::hasThisSubGrid (const GridObj & pGrid, int RegNum) [static]`
- 4.6.2.13 `double GridUtils::indexToPosition (int index, double dx) [static]`
- 4.6.2.14 `bool GridUtils::isOffGrid (int i, int j, int k, int N_lim, int M_lim, int K_lim, GridObj & g) [static]`
- 4.6.2.15 `bool GridUtils::isOnRecvLayer (double pos_x, double pos_y, double pos_z) [static]`
- 4.6.2.16 `bool GridUtils::isOnRecvLayer (double site_position, enum eCartesianDirection xyz, enum eMinMax minmax) [static]`
- 4.6.2.17 `bool GridUtils::isOnSenderLayer (double pos_x, double pos_y, double pos_z) [static]`
- 4.6.2.18 `bool GridUtils::isOnSenderLayer (double site_position, enum eCartesianDirection xyz, enum eMinMax minmax) [static]`
- 4.6.2.19 `bool GridUtils::isOnThisRank (int gi, int gj, int gk, const GridObj & pGrid) [static]`
- 4.6.2.20 `bool GridUtils::isOnThisRank (int gl, enum eCartesianDirection xyz, const GridObj & pGrid) [static]`
- 4.6.2.21 `bool GridUtils::isOverlapPeriodic (int i, int j, int k, const GridObj & pGrid) [static]`
- 4.6.2.22 `std::vector< double > GridUtils::linspace (double min, double max, int n) [static]`
- 4.6.2.23 `template<typename NumType > static void GridUtils::local_to_global (int i, int j, int k, GridObj * g, std::vector< NumType > & globals) [inline],[static]`
- 4.6.2.24 `std::vector< double > GridUtils::matrix_multiply (const std::vector< std::vector< double > > & A, const std::vector< double > & x) [static]`
- 4.6.2.25 `std::vector< int > GridUtils::onespace (int min, int max) [static]`
- 4.6.2.26 `template<typename NumType > static void GridUtils::stridedCopy (NumType * dest, NumType * src, size_t block, size_t offset, size_t stride, size_t count, size_t buf_offset=0) [inline],[static]`
- 4.6.2.27 `std::vector< double > GridUtils::subtract (std::vector< double > a, std::vector< double > b) [static]`
- 4.6.2.28 `template<typename NumType > static NumType GridUtils::upToZero (NumType x) [inline],[static]`
- 4.6.2.29 `std::vector< double > GridUtils::vecmultiply (double scalar, std::vector< double > vec) [static]`
- 4.6.2.30 `double GridUtils::vecnorm (double vec[]) [static]`
- 4.6.2.31 `double GridUtils::vecnorm (double val1, double val2) [static]`
- 4.6.2.32 `double GridUtils::vecnorm (double val1, double val2, double val3) [static]`

4.6.2.33 `double GridUtils::vecnorm (std::vector< double > vec) [static]`

4.6.2.34 `template<typename NumType > static NumType GridUtils::vecnorm (NumType a1, NumType a2, NumType a3) [inline],[static]`

4.6.2.35 `template<typename NumType > static NumType GridUtils::vecnorm (NumType a1, NumType a2) [inline],[static]`

4.6.3 Member Data Documentation

4.6.3.1 `const int GridUtils::dir_reflect [static]`

Initial value:

```
=
{
    {1, 0, 2, 3, 4, 5, 9, 8, 7, 6, 10, 11, 12, 13, 16, 17, 14, 15, 18},
    {1, 0, 2, 3, 4, 5, 9, 8, 7, 6, 10, 11, 12, 13, 16, 17, 14, 15, 18},
    {0, 1, 3, 2, 4, 5, 8, 9, 6, 7, 13, 12, 11, 10, 14, 15, 16, 17, 18},
    {0, 1, 3, 2, 4, 5, 8, 9, 6, 7, 13, 12, 11, 10, 14, 15, 16, 17, 18},
    {0, 1, 2, 3, 5, 4, 6, 7, 8, 9, 12, 13, 10, 11, 17, 16, 15, 14, 18},
    {0, 1, 2, 3, 5, 4, 6, 7, 8, 9, 12, 13, 10, 11, 17, 16, 15, 14, 18}
}
```

4.6.3.2 `std::ofstream * GridUtils::logfile [static]`

4.6.3.3 `std::string GridUtils::path_str [static]`

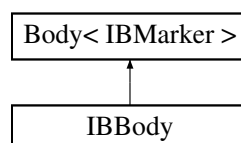
The documentation for this class was generated from the following files:

- [GridUtils.h](#)
- [GridObj.cpp](#)
- [GridUtils.cpp](#)
- [main_ibm.cpp](#)

4.7 IBody Class Reference

```
#include <IBody.h>
```

Inheritance diagram for IBody:



Public Member Functions

- [IBBody](#) (void)
- [~IBBody](#) (void)
- [IBBody](#) ([GridObj](#) *g)
- void [addMarker](#) (double x, double y, double z, bool [flex_rigid](#))
- void [makeBody](#) (double radius, std::vector< double > centre, bool [flex_rigid](#), bool moving, int group)
- void [makeBody](#) (std::vector< double > width_length_depth, std::vector< double > angles, std::vector< double > centre, bool [flex_rigid](#), bool deform, int group)
- void [makeBody](#) (int numbermarkers, std::vector< double > start_point, double fil_length, std::vector< double > angles, std::vector< int > [BCs](#), bool [flex_rigid](#), bool deform, int group)
- double [makeBody](#) (std::vector< double > width_length, double angle, std::vector< double > centre, bool [flex_rigid](#), bool deform, int group, bool plate)
- void [makeBody](#) ([PCpts](#) *_PCpts)

Protected Attributes

- bool [flex_rigid](#)
- bool [deformable](#)
- int [groupID](#)
- double [delta_rho](#)
- double [flexural_rigidity](#)
- std::vector< double > [tension](#)
- std::vector< int > [BCs](#)

Friends

- class [ObjectManager](#)

Additional Inherited Members

4.7.1 Detailed Description

Represents an IB body

4.7.2 Constructor & Destructor Documentation

4.7.2.1 [IBBody::IBBody \(void \)](#)

4.7.2.2 [IBBody::~~IBBody \(void \)](#)

4.7.2.3 [IBBody::IBBody \(\[GridObj\]\(#\) * g \)](#)

4.7.3 Member Function Documentation

4.7.3.1 [void IBBody::addMarker \(double x, double y, double z, bool \[flex_rigid\]\(#\) \)](#)

4.7.3.2 void IBody::makeBody (double *radius*, std::vector< double > *centre*, bool *flex_rigid*, bool *moving*, int *group*)

4.7.3.3 void IBody::makeBody (std::vector< double > *width_length_depth*, std::vector< double > *angles*, std::vector< double > *centre*, bool *flex_rigid*, bool *deform*, int *group*)

4.7.3.4 void IBody::makeBody (int *numbermarkers*, std::vector< double > *start_point*, double *fil_length*, std::vector< double > *angles*, std::vector< int > *BCs*, bool *flex_rigid*, bool *deform*, int *group*)

4.7.3.5 double IBody::makeBody (std::vector< double > *width_length*, double *angle*, std::vector< double > *centre*, bool *flex_rigid*, bool *deform*, int *group*, bool *plate*)

4.7.3.6 void IBody::makeBody (PCpts * *_PCpts*)

4.7.4 Friends And Related Function Documentation

4.7.4.1 friend class ObjectManager [friend]

4.7.5 Member Data Documentation

4.7.5.1 std::vector<int> IBody::BCs [protected]

4.7.5.2 bool IBody::deformable [protected]

4.7.5.3 double IBody::delta_rho [protected]

4.7.5.4 bool IBody::flex_rigid [protected]

4.7.5.5 double IBody::flexural_rigidity [protected]

4.7.5.6 int IBody::groupID [protected]

4.7.5.7 std::vector<double> IBody::tension [protected]

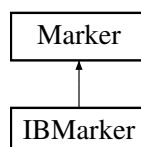
The documentation for this class was generated from the following files:

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

4.8 IBody Class Reference

```
#include <IBody.h>
```

Inheritance diagram for IBody:



Public Member Functions

- [IBMarker](#) (void)
- [~IBMarker](#) (void)
- [IBMarker](#) (double xPos, double yPos, double zPos, bool [flex_rigid](#)=false)

Protected Attributes

- std::vector< double > [fluid_vel](#)
- std::vector< double > [desired_vel](#)
- std::vector< double > [force_xyz](#)
- std::vector< double > [position_old](#)
- std::vector< double > [deltaval](#)
- bool [flex_rigid](#)
- double [epsilon](#)
- double [local_area](#)
- double [dilation](#)

Friends

- class [ObjectManager](#)
- class [IBBody](#)

Additional Inherited Members

4.8.1 Constructor & Destructor Documentation

4.8.1.1 `IBMarker::IBMarker (void)` `[inline]`

4.8.1.2 `IBMarker::~~IBMarker (void)` `[inline]`

4.8.1.3 `IBMarker::IBMarker (double xPos, double yPos, double zPos, bool flex_rigid = false)`

4.8.2 Friends And Related Function Documentation

4.8.2.1 `friend class IBBody` `[friend]`

4.8.2.2 `friend class ObjectManager` `[friend]`

4.8.3 Member Data Documentation

4.8.3.1 `std::vector<double> IBMarker::deltaval` `[protected]`

4.8.3.2 `std::vector<double> IBMarker::desired_vel` `[protected]`

4.8.3.3 `double IBMarker::dilation` `[protected]`

4.8.3.4 `double IBMarker::epsilon` `[protected]`

4.8.3.5 `bool IBMarker::flex_rigid` `[protected]`

4.8.3.6 `std::vector<double> IBMarker::fluid_vel` `[protected]`

4.8.3.7 `std::vector<double> IBMarker::force_xyz` `[protected]`

4.8.3.8 `double IBMarker::local_area` `[protected]`

4.8.3.9 `std::vector<double> IBMarker::position_old` `[protected]`

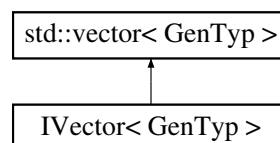
The documentation for this class was generated from the following files:

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

4.9 IVector< GenTyp > Class Template Reference

```
#include <IVector.h>
```

Inheritance diagram for IVector< GenTyp >:



Public Member Functions

- [IVector](#) ()
- [~IVector](#) ()
- [IVector](#) (size_t size, GenTyp val)
- GenTyp & [operator\(\)](#) (size_t i, size_t j, size_t k, size_t v, size_t j_max, size_t k_max, size_t v_max)
- GenTyp & [operator\(\)](#) (size_t i, size_t j, size_t k, size_t j_max, size_t k_max)
- GenTyp & [operator\(\)](#) (size_t i, size_t j, size_t j_max)

4.9.1 Constructor & Destructor Documentation

4.9.1.1 `template<typename GenTyp> IVector< GenTyp >::IVector () [inline]`

4.9.1.2 `template<typename GenTyp> IVector< GenTyp >::~~IVector () [inline]`

4.9.1.3 `template<typename GenTyp> IVector< GenTyp >::IVector (size_t size, GenTyp val) [inline]`

4.9.2 Member Function Documentation

4.9.2.1 `template<typename GenTyp> GenTyp& IVector< GenTyp >::operator() (size_t i, size_t j, size_t k, size_t v, size_t j_max, size_t k_max, size_t v_max) [inline]`

4.9.2.2 `template<typename GenTyp> GenTyp& IVector< GenTyp >::operator() (size_t i, size_t j, size_t k, size_t j_max, size_t k_max) [inline]`

4.9.2.3 `template<typename GenTyp> GenTyp& IVector< GenTyp >::operator() (size_t i, size_t j, size_t j_max) [inline]`

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

- [IVector.h](#)

4.10 MpiManager::layer_edges Struct Reference

```
#include <MpiManager.h>
```

Public Attributes

- double [X](#) [4]
- double [Y](#) [4]
- double [Z](#) [4]

4.10.1 Member Data Documentation

4.10.1.1 `double MpiManager::layer_edges::X[4]`

4.10.1.2 `double MpiManager::layer_edges::Y[4]`

4.10.1.3 `double MpiManager::layer_edges::Z[4]`

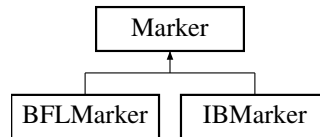
The documentation for this struct was generated from the following file:

- [MpiManager.h](#)

4.11 Marker Class Reference

```
#include <Marker.h>
```

Inheritance diagram for Marker:



Public Member Functions

- [Marker](#) (void)
- [~Marker](#) (void)
- [Marker](#) (double x, double y, double z)

Public Attributes

- `std::vector< double >` [position](#)
- `std::vector< int >` [supp_i](#)
- `std::vector< int >` [supp_j](#)
- `std::vector< int >` [supp_k](#)
- `std::vector< int >` [support_rank](#)

4.11.1 Detailed Description

Represents a generic marker for a body

4.11.2 Constructor & Destructor Documentation

4.11.2.1 `Marker::Marker (void)` [\[inline\]](#)

4.11.2.2 `Marker::~~Marker (void)` [\[inline\]](#)

4.11.2.3 `Marker::Marker (double x, double y, double z)` [\[inline\]](#)

4.11.3 Member Data Documentation

4.11.3.1 `std::vector<double>` `Marker::position`

4.11.3.2 `std::vector<int>` `Marker::supp_i`

4.11.3.3 `std::vector<int>` `Marker::supp_j`

4.11.3.4 `std::vector<int>` `Marker::supp_k`

4.11.3.5 `std::vector<int>` `Marker::support_rank`

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

- [Marker.h](#)

4.12 MarkerData Class Reference

```
#include <Body.h>
```

Public Member Functions

- [MarkerData](#) (int *i*, int *j*, int *k*, double *x*, double *y*, double *z*, int *ID*)
- [MarkerData](#) (void)
- [~MarkerData](#) (void)

Public Attributes

- int *i*
- int *j*
- int *k*
- int *ID*
- double *x*
- double *y*
- double *z*

4.12.1 Constructor & Destructor Documentation

4.12.1.1 `MarkerData::MarkerData (int i, int j, int k, double x, double y, double z, int ID)` `[inline]`

4.12.1.2 `MarkerData::MarkerData (void)` `[inline]`

4.12.1.3 `MarkerData::~~MarkerData (void)` `[inline]`

4.12.2 Member Data Documentation

4.12.2.1 `int MarkerData::i`

4.12.2.2 `int MarkerData::ID`

4.12.2.3 `int MarkerData::j`

4.12.2.4 `int MarkerData::k`

4.12.2.5 `double MarkerData::x`

4.12.2.6 `double MarkerData::y`

4.12.2.7 `double MarkerData::z`

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

- [Body.h](#)

4.13 MpiManager Class Reference

```
#include <MpiManager.h>
```

Classes

- struct [buffer_struct](#)
- struct [layer_edges](#)
- struct [phdf5_struct](#)

Public Member Functions

- void [mpi_init](#) ()
- void [mpi_gridbuild](#) ()
- int [mpi_buildCommunicators](#) ()
- void [mpi_buffer_pack](#) (int dir, [GridObj](#) *g)
- void [mpi_buffer_unpack](#) (int dir, [GridObj](#) *g)
- void [mpi_buffer_size](#) ()
- void [mpi_buffer_size_send](#) ([GridObj](#) *&g)
- void [mpi_buffer_size_recv](#) ([GridObj](#) *&g)
- void [mpi_writeout_buf](#) (std::string filename, int dir)
- void [mpi_communicate](#) (int level, int regnum)
- int [mpi_getOpposite](#) (int direction)

Static Public Member Functions

- static [MpiManager](#) * [getInstance](#) ()
- static void [destroyInstance](#) ()

Public Attributes

- MPI_Comm [world_comm](#)
- int [MPI_dims](#) [[L_dims](#)]
- int [neighbour_rank](#) [[L_MPI_dir](#)]
- int [neighbour_coords](#) [[L_dims](#)][[L_MPI_dir](#)]
- MPI_Comm [subGrid_comm](#) [[L_NumLev](#) *[L_NumReg](#)]
- std::vector< [phdf5_struct](#) > [p_data](#)
- int [global_dims](#) [3]
- std::vector< int > [local_size](#)
- std::vector< std::vector< int > > [global_edge_ind](#)
- std::vector< std::vector< double > > [global_edge_pos](#)
- struct [MpiManager::layer_edges](#) [sender_layer_pos](#)
- struct [MpiManager::layer_edges](#) [recv_layer_pos](#)
- std::vector< std::vector< double > > [f_buffer_send](#)
- std::vector< std::vector< double > > [f_buffer_recv](#)
- MPI_Status [recv_stat](#)
- MPI_Request [send_requests](#) [[L_MPI_dir](#)]
- MPI_Status [send_stat](#) [[L_MPI_dir](#)]
- std::vector< [buffer_struct](#) > [buffer_send_info](#)
- std::vector< [buffer_struct](#) > [buffer_recv_info](#)

Static Public Attributes

- static const int [MPI_cartlab](#) [3][26]
- static int [my_rank](#)
- static int [num_ranks](#)
- static int [MPI_coords](#) [[L_dims](#)]
- static [GridObj](#) * [Grids](#)
- static std::ofstream * [logout](#)

4.13.1 Member Function Documentation

- 4.13.1.1 void [MpiManager::destroyInstance](#) () [[static](#)]
- 4.13.1.2 [MpiManager](#) * [MpiManager::getInstance](#) () [[static](#)]
- 4.13.1.3 void [MpiManager::mpi_buffer_pack](#) (int *dir*, [GridObj](#) * *g*)
- 4.13.1.4 void [MpiManager::mpi_buffer_size](#) ()
- 4.13.1.5 void [MpiManager::mpi_buffer_size_recv](#) ([GridObj](#) *& *g*)
- 4.13.1.6 void [MpiManager::mpi_buffer_size_send](#) ([GridObj](#) *& *g*)
- 4.13.1.7 void [MpiManager::mpi_buffer_unpack](#) (int *dir*, [GridObj](#) * *g*)
- 4.13.1.8 int [MpiManager::mpi_buildCommunicators](#) ()
- 4.13.1.9 void [MpiManager::mpi_communicate](#) (int *level*, int *regnum*)
- 4.13.1.10 int [MpiManager::mpi_getOpposite](#) (int *direction*)
- 4.13.1.11 void [MpiManager::mpi_gridbuild](#) ()
- 4.13.1.12 void [MpiManager::mpi_init](#) ()
- 4.13.1.13 void [MpiManager::mpi_writeout_buf](#) (std::string *filename*, int *dir*)

4.13.2 Member Data Documentation

- 4.13.2.1 std::vector<[buffer_struct](#)> [MpiManager::buffer_recv_info](#)
- 4.13.2.2 std::vector<[buffer_struct](#)> [MpiManager::buffer_send_info](#)
- 4.13.2.3 std::vector< std::vector<double> > [MpiManager::f_buffer_recv](#)
- 4.13.2.4 std::vector< std::vector<double> > [MpiManager::f_buffer_send](#)

4.13.2.5 `int MpiManager::global_dims[3]`

4.13.2.6 `std::vector< std::vector<int> > MpiManager::global_edge_ind`

4.13.2.7 `std::vector< std::vector<double> > MpiManager::global_edge_pos`

4.13.2.8 `GridObj * MpiManager::Grids` `[static]`

4.13.2.9 `std::vector<int> MpiManager::local_size`

4.13.2.10 `std::ofstream * MpiManager::logout` `[static]`

4.13.2.11 `const int MpiManager::MPI_cartlab` `[static]`

Initial value:

```
=
{
    {1, -1, 1, -1, 0, 0, -1, 1, 0, 0, 1, -1, 1, -1, 0, 0, -1, 1, -1, 1, -1, 1, 0,
    0, 1, -1},
    {0, 0, 1, -1, 1, -1, 1, -1, 0, 0, 0, 0, 1, -1, 1, -1, 1, -1, 0, 0, -1, 1, -1,
    1, -1, 1},
    {0, 0, 0, 0, 0, 0, 0, 0, 1, -1, 1, -1, 1, -1, 1, -1, 1, -1, 1, -1, 1, -1,
    -1, 1, -1}
}
```

4.13.2.12 `int MpiManager::MPI_coords` `[static]`

4.13.2.13 `int MpiManager::MPI_dims[L_dims]`

4.13.2.14 `int MpiManager::my_rank` `[static]`

4.13.2.15 `int MpiManager::neighbour_coords[L_dims][L_MPI_dir]`

4.13.2.16 `int MpiManager::neighbour_rank[L_MPI_dir]`

4.13.2.17 `int MpiManager::num_ranks` `[static]`

4.13.2.18 `std::vector<phdf5_struct> MpiManager::p_data`

4.13.2.19 `struct MpiManager::layer_edges MpiManager::recv_layer_pos`

4.13.2.20 `MPI_Status MpiManager::recv_stat`

4.13.2.21 `MPI_Request MpiManager::send_requests[L_MPI_dir]`

4.13.2.22 `MPI_Status MpiManager::send_stat[L_MPI_dir]`

4.13.2.23 `struct MpiManager::layer_edges MpiManager::sender_layer_pos`

4.13.2.24 `MPI_Comm MpiManager::subGrid_comm[L_NumLev * L_NumReg]`

4.13.2.25 `MPI_Comm MpiManager::world_comm`

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

- [MpiManager.h](#)
- [GridObj.cpp](#)
- [main_ibm.cpp](#)
- [Mpi_buffer_pack.cpp](#)
- [Mpi_buffer_size_recv.cpp](#)
- [Mpi_buffer_size_send.cpp](#)
- [Mpi_buffer_unpk.cpp](#)
- [MpiManager.cpp](#)

4.14 ObjectManager Class Reference

```
#include <ObjectManager.h>
```

Public Member Functions

- void [ibm_apply](#) ([GridObj](#) &g)
- void [ibm_build_body](#) (int body_type)
- void [ibm_build_body](#) ([PCpts](#) *_PCpts, [GridObj](#) *owner)
- void [ibm_initialise](#) ([GridObj](#) &g)
- double [ibm_deltakernel](#) (double rad, double dilation)
- void [ibm_interpol](#) (int ib, [GridObj](#) &g)
- void [ibm_spread](#) (int ib, [GridObj](#) &g)
- void [ibm_findsupport](#) (int ib, int m, [GridObj](#) &g)
- void [ibm_computeforce](#) (int ib, [GridObj](#) &g)
- double [ibm_findepsilon](#) (int ib, [GridObj](#) &g)
- void [ibm_move_bodies](#) ([GridObj](#) &g)
- double [ibm_bicgstab](#) (std::vector< std::vector< double > > &Amatrix, std::vector< double > &bVector, std::vector< double > &epsilon, double tolerance, int maxiterations)
- void [ibm_jacowire](#) (int ib, [GridObj](#) &g)
- void [ibm_position_update](#) (int ib, [GridObj](#) &g)
- void [ibm_position_update_grp](#) (int group, [GridObj](#) &g)
- void [ibm_banbks](#) (double **a, long n, int m1, int m2, double **al, unsigned long indx[], double b[])
- void [ibm_bandec](#) (double **a, long n, int m1, int m2, double **al, unsigned long indx[], double *d)
- void [bfl_build_body](#) (int body_type)
- void [bfl_build_body](#) ([PCpts](#) *_PCpts)
- std::vector< int > [getVoxInd](#) (double x, double y, double z)
- int [getVoxInd](#) (double p)
- void [io_vtk_IBwriter](#) (double tval)
- void [io_write_body_pos](#) (int timestep)
- void [io_write_lift_drag](#) (int timestep)
- void [io_restart](#) (bool IO_flag, int level)
- void [io_readInCloud](#) ([PCpts](#) *_PCpts, [eObjectType](#) objtype)

Static Public Member Functions

- static [ObjectManager](#) * [getInstance](#) ()
- static void [destroyInstance](#) ()
- static [ObjectManager](#) * [getInstance](#) ([GridObj](#) *g)

Friends

- class [GridObj](#)

4.14.1 Member Function Documentation

- 4.14.1.1 void ObjectManager::bfl_build_body (int *body_type*)
- 4.14.1.2 void ObjectManager::bfl_build_body (PCpts * *_PCpts*)
- 4.14.1.3 void ObjectManager::destroyInstance () [static]
- 4.14.1.4 ObjectManager * ObjectManager::getInstance () [static]
- 4.14.1.5 ObjectManager * ObjectManager::getInstance (GridObj * *g*) [static]
- 4.14.1.6 std::vector< int > ObjectManager::getVoxInd (double *x*, double *y*, double *z*)
- 4.14.1.7 int ObjectManager::getVoxInd (double *p*)
- 4.14.1.8 void ObjectManager::ibm_apply (GridObj & *g*)
- 4.14.1.9 void ObjectManager::ibm_banbks (double ** *a*, long *n*, int *m1*, int *m2*, double ** *a1*, unsigned long *indx[]*, double *b[]*)
- 4.14.1.10 void ObjectManager::ibm_bandec (double ** *a*, long *n*, int *m1*, int *m2*, double ** *a1*, unsigned long *indx[]*, double * *d*)
- 4.14.1.11 double ObjectManager::ibm_bicgstab (std::vector< std::vector< double > > & *Amatrix*, std::vector< double > & *bVector*, std::vector< double > & *epsilon*, double *tolerance*, int *maxiterations*)
- 4.14.1.12 void ObjectManager::ibm_build_body (int *body_type*)
- 4.14.1.13 void ObjectManager::ibm_build_body (PCpts * *_PCpts*, GridObj * *owner*)
- 4.14.1.14 void ObjectManager::ibm_computeforce (int *ib*, GridObj & *g*)
- 4.14.1.15 double ObjectManager::ibm_deltakernel (double *rad*, double *dilation*)
- 4.14.1.16 double ObjectManager::ibm_findepsilon (int *ib*, GridObj & *g*)
- 4.14.1.17 void ObjectManager::ibm_findsupport (int *ib*, int *m*, GridObj & *g*)
- 4.14.1.18 void ObjectManager::ibm_initialise (GridObj & *g*)
- 4.14.1.19 void ObjectManager::ibm_interpol (int *ib*, GridObj & *g*)

- 4.14.1.20 void ObjectManager::ibm_jacowire (int *ib*, GridObj & *g*)
- 4.14.1.21 void ObjectManager::ibm_move_bodies (GridObj & *g*)
- 4.14.1.22 void ObjectManager::ibm_position_update (int *ib*, GridObj & *g*)
- 4.14.1.23 void ObjectManager::ibm_position_update_grp (int *group*, GridObj & *g*)
- 4.14.1.24 void ObjectManager::ibm_spread (int *ib*, GridObj & *g*)
- 4.14.1.25 void ObjectManager::io_readInCloud (PCpts * *_PCpts*, eObjectType *objtype*)
- 4.14.1.26 void ObjectManager::io_restart (bool *IO_flag*, int *level*)
- 4.14.1.27 void ObjectManager::io_vtk_IBwriter (double *tval*)
- 4.14.1.28 void ObjectManager::io_write_body_pos (int *timestep*)
- 4.14.1.29 void ObjectManager::io_write_lift_drag (int *timestep*)

4.14.2 Friends And Related Function Documentation

- 4.14.2.1 friend class GridObj [friend]

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

- [ObjectManager.h](#)
- [ObjectManager.cpp](#)
- [ObjectManager_init_bflbody.cpp](#)
- [ObjectManager_init_ibmbody.cpp](#)
- [ObjectManager_ops_ibm.cpp](#)
- [ObjectManager_ops_ibmflex.cpp](#)
- [ObjectManager_ops_io.cpp](#)

4.15 PCpts Class Reference

```
#include <PCpts.h>
```

Public Member Functions

- [PCpts](#) (void)
- [~PCpts](#) (void)

Public Attributes

- `std::vector< double > x`
- `std::vector< double > y`
- `std::vector< double > z`

4.15.1 Constructor & Destructor Documentation

4.15.1.1 `PCpts::PCpts(void)` `[inline]`

4.15.1.2 `PCpts::~~PCpts(void)` `[inline]`

4.15.2 Member Data Documentation

4.15.2.1 `std::vector<double> PCpts::x`

4.15.2.2 `std::vector<double> PCpts::y`

4.15.2.3 `std::vector<double> PCpts::z`

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

- [PCpts.h](#)

4.16 MpiManager::phdf5_struct Struct Reference

```
#include <MpiManager.h>
```

Public Attributes

- `int i_start`
- `int i_end`
- `int j_start`
- `int j_end`
- `int k_start`
- `int k_end`
- `int halo_min`
- `int halo_max`
- `int level`
- `int region`
- `unsigned int writable_data_count = 0`

4.16.1 Member Data Documentation

4.16.1.1 `int MpiManager::phdf5_struct::halo_max`

4.16.1.2 `int MpiManager::phdf5_struct::halo_min`

4.16.1.3 `int MpiManager::phdf5_struct::i_end`

4.16.1.4 `int MpiManager::phdf5_struct::i_start`

4.16.1.5 `int MpiManager::phdf5_struct::j_end`

4.16.1.6 `int MpiManager::phdf5_struct::j_start`

4.16.1.7 `int MpiManager::phdf5_struct::k_end`

4.16.1.8 `int MpiManager::phdf5_struct::k_start`

4.16.1.9 `int MpiManager::phdf5_struct::level`

4.16.1.10 `int MpiManager::phdf5_struct::region`

4.16.1.11 `unsigned int MpiManager::phdf5_struct::writable_data_count = 0`

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

- [MpiManager.h](#)

Chapter 5

File Documentation

5.1 BFLBody.cpp File Reference

```
#include "../inc/stdafx.h"  
#include "../inc/globalvars.h"  
#include "../inc/MpiManager.h"  
#include "../inc/BFLBody.h"
```

5.2 BFLBody.h File Reference

```
#include "BFLMarker.h"  
#include "Body.h"  
#include "PCpts.h"  
#include "ObjectManager.h"
```

Classes

- class [BFLBody](#)

5.3 BFLMarker.cpp File Reference

```
#include "../inc/stdafx.h"  
#include "../inc/BFLMarker.h"  
#include "../inc/BFLBody.h"  
#include "../inc/ObjectManager.h"
```

5.4 BFLMarker.h File Reference

```
#include "Marker.h"
```

Classes

- class [BFLMarker](#)

5.5 Body.h File Reference

```
#include <vector>
```

Classes

- class [MarkerData](#)
- class [Body](#)< [MarkerType](#) >

5.6 definitions.h File Reference

```
#include <time.h>
#include <iostream>
#include <fstream>
#include <vector>
#include <iomanip>
#include <math.h>
#include <string>
#include <mpi.h>
```

Macros

- #define [LUMA_VERSION](#) "1.1.1 (WORKING VERSION)"
- #define [L_PI](#) 3.14159265358979323846
- #define [L_BUILD_FOR_MPI](#)
- #define [L_out_every](#) 100
- #define [L_output_precision](#) 3
- #define [L_HDF5_OUTPUT](#)
- #define [L_out_every_probe](#) 250
- #define [L_grav_force](#) 1e-10
- #define [L_grav_direction](#) [eXDirection](#)
- #define [L_restart_out_every](#) 10000
- #define [L_USE_KBC_COLLISION](#)
- #define [L_Timesteps](#) 500
- #define [L_Xcores](#) 2
- #define [L_Ycores](#) 2
- #define [L_Zcores](#) 2
- #define [L_dims](#) 3
- #define [L_N](#) 100
- #define [L_M](#) 60
- #define [L_K](#) 60
- #define [L_a_x](#) 0
- #define [L_b_x](#) 5

- `#define L_a_y 0`
- `#define L_b_y 3`
- `#define L_a_z 0`
- `#define L_b_z 3`
- `#define L_u_ref 0.04`
- `#define L_u_max 0.06`
- `#define L_u_0x L_u_ref`
- `#define L_u_0y 0`
- `#define L_u_0z 0`
- `#define L_rho_in 1`
- `#define L_Re 5000`
- `#define L_CHEAP_NEAREST_NODE_DETECTION`
- `#define L_ibb_on_grid_lev 2`
- `#define L_ibb_on_grid_reg 0`
- `#define L_start_ibb_x 0.3`
- `#define L_start_ibb_y 0.2`
- `#define L_centre_ibb_z 0.5`
- `#define L_ibb_length 0.5`
- `#define L_ibb_scale_direction eXDirection`
- `#define L_ibb_length_ref 0.5`
- `#define L_num_markers 19`
- `#define L_ibb_deform false`
- `#define L_ibb_flex_rigid false`
- `#define L_ibb_x 75.0`
- `#define L_ibb_y 75.0`
- `#define L_ibb_z 0.0`
- `#define L_ibb_w 10.0`
- `#define L_ibb_l 10.0`
- `#define L_ibb_d 0.0`
- `#define L_ibb_r 10.0`
- `#define L_ibb_filament_length 0.2`
- `#define L_ibb_filament_start_x 0.3`
- `#define L_ibb_filament_start_y 0.0`
- `#define L_ibb_filament_start_z 0.0`
- `#define L_ibb_angle_vert 90`
- `#define L_ibb_angle_horz 0`
- `#define L_start_BC 2`
- `#define L_end_BC 0`
- `#define L_ibb_delta_rho 1.0`
- `#define L_ibb_EI 2.0`
- `#define L_FREESTREAM_TUNNEL`
- `#define L_INLET_ON`
- `#define L_OUTLET_ON`
- `#define L_wall_thickness 1`
- `#define L_SOLID_BLOCK_ON`
- `#define L_block_on_grid_lev 2`
- `#define L_block_on_grid_reg 0`
- `#define L_block_x_min 20`
- `#define L_block_x_max 60`
- `#define L_block_y_min 4`
- `#define L_block_y_max 44`
- `#define L_block_z_min 10`
- `#define L_block_z_max 50`
- `#define L_object_on_grid_lev 2`
- `#define L_object_on_grid_reg 0`

- `#define L_start_object_x 20`
- `#define L_start_object_y 4`
- `#define L_centre_object_z 30`
- `#define L_object_length 80`
- `#define L_object_scale_direction eXDirection`
- `#define L_object_length_ref 80`
- `#define L_bfl_on_grid_lev 1`
- `#define L_bfl_on_grid_reg 0`
- `#define L_start_bfl_x 50`
- `#define L_start_bfl_y 100`
- `#define L_centre_bfl_z 20`
- `#define L_bfl_length 50`
- `#define L_bfl_scale_direction eXDirection`
- `#define L_bfl_length_ref 10`
- `#define L_NumLev 2`
- `#define L_NumReg 1`
- `#define L_nVels 27`
- `#define L_MPI_dir 26`

Variables

- static const int `nProbes` [3] = {3, 3, 3}
- static const int `xProbeLims` [2] = {90, 270}
- static const int `yProbeLims` [2] = {15, 45}
- static const int `zProbeLims` [2] = {30, 120}
- static const int `RefXstart` [L_NumLev][L_NumReg] = { {30}, {10} }
- static const int `RefXend` [L_NumLev][L_NumReg] = { {70}, {70} }
- static const int `RefYstart` [L_NumLev][L_NumReg] = { {0}, {0} }
- static const int `RefYend` [L_NumLev][L_NumReg] = { {20}, {30} }
- static int `RefZstart` [L_NumLev][L_NumReg] = { {20}, {5} }
- static int `RefZend` [L_NumLev][L_NumReg] = { {40}, {35} }

5.6.1 Macro Definition Documentation

5.6.1.1 `#define L_a_x 0`

5.6.1.2 `#define L_a_y 0`

5.6.1.3 `#define L_a_z 0`

5.6.1.4 `#define L_b_x 5`

5.6.1.5 `#define L_b_y 3`

5.6.1.6 `#define L_b_z 3`

5.6.1.7 `#define L_bfl_length 50`

5.6.1.8 `#define L_bfl_length_ref 10`

- 5.6.1.9 `#define L_bfl_on_grid_lev 1`
- 5.6.1.10 `#define L_bfl_on_grid_reg 0`
- 5.6.1.11 `#define L_bfl_scale_direction eXDirection`
- 5.6.1.12 `#define L_block_on_grid_lev 2`
- 5.6.1.13 `#define L_block_on_grid_reg 0`
- 5.6.1.14 `#define L_block_x_max 60`
- 5.6.1.15 `#define L_block_x_min 20`
- 5.6.1.16 `#define L_block_y_max 44`
- 5.6.1.17 `#define L_block_y_min 4`
- 5.6.1.18 `#define L_block_z_max 50`
- 5.6.1.19 `#define L_block_z_min 10`
- 5.6.1.20 `#define L_BUILD_FOR_MPI`
- 5.6.1.21 `#define L_centre_bfl_z 20`
- 5.6.1.22 `#define L_centre_ibb_z 0.5`
- 5.6.1.23 `#define L_centre_object_z 30`
- 5.6.1.24 `#define L_CHEAP_NEAREST_NODE_DETECTION`
- 5.6.1.25 `#define L_dims 3`
- 5.6.1.26 `#define L_end_BC 0`
- 5.6.1.27 `#define L_FREESTREAM_TUNNEL`
- 5.6.1.28 `#define L_grav_direction eXDirection`
- 5.6.1.29 `#define L_grav_force 1e-10`
- 5.6.1.30 `#define L_HDF5_OUTPUT`
- 5.6.1.31 `#define L_ibb_angle_horz 0`

5.6.1.32 `#define L_ibb_angle_vert 90`

5.6.1.33 `#define L_ibb_d 0.0`

5.6.1.34 `#define L_ibb_deform false`

5.6.1.35 `#define L_ibb_delta_rho 1.0`

5.6.1.36 `#define L_ibb_EI 2.0`

5.6.1.37 `#define L_ibb_filament_length 0.2`

5.6.1.38 `#define L_ibb_filament_start_x 0.3`

5.6.1.39 `#define L_ibb_filament_start_y 0.0`

5.6.1.40 `#define L_ibb_filament_start_z 0.0`

5.6.1.41 `#define L_ibb_flex_rigid false`

5.6.1.42 `#define L_ibb_I 10.0`

5.6.1.43 `#define L_ibb_length 0.5`

5.6.1.44 `#define L_ibb_length_ref 0.5`

5.6.1.45 `#define L_ibb_on_grid_lev 2`

5.6.1.46 `#define L_ibb_on_grid_reg 0`

5.6.1.47 `#define L_ibb_r 10.0`

5.6.1.48 `#define L_ibb_scale_direction eXDirection`

5.6.1.49 `#define L_ibb_w 10.0`

5.6.1.50 `#define L_ibb_x 75.0`

5.6.1.51 `#define L_ibb_y 75.0`

5.6.1.52 `#define L_ibb_z 0.0`

5.6.1.53 `#define L_INLET_ON`

5.6.1.54 `#define L_K 60`

- 5.6.1.55 `#define L_M 60`
- 5.6.1.56 `#define L_MPI_dir 26`
- 5.6.1.57 `#define L_N 100`
- 5.6.1.58 `#define L_num_markers 19`
- 5.6.1.59 `#define L_NumLev 2`
- 5.6.1.60 `#define L_NumReg 1`
- 5.6.1.61 `#define L_nVels 27`
- 5.6.1.62 `#define L_object_length 80`
- 5.6.1.63 `#define L_object_length_ref 80`
- 5.6.1.64 `#define L_object_on_grid_lev 2`
- 5.6.1.65 `#define L_object_on_grid_reg 0`
- 5.6.1.66 `#define L_object_scale_direction eXDirection`
- 5.6.1.67 `#define L_out_every 100`
- 5.6.1.68 `#define L_out_every_probe 250`
- 5.6.1.69 `#define L_OUTLET_ON`
- 5.6.1.70 `#define L_output_precision 3`
- 5.6.1.71 `#define L_PI 3.14159265358979323846`
- 5.6.1.72 `#define L_Re 5000`
- 5.6.1.73 `#define L_restart_out_every 10000`
- 5.6.1.74 `#define L_rho_in 1`
- 5.6.1.75 `#define L_SOLID_BLOCK_ON`
- 5.6.1.76 `#define L_start_BC 2`
- 5.6.1.77 `#define L_start_bfl_x 50`

5.6.1.78 `#define L_start_bfl_y 100`

5.6.1.79 `#define L_start_ibb_x 0.3`

5.6.1.80 `#define L_start_ibb_y 0.2`

5.6.1.81 `#define L_start_object_x 20`

5.6.1.82 `#define L_start_object_y 4`

5.6.1.83 `#define L_Timesteps 500`

5.6.1.84 `#define L_u_0x L_u_ref`

5.6.1.85 `#define L_u_0y 0`

5.6.1.86 `#define L_u_0z 0`

5.6.1.87 `#define L_u_max 0.06`

5.6.1.88 `#define L_u_ref 0.04`

5.6.1.89 `#define L_USE_KBC_COLLISION`

5.6.1.90 `#define L_wall_thickness 1`

5.6.1.91 `#define L_Xcores 2`

5.6.1.92 `#define L_Ycores 2`

5.6.1.93 `#define L_Zcores 2`

5.6.1.94 `#define LUMA_VERSION "1.1.1 (WORKING VERSION)"`

5.6.2 Variable Documentation

5.6.2.1 `const int nProbes[3] = {3, 3, 3} [static]`

5.6.2.2 `const int RefXend[L_NumLev][L_NumReg] = { {70}, {70} } [static]`

5.6.2.3 `const int RefXstart[L_NumLev][L_NumReg] = { {30}, {10} } [static]`

5.6.2.4 `const int RefYend[L_NumLev][L_NumReg] = { {20}, {30} } [static]`

5.6.2.5 `const int RefYstart[L_NumLev][L_NumReg] = { {0}, {0} } [static]`

5.6.2.6 `int RefZend[L_NumLev][L_NumReg] = { {40}, {35} } [static]`

5.6.2.7 `int RefZstart[L_NumLev][L_NumReg] = { {20}, {5} } [static]`

5.6.2.8 `const int xProbeLims[2] = {90, 270} [static]`

5.6.2.9 `const int yProbeLims[2] = {15, 45} [static]`

5.6.2.10 `const int zProbeLims[2] = {30, 120} [static]`

5.7 Gen_init_globalvars.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/definitions.h"
#include "../inc/globalvars.h"
```

Variables

- `const int c [3][L_nVels]`
- `const double w [L_nVels]`
- `const double cs = 1.0 / sqrt(3.0)`

5.7.1 Variable Documentation

5.7.1.1 `const int c[3][L_nVels]`

Initial value:

```
=
{
    {1, -1, 0, 0, 0, 0, 0, 0, 0, 0, 1, -1, 1, -1, 1, -1, 1, -1, 1, -1, -1, 1,
    -1, 1, 1, -1, 0},
    {0, 0, 1, -1, 0, 0, 1, -1, 1, -1, 0, 0, 0, 0, 1, -1, -1, 1, 1, -1, -1, 1,
    1, -1, -1, 1, 0},
    {0, 0, 0, 1, -1, 1, -1, 1, -1, -1, 1, 1, -1, -1, 1, 0, 0, 0, 0, 0, 1, -1, 1, -1,
    1, -1, 1, -1, 0}
}
```

5.7.1.2 `const double cs = 1.0 / sqrt(3.0)`

5.7.1.3 `const double w[L_nVels]`

Initial value:

```
=
{2.0/27.0, 2.0/27.0, 2.0/27.0, 2.0/27.0, 2.0/27.0, 2.0/27.0,
1.0/54.0, 1.0/54.0, 1.0/54.0, 1.0/54.0, 1.0/54.0, 1.0/54.0, 1.0/54.0, 1.0/54.0, 1.0/54.0, 1.0/54.0,
1.0/54.0, 1.0/54.0,
1.0/216.0, 1.0/216.0, 1.0/216.0, 1.0/216.0, 1.0/216.0, 1.0/216.0, 1.0/216.0, 1.0/216.0,
8.0/27.0}
```

5.8 globalvars.h File Reference

```
#include "definitions.h"
```

Variables

- const int [c](#) [3][[L_nVels](#)]
- const double [w](#) [[L_nVels](#)]
- const double [cs](#)

5.8.1 Variable Documentation

5.8.1.1 [const int c\[3\]\[L_nVels\]](#)

5.8.1.2 [const double cs](#)

5.8.1.3 [const double w\[L_nVels\]](#)

5.9 GridObj.cpp File Reference

```
#include "../inc/stdafx.h"  
#include "../inc/definitions.h"  
#include "../inc/GridObj.h"  
#include "../inc/MpiManager.h"
```

5.10 GridObj.h File Reference

```
#include <vector>  
#include "IVector.h"  
#include "IBBody.h"  
#include <iostream>  
#include <fstream>  
#include "hdf5luma.h"
```

Classes

- class [GridObj](#)

Enumerations

- enum [eType](#) {
 [eSolid](#), [eFluid](#), [eRefined](#), [eTransitionToCoarser](#),
 [eTransitionToFiner](#), [eBFL](#), [eSymmetry](#), [eInlet](#),
 [eOutlet](#), [eRefinedSolid](#), [eRefinedSymmetry](#), [eRefinedInlet](#) }
- enum [eBCType](#) {
 [eBCAll](#), [eBCSolidSymmetry](#), [eBCInlet](#), [eBCOutlet](#),
 [eBCInletOutlet](#), [eBCBFL](#) }

5.10.1 Enumeration Type Documentation

5.10.1.1 enum [eBCType](#)

Enumerator

[eBCAll](#)

[eBCSolidSymmetry](#)

[eBCInlet](#)

[eBCOutlet](#)

[eBCInletOutlet](#)

[eBCBFL](#)

5.10.1.2 enum [eType](#)

[GridObj](#) class which represents a lattice

Enumerator

[eSolid](#)

[eFluid](#)

[eRefined](#)

[eTransitionToCoarser](#)

[eTransitionToFiner](#)

[eBFL](#)

[eSymmetry](#)

[eInlet](#)

[eOutlet](#)

[eRefinedSolid](#)

[eRefinedSymmetry](#)

[eRefinedInlet](#)

5.11 GridObj_init_grids.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/GridObj.h"
#include "../inc/MpiManager.h"
#include "../inc/definitions.h"
#include "../inc/globalvars.h"
#include <fstream>
#include <iostream>
#include <sstream>
#include <math.h>
```

5.12 GridObj_ops_boundary.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/GridObj.h"
#include "../inc/definitions.h"
#include "../inc/globalvars.h"
#include "../inc/BFLBody.h"
#include "../inc/ObjectManager.h"
#include <numeric>
```

5.13 GridObj_ops_io.cpp File Reference

```
#include "../inc/stdafx.h"
#include <sstream>
#include "../inc/GridObj.h"
#include "../inc/MpiManager.h"
#include "../inc/ObjectManager.h"
#include "../inc/definitions.h"
#include "../inc/globalvars.h"
#include "../inc/hdf5luma.h"
```

5.14 GridObj_ops_lbm.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/GridObj.h"
#include "../inc/definitions.h"
#include "../inc/globalvars.h"
#include "../inc/IVector.h"
#include "../inc/ObjectManager.h"
#include "../inc/MpiManager.h"
```

5.15 GridUtils.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/GridObj.h"
#include <sstream>
#include <iostream>
#include "../inc/definitions.h"
#include "../inc/globalvars.h"
#include "../inc/MpiManager.h"
```

5.16 GridUtils.h File Reference

```
#include "stdafx.h"
#include "definitions.h"
#include "GridObj.h"
#include "hdf5luma.h"
```

Classes

- class [GridUtils](#)

Enumerations

- enum [eCartesianDirection](#) { [eXDirection](#), [eYDirection](#), [eZDirection](#) }
- enum [eMinMax](#) { [eMinimum](#), [eMaximum](#) }

5.16.1 Enumeration Type Documentation

5.16.1.1 enum [eCartesianDirection](#)

Enumerator

eXDirection

eYDirection

eZDirection

5.16.1.2 enum [eMinMax](#)

Enumerator

eMinimum

eMaximum

5.17 hdf5luma.h File Reference

```
#include "hdf5.h"
#include "MpiManager.h"
```

Macros

- `#define H5_BUILT_AS_DYNAMIC_LIB`
- `#define HDF5_EXT_ZLIB`
- `#define HDF5_EXT_SZIP`

Enumerations

- enum `eHdf5SlabType` {
`eScalar`, `eVector`, `eProductVector`, `ePosX`,
`ePosY`, `ePosZ` }

Functions

- `template<typename T >`
`void hdf5_writeDataSet (hid_t &memspace, hid_t &filespace, hid_t &dataset_id, eHdf5SlabType slab_type,`
`int N_lim, int M_lim, int K_lim, int N_mod, int M_mod, int K_mod, GridObj *g, T *data, hid_t hdf_datatype, int`
`TL_thickness, MpiManager::phdf5_struct hdf_data)`

5.17.1 Macro Definition Documentation

5.17.1.1 `#define H5_BUILT_AS_DYNAMIC_LIB`

5.17.1.2 `#define HDF5_EXT_SZIP`

5.17.1.3 `#define HDF5_EXT_ZLIB`

5.17.2 Enumeration Type Documentation

5.17.2.1 enum `eHdf5SlabType`

Enumerator

eScalar

eVector

eProductVector

ePosX

ePosY

ePosZ

5.17.3 Function Documentation

5.17.3.1 `template<typename T> void hdf5_writeDataSet (hid_t & memspace, hid_t & filespace, hid_t & dataset_id, eHdf5SlabType slab_type, int N_lim, int M_lim, int K_lim, int N_mod, int M_mod, int K_mod, GridObj * g, T * data, hid_t hdf_datatype, int TL_thickness, MPIManager::phdf5_struct hdf_data)`

5.18 IBody.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/IBody.h"
#include "../inc/definitions.h"
#include <math.h>
```

5.19 IBody.h File Reference

```
#include "IBMarker.h"
#include "Body.h"
#include "PCpts.h"
#include "BFLBody.h"
```

Classes

- class [IBody](#)

5.20 IBMarker.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/IBMarker.h"
#include "../inc/definitions.h"
```

5.21 IBMarker.h File Reference

```
#include <vector>
#include "Marker.h"
```

Classes

- class [IBMarker](#)

5.22 IVector.h File Reference

```
#include <vector>
```

Classes

- class [IVector](#)< [GenTyp](#) >

5.23 main_lbm.cpp File Reference

```
#include "../inc/stdafx.h"  
#include "../inc/definitions.h"  
#include "../inc/globalvars.h"  
#include "../inc/GridObj.h"  
#include "../inc/MpiManager.h"  
#include "../inc/ObjectManager.h"
```

Functions

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

5.23.1 Function Documentation

5.23.1.1 int main (int *argc*, char * *argv*[])

5.24 Marker.h File Reference

```
#include <vector>
```

Classes

- class [Marker](#)

5.25 Mpi_buffer_pack.cpp File Reference

```
#include "../inc/stdafx.h"  
#include <mpi.h>  
#include "../inc/definitions.h"  
#include <iostream>  
#include <fstream>  
#include "../inc/MpiManager.h"  
#include "../inc/GridObj.h"
```

5.26 `Mpi_buffer_size_recv.cpp` File Reference

```
#include "../inc/stdafx.h"
#include <mpi.h>
#include "../inc/definitions.h"
#include <iostream>
#include <fstream>
#include "../inc/MpiManager.h"
#include "../inc/GridObj.h"
```

5.27 `Mpi_buffer_size_send.cpp` File Reference

```
#include "../inc/stdafx.h"
#include <mpi.h>
#include "../inc/definitions.h"
#include <iostream>
#include <fstream>
#include "../inc/MpiManager.h"
#include "../inc/GridObj.h"
```

5.28 `Mpi_buffer_unpk.cpp` File Reference

```
#include "../inc/stdafx.h"
#include <mpi.h>
#include "../inc/definitions.h"
#include <iostream>
#include <fstream>
#include "../inc/MpiManager.h"
#include "../inc/GridObj.h"
```

5.29 `MpiManager.cpp` File Reference

```
#include "../inc/stdafx.h"
#include <mpi.h>
#include <iostream>
#include <fstream>
#include "../inc/definitions.h"
#include "../inc/MpiManager.h"
#include "../inc/GridObj.h"
#include "../inc/globalvars.h"
```

5.30 `MpiManager.h` File Reference

```
#include "definitions.h"
#include "GridObj.h"
```

Classes

- class [MpiManager](#)
- struct [MpiManager::phdf5_struct](#)
- struct [MpiManager::layer_edges](#)
- struct [MpiManager::buffer_struct](#)

Macros

- `#define range_i_left i = 0; i < GridUtils::downToLimit((int)pow(2, g->level + 1), N_lim); i++`
- `#define range_j_down j = 0; j < GridUtils::downToLimit((int)pow(2, g->level + 1), M_lim); j++`
- `#define range_k_front k = 0; k < GridUtils::downToLimit((int)pow(2, g->level + 1), K_lim); k++`
- `#define range_i_right i = GridUtils::upToZero(N_lim - (int)pow(2, g->level + 1)); i < N_lim; i++`
- `#define range_j_up j = GridUtils::upToZero(M_lim - (int)pow(2, g->level + 1)); j < M_lim; j++`
- `#define range_k_back k = GridUtils::upToZero(K_lim - (int)pow(2, g->level + 1)); k < K_lim; k++`

5.30.1 Macro Definition Documentation

5.30.1.1 `#define range_i_left i = 0; i < GridUtils::downToLimit((int)pow(2, g->level + 1), N_lim); i++`

5.30.1.2 `#define range_i_right i = GridUtils::upToZero(N_lim - (int)pow(2, g->level + 1)); i < N_lim; i++`

5.30.1.3 `#define range_j_down j = 0; j < GridUtils::downToLimit((int)pow(2, g->level + 1), M_lim); j++`

5.30.1.4 `#define range_j_up j = GridUtils::upToZero(M_lim - (int)pow(2, g->level + 1)); j < M_lim; j++`

5.30.1.5 `#define range_k_back k = GridUtils::upToZero(K_lim - (int)pow(2, g->level + 1)); k < K_lim; k++`

5.30.1.6 `#define range_k_front k = 0; k < GridUtils::downToLimit((int)pow(2, g->level + 1), K_lim); k++`

5.31 ObjectManager.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/ObjectManager.h"
```

5.32 ObjectManager.h File Reference

```
#include <vector>
#include "IBody.h"
#include "Body.h"
#include "BFLBody.h"
#include "IVector.h"
```


Classes

- class [ObjectManager](#)

Enumerations

- enum [eObjectType](#) { [eBBBCloud](#), [eBFLCloud](#), [eIBBCloud](#) }

5.32.1 Enumeration Type Documentation

5.32.1.1 enum eObjectType

Enumerator

eBBBCloud

eBFLCloud

eIBBCloud

5.33 ObjectManager_init_bflbody.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/ObjectManager.h"
```

5.34 ObjectManager_init_ibmbody.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/ObjectManager.h"
#include "../inc/definitions.h"
```

5.35 ObjectManager_ops_ibm.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/GridObj.h"
#include "../inc/ObjectManager.h"
#include "../inc/definitions.h"
#include "../inc/MpiManager.h"
```

5.36 ObjectManager_ops_ibmflex.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/GridObj.h"
#include "../inc/ObjectManager.h"
#include "../inc/definitions.h"
#include "../inc/MpiManager.h"
```

Macros

- `#define SWAP(a, b) {dum=(a);(a)=(b);(b)=dum;}`
- `#define TINY 1.0e-20`
- `#define SWAP(a, b) {dum=(a);(a)=(b);(b)=dum;}`

5.36.1 Macro Definition Documentation

5.36.1.1 `#define SWAP(a, b) {dum=(a);(a)=(b);(b)=dum;}`

5.36.1.2 `#define SWAP(a, b) {dum=(a);(a)=(b);(b)=dum;}`

5.36.1.3 `#define TINY 1.0e-20`

5.37 ObjectManager_ops_io.cpp File Reference

```
#include "../inc/stdafx.h"
#include "../inc/ObjectManager.h"
#include "../inc/MpiManager.h"
#include <sstream>
```

5.38 PCpts.h File Reference

```
#include "stdafx.h"
```

Classes

- class [PCpts](#)

5.39 stdafx.cpp File Reference

```
#include "../inc/stdafx.h"
```

5.40 stdafx.h File Reference

```
#include <algorithm>
#include <cmath>
#include <vector>
#include <iostream>
#include <fstream>
#include <stdlib.h>
#include <cstring>
#include <stdio.h>
#include "../inc/GridUtils.h"
```

Macros

- `#define LUMA_FAILED 12345`

Functions

- `template<typename NumType >`
`static bool is_nan (NumType n)`

5.40.1 Macro Definition Documentation

5.40.1.1 `#define LUMA_FAILED 12345`

5.40.2 Function Documentation

5.40.2.1 `template<typename NumType > static bool is_nan (NumType n)` `[inline], [static]`

5.41 targetver.h File Reference

```
#include <SDKDDKVer.h>
```


Index

- _Owner
 - Body, [10](#)
- ~BFLBody
 - BFLBody, [8](#)
- ~BFLMarker
 - BFLMarker, [9](#)
- ~Body
 - Body, [10](#)
- ~GridObj
 - GridObj, [13](#)
- ~IBBody
 - IBBody, [20](#)
- ~IBMarker
 - IBMarker, [22](#)
- ~IVector
 - IVector, [24](#)
- ~Marker
 - Marker, [25](#)
- ~MarkerData
 - MarkerData, [26](#)
- ~PCpts
 - PCpts, [33](#)
- add
 - GridUtils, [17](#)
- addMarker
 - Body, [10](#)
 - IBBody, [20](#)
- BCs
 - IBBody, [21](#)
- BFLBody, [7](#)
 - ~BFLBody, [8](#)
 - BFLBody, [8](#)
 - BFLMarker, [9](#)
 - computeQ, [8](#)
 - GridObj, [8](#)
 - Q, [8](#)
- BFLBody.cpp, [35](#)
- BFLBody.h, [35](#)
- BFLMarker, [8](#)
 - ~BFLMarker, [9](#)
 - BFLBody, [9](#)
 - BFLMarker, [9](#)
- BFLMarker.cpp, [35](#)
- BFLMarker.h, [35](#)
- bc_applyBfl
 - GridObj, [13](#)
- bc_applyBounceBack
 - GridObj, [13](#)
- bc_applyExtrapolation
 - GridObj, [13](#)
- bc_applyNrbc
 - GridObj, [13](#)
- bc_applyRegularised
 - GridObj, [13](#)
- bc_applySpecReflect
 - GridObj, [13](#)
- bc_applyZouHe
 - GridObj, [14](#)
- bc_getWallDensityForRBC
 - GridObj, [14](#)
- bc_solidSiteReset
 - GridObj, [14](#)
- bfl_build_body
 - ObjectManager, [31](#)
- Body
 - _Owner, [10](#)
 - ~Body, [10](#)
 - addMarker, [10](#)
 - Body, [10](#)
 - closed_surface, [10](#)
 - getMarkerData, [10](#)
 - isInVoxel, [10](#)
 - isVoxelMarkerVoxel, [10](#)
 - markerAdder, [10](#)
 - markers, [10](#)
 - spacing, [10](#)
- Body< MarkerType >, [9](#)
- Body.h, [36](#)
- buffer_recv_info
 - MpiManager, [28](#)
- buffer_send_info
 - MpiManager, [28](#)
- c
 - Gen_init_globalvars.cpp, [43](#)
 - globalvars.h, [44](#)
- closed_surface
 - Body, [10](#)
- computeQ
 - BFLBody, [8](#)
- createOutputDirectory
 - GridUtils, [17](#)
- crossprod
 - GridUtils, [17](#)
- cs
 - Gen_init_globalvars.cpp, [43](#)
 - globalvars.h, [44](#)

[definitions.h](#), [36](#)
[L_BUILD_FOR_MPI](#), [39](#)
[L_CHEAP_NEAREST_NODE_DETECTION](#), [39](#)
[L_FREESTREAM_TUNNEL](#), [39](#)
[L_HDF5_OUTPUT](#), [39](#)
[L_INLET_ON](#), [40](#)
[L_MPI_dir](#), [41](#)
[L_NumLev](#), [41](#)
[L_NumReg](#), [41](#)
[L_OUTLET_ON](#), [41](#)
[L_PI](#), [41](#)
[L_Re](#), [41](#)
[L_SOLID_BLOCK_ON](#), [41](#)
[L_Timesteps](#), [42](#)
[L_USE_KBC_COLLISION](#), [42](#)
[L_Xcores](#), [42](#)
[L_Ycores](#), [42](#)
[L_Zcores](#), [42](#)
[L_a_x](#), [38](#)
[L_a_y](#), [38](#)
[L_a_z](#), [38](#)
[L_b_x](#), [38](#)
[L_b_y](#), [38](#)
[L_b_z](#), [38](#)
[L_bfl_length](#), [38](#)
[L_bfl_length_ref](#), [38](#)
[L_bfl_on_grid_lev](#), [38](#)
[L_bfl_on_grid_reg](#), [39](#)
[L_bfl_scale_direction](#), [39](#)
[L_block_on_grid_lev](#), [39](#)
[L_block_on_grid_reg](#), [39](#)
[L_block_x_max](#), [39](#)
[L_block_x_min](#), [39](#)
[L_block_y_max](#), [39](#)
[L_block_y_min](#), [39](#)
[L_block_z_max](#), [39](#)
[L_block_z_min](#), [39](#)
[L_centre_bfl_z](#), [39](#)
[L_centre_ibb_z](#), [39](#)
[L_centre_object_z](#), [39](#)
[L_dims](#), [39](#)
[L_end_BC](#), [39](#)
[L_grav_direction](#), [39](#)
[L_grav_force](#), [39](#)
[L_ibb_EI](#), [40](#)
[L_ibb_angle_horz](#), [39](#)
[L_ibb_angle_vert](#), [39](#)
[L_ibb_d](#), [40](#)
[L_ibb_deform](#), [40](#)
[L_ibb_delta_rho](#), [40](#)
[L_ibb_filament_length](#), [40](#)
[L_ibb_filament_start_x](#), [40](#)
[L_ibb_filament_start_y](#), [40](#)
[L_ibb_filament_start_z](#), [40](#)
[L_ibb_flex_rigid](#), [40](#)
[L_ibb_I](#), [40](#)
[L_ibb_length](#), [40](#)
[L_ibb_length_ref](#), [40](#)
[L_ibb_on_grid_lev](#), [40](#)
[L_ibb_on_grid_reg](#), [40](#)
[L_ibb_r](#), [40](#)
[L_ibb_scale_direction](#), [40](#)
[L_ibb_w](#), [40](#)
[L_ibb_x](#), [40](#)
[L_ibb_y](#), [40](#)
[L_ibb_z](#), [40](#)
[L_K](#), [40](#)
[L_M](#), [40](#)
[L_N](#), [41](#)
[L_nVels](#), [41](#)
[L_num_markers](#), [41](#)
[L_object_length](#), [41](#)
[L_object_length_ref](#), [41](#)
[L_object_on_grid_lev](#), [41](#)
[L_object_on_grid_reg](#), [41](#)
[L_object_scale_direction](#), [41](#)
[L_out_every](#), [41](#)
[L_out_every_probe](#), [41](#)
[L_output_precision](#), [41](#)
[L_restart_out_every](#), [41](#)
[L_rho_in](#), [41](#)
[L_start_BC](#), [41](#)
[L_start_bfl_x](#), [41](#)
[L_start_bfl_y](#), [41](#)
[L_start_ibb_x](#), [42](#)
[L_start_ibb_y](#), [42](#)
[L_start_object_x](#), [42](#)
[L_start_object_y](#), [42](#)
[L_u_0x](#), [42](#)
[L_u_0y](#), [42](#)
[L_u_0z](#), [42](#)
[L_u_max](#), [42](#)
[L_u_ref](#), [42](#)
[L_wall_thickness](#), [42](#)
[LUMA_VERSION](#), [42](#)
[nProbes](#), [42](#)
[RefXend](#), [42](#)
[RefXstart](#), [42](#)
[RefYend](#), [42](#)
[RefYstart](#), [42](#)
[RefZend](#), [42](#)
[RefZstart](#), [43](#)
[xProbeLims](#), [43](#)
[yProbeLims](#), [43](#)
[zProbeLims](#), [43](#)
[deformable](#)
 [IBBody](#), [21](#)
[delta_rho](#)
 [IBBody](#), [21](#)
[deltaval](#)
 [IBMarker](#), [22](#)
[desired_vel](#)
 [IBMarker](#), [22](#)
[destroyInstance](#)
 [MpiManager](#), [28](#)
 [ObjectManager](#), [31](#)

- dilation
 - IBMarker, [22](#)
- dir_reflect
 - GridUtils, [19](#)
- dotprod
 - GridUtils, [17](#)
- downToLimit
 - GridUtils, [17](#)
- dt
 - GridObj, [15](#)
- eBBBCloud
 - ObjectManager.h, [53](#)
- eBCAll
 - GridObj.h, [45](#)
- eBCBFL
 - GridObj.h, [45](#)
- eBCInlet
 - GridObj.h, [45](#)
- eBCInletOutlet
 - GridObj.h, [45](#)
- eBCOutlet
 - GridObj.h, [45](#)
- eBCSolidSymmetry
 - GridObj.h, [45](#)
- eBCType
 - GridObj.h, [45](#)
- eBFLCloud
 - ObjectManager.h, [53](#)
- eBFL
 - GridObj.h, [45](#)
- eCartesianDirection
 - GridUtils.h, [47](#)
- eFluid
 - GridObj.h, [45](#)
- eHdf5SlabType
 - hdf5luma.h, [48](#)
- eIBBCloud
 - ObjectManager.h, [53](#)
- eInlet
 - GridObj.h, [45](#)
- eMaximum
 - GridUtils.h, [47](#)
- eMinMax
 - GridUtils.h, [47](#)
- eMinimum
 - GridUtils.h, [47](#)
- eObjectType
 - ObjectManager.h, [53](#)
- eOutlet
 - GridObj.h, [45](#)
- ePosX
 - hdf5luma.h, [48](#)
- ePosY
 - hdf5luma.h, [48](#)
- ePosZ
 - hdf5luma.h, [48](#)
- eProductVector
 - hdf5luma.h, [48](#)
- eRefined
 - GridObj.h, [45](#)
- eRefinedInlet
 - GridObj.h, [45](#)
- eRefinedSolid
 - GridObj.h, [45](#)
- eRefinedSymmetry
 - GridObj.h, [45](#)
- eScalar
 - hdf5luma.h, [48](#)
- eSolid
 - GridObj.h, [45](#)
- eSymmetry
 - GridObj.h, [45](#)
- eTransitionToCoarser
 - GridObj.h, [45](#)
- eTransitionToFiner
 - GridObj.h, [45](#)
- eType
 - GridObj.h, [45](#)
- eVector
 - hdf5luma.h, [48](#)
- eXDirection
 - GridUtils.h, [47](#)
- eYDirection
 - GridUtils.h, [47](#)
- eZDirection
 - GridUtils.h, [47](#)
- epsilon
 - IBMarker, [22](#)
- f_buffer_recv
 - MpiManager, [28](#)
- f_buffer_send
 - MpiManager, [28](#)
- factorial
 - GridUtils, [17](#)
- flex_rigid
 - IBBody, [21](#)
 - IBMarker, [23](#)
- flexural_rigidity
 - IBBody, [21](#)
- fluid_vel
 - IBMarker, [23](#)
- force_xyz
 - IBMarker, [23](#)
- Gen_init_globalvars.cpp, [43](#)
 - c, [43](#)
 - cs, [43](#)
 - w, [43](#)
- getCoarseIndices
 - GridUtils, [17](#)
- getFineIndices
 - GridUtils, [17](#)
- getGrid
 - GridUtils, [17](#)
- getInstance
 - MpiManager, [28](#)

- ObjectManager, 31
- getMarkerData
 - Body, 10
- getOpposite
 - GridUtils, 17
- getVoxInd
 - ObjectManager, 31
- global_dims
 - MpiManager, 28
- global_edge_ind
 - MpiManager, 29
- global_edge_pos
 - MpiManager, 29
- global_to_local
 - GridUtils, 17
- globalvars.h, 44
 - c, 44
 - cs, 44
 - w, 44
- GridObj, 11
 - ~GridObj, 13
 - BFLBody, 8
 - bc_applyBfl, 13
 - bc_applyBounceBack, 13
 - bc_applyExtrapolation, 13
 - bc_applyNrbc, 13
 - bc_applyRegularised, 13
 - bc_applySpecReflect, 13
 - bc_applyZouHe, 14
 - bc_getWallDensityForRBC, 14
 - bc_solidSiteReset, 14
 - dt, 15
 - GridObj, 13
 - GridUtils, 15
 - io_hdf5, 14
 - io_lite, 14
 - io_probeOutput, 14
 - io_restart, 14
 - io_textout, 14
 - LBM_addSubGrid, 14
 - LBM_boundary, 14
 - LBM_coalesce, 14
 - LBM_collide, 14
 - LBM_explode, 14
 - LBM_forcegrid, 14
 - LBM_init_getInletProfile, 14
 - LBM_initBoundLab, 14
 - LBM_initGrid, 14
 - LBM_initRefinedLab, 14
 - LBM_initRho, 14
 - LBM_initSolidLab, 14
 - LBM_initSubGrid, 15
 - LBM_initVelocity, 15
 - LBM_kbcCollide, 15
 - LBM_macro, 15
 - LBM_multi, 15
 - LBM_stream, 15
 - LatTyp, 15
 - level, 15
 - MpiManager, 15
 - nu, 15
 - ObjectManager, 15, 32
 - omega, 15
 - t, 15
 - timeav_mpi_overhead, 15
 - timeav_timestep, 15
 - XInd, 15
 - XPos, 15
 - YInd, 15
 - YPos, 15
 - ZInd, 15
 - ZPos, 15
- GridObj.cpp, 44
- GridObj.h, 44
 - eBCAll, 45
 - eBCBFL, 45
 - eBCInlet, 45
 - eBCInletOutlet, 45
 - eBCOutlet, 45
 - eBCSolidSymmetry, 45
 - eBCType, 45
 - eBFL, 45
 - eFluid, 45
 - eInlet, 45
 - eOutlet, 45
 - eRefined, 45
 - eRefinedInlet, 45
 - eRefinedSolid, 45
 - eRefinedSymmetry, 45
 - eSolid, 45
 - eSymmetry, 45
 - eTransitionToCoarser, 45
 - eTransitionToFiner, 45
 - eType, 45
- GridObj_init_grids.cpp, 46
- GridObj_ops_boundary.cpp, 46
- GridObj_ops_io.cpp, 46
- GridObj_ops_lbm.cpp, 46
- GridUtils, 16
 - add, 17
 - createOutputDirectory, 17
 - crossprod, 17
 - dir_reflect, 19
 - dotprod, 17
 - downToLimit, 17
 - factorial, 17
 - getCoarseIndices, 17
 - getFineIndices, 17
 - getGrid, 17
 - getOpposite, 17
 - global_to_local, 17
 - GridObj, 15
 - hasThisSubGrid, 17
 - indexToPosition, 18
 - isOffGrid, 18
 - isOnRecvLayer, 18

- isOnSenderLayer, [18](#)
- isOnThisRank, [18](#)
- isOverlapPeriodic, [18](#)
- linspace, [18](#)
- local_to_global, [18](#)
- logfile, [19](#)
- matrix_multiply, [18](#)
- onespace, [18](#)
- path_str, [19](#)
- stridedCopy, [18](#)
- subtract, [18](#)
- upToZero, [18](#)
- vecmultiply, [18](#)
- vecnorm, [18](#), [19](#)
- GridUtils.cpp, [47](#)
- GridUtils.h, [47](#)
 - eCartesianDirection, [47](#)
 - eMaximum, [47](#)
 - eMinMax, [47](#)
 - eMinimum, [47](#)
 - eXDirection, [47](#)
 - eYDirection, [47](#)
 - eZDirection, [47](#)
- Grids
 - MpiManager, [29](#)
- groupID
 - IBBody, [21](#)
- H5_BUILT_AS_DYNAMIC_LIB
 - hdf5luma.h, [48](#)
- HDF5_EXT_SZIP
 - hdf5luma.h, [48](#)
- HDF5_EXT_ZLIB
 - hdf5luma.h, [48](#)
- halo_max
 - MpiManager::phdf5_struct, [34](#)
- halo_min
 - MpiManager::phdf5_struct, [34](#)
- hasThisSubGrid
 - GridUtils, [17](#)
- hdf5_writeDataSet
 - hdf5luma.h, [49](#)
- hdf5luma.h, [48](#)
 - eHdf5SlabType, [48](#)
 - ePosX, [48](#)
 - ePosY, [48](#)
 - ePosZ, [48](#)
 - eProductVector, [48](#)
 - eScalar, [48](#)
 - eVector, [48](#)
 - H5_BUILT_AS_DYNAMIC_LIB, [48](#)
 - HDF5_EXT_SZIP, [48](#)
 - HDF5_EXT_ZLIB, [48](#)
 - hdf5_writeDataSet, [49](#)
- i
 - MarkerData, [26](#)
- i_end
 - MpiManager::phdf5_struct, [34](#)
- i_start
 - MpiManager::phdf5_struct, [34](#)
- IBBody, [19](#)
 - ~IBBody, [20](#)
 - addMarker, [20](#)
 - BCs, [21](#)
 - deformable, [21](#)
 - delta_rho, [21](#)
 - flex_rigid, [21](#)
 - flexural_rigidity, [21](#)
 - groupID, [21](#)
 - IBBody, [20](#)
 - IBMarker, [22](#)
 - makeBody, [20](#), [21](#)
 - ObjectManager, [21](#)
 - tension, [21](#)
- IBBody.cpp, [49](#)
- IBBody.h, [49](#)
- IBMarker, [21](#)
 - ~IBMarker, [22](#)
 - deltaval, [22](#)
 - desired_vel, [22](#)
 - dilation, [22](#)
 - epsilon, [22](#)
 - flex_rigid, [23](#)
 - fluid_vel, [23](#)
 - force_xyz, [23](#)
 - IBBody, [22](#)
 - IBMarker, [22](#)
 - local_area, [23](#)
 - ObjectManager, [22](#)
 - position_old, [23](#)
- IBMarker.cpp, [49](#)
- IBMarker.h, [49](#)
- IVector
 - ~IVector, [24](#)
 - IVector, [24](#)
 - operator(), [24](#)
- IVector< GenTyp >, [23](#)
- IVector.h, [50](#)
- ibm_apply
 - ObjectManager, [31](#)
- ibm_banbks
 - ObjectManager, [31](#)
- ibm_banded
 - ObjectManager, [31](#)
- ibm_bicgstab
 - ObjectManager, [31](#)
- ibm_build_body
 - ObjectManager, [31](#)
- ibm_computeforce
 - ObjectManager, [31](#)
- ibm_deltakernel
 - ObjectManager, [31](#)
- ibm_findepsilon
 - ObjectManager, [31](#)
- ibm_findsupport
 - ObjectManager, [31](#)

- ibm_initialise
 - ObjectManager, [31](#)
- ibm_interpol
 - ObjectManager, [31](#)
- ibm_jacowire
 - ObjectManager, [31](#)
- ibm_move_bodies
 - ObjectManager, [32](#)
- ibm_position_update
 - ObjectManager, [32](#)
- ibm_position_update_grp
 - ObjectManager, [32](#)
- ibm_spread
 - ObjectManager, [32](#)
- ID
 - MarkerData, [26](#)
- indexToPosition
 - GridUtils, [18](#)
- io_hdf5
 - GridObj, [14](#)
- io_lite
 - GridObj, [14](#)
- io_probeOutput
 - GridObj, [14](#)
- io_readInCloud
 - ObjectManager, [32](#)
- io_restart
 - GridObj, [14](#)
 - ObjectManager, [32](#)
- io_textout
 - GridObj, [14](#)
- io_vtk_IBwriter
 - ObjectManager, [32](#)
- io_write_body_pos
 - ObjectManager, [32](#)
- io_write_lift_drag
 - ObjectManager, [32](#)
- is_nan
 - stdafx.h, [55](#)
- isInVoxel
 - Body, [10](#)
- isOffGrid
 - GridUtils, [18](#)
- isOnRecvLayer
 - GridUtils, [18](#)
- isOnSenderLayer
 - GridUtils, [18](#)
- isOnThisRank
 - GridUtils, [18](#)
- isOverlapPeriodic
 - GridUtils, [18](#)
- isVoxelMarkerVoxel
 - Body, [10](#)
- j
 - MarkerData, [26](#)
- j_end
 - MpiManager::phdf5_struct, [34](#)
- j_start
- MpiManager::phdf5_struct, [34](#)
- k
 - MarkerData, [26](#)
- k_end
 - MpiManager::phdf5_struct, [34](#)
- k_start
 - MpiManager::phdf5_struct, [34](#)
- L_BUILD_FOR_MPI
 - definitions.h, [39](#)
- L_CHEAP_NEAREST_NODE_DETECTION
 - definitions.h, [39](#)
- L_FREESTREAM_TUNNEL
 - definitions.h, [39](#)
- L_HDF5_OUTPUT
 - definitions.h, [39](#)
- L_INLET_ON
 - definitions.h, [40](#)
- L_MPI_dir
 - definitions.h, [41](#)
- L_NumLev
 - definitions.h, [41](#)
- L_NumReg
 - definitions.h, [41](#)
- L_OUTLET_ON
 - definitions.h, [41](#)
- L_PI
 - definitions.h, [41](#)
- L_Re
 - definitions.h, [41](#)
- L_SOLID_BLOCK_ON
 - definitions.h, [41](#)
- L_Timesteps
 - definitions.h, [42](#)
- L_USE_KBC_COLLISION
 - definitions.h, [42](#)
- L_Xcores
 - definitions.h, [42](#)
- L_Ycores
 - definitions.h, [42](#)
- L_Zcores
 - definitions.h, [42](#)
- L_a_x
 - definitions.h, [38](#)
- L_a_y
 - definitions.h, [38](#)
- L_a_z
 - definitions.h, [38](#)
- L_b_x
 - definitions.h, [38](#)
- L_b_y
 - definitions.h, [38](#)
- L_b_z
 - definitions.h, [38](#)
- L_bfl_length
 - definitions.h, [38](#)
- L_bfl_length_ref
 - definitions.h, [38](#)

- L_bfl_on_grid_lev
 - definitions.h, [38](#)
- L_bfl_on_grid_reg
 - definitions.h, [39](#)
- L_bfl_scale_direction
 - definitions.h, [39](#)
- L_block_on_grid_lev
 - definitions.h, [39](#)
- L_block_on_grid_reg
 - definitions.h, [39](#)
- L_block_x_max
 - definitions.h, [39](#)
- L_block_x_min
 - definitions.h, [39](#)
- L_block_y_max
 - definitions.h, [39](#)
- L_block_y_min
 - definitions.h, [39](#)
- L_block_z_max
 - definitions.h, [39](#)
- L_block_z_min
 - definitions.h, [39](#)
- L_centre_bfl_z
 - definitions.h, [39](#)
- L_centre_obb_z
 - definitions.h, [39](#)
- L_centre_object_z
 - definitions.h, [39](#)
- L_dims
 - definitions.h, [39](#)
- L_end_BC
 - definitions.h, [39](#)
- L_grav_direction
 - definitions.h, [39](#)
- L_grav_force
 - definitions.h, [39](#)
- L_obb_EI
 - definitions.h, [40](#)
- L_obb_angle_horz
 - definitions.h, [39](#)
- L_obb_angle_vert
 - definitions.h, [39](#)
- L_obb_d
 - definitions.h, [40](#)
- L_obb_deform
 - definitions.h, [40](#)
- L_obb_delta_rho
 - definitions.h, [40](#)
- L_obb_filament_length
 - definitions.h, [40](#)
- L_obb_filament_start_x
 - definitions.h, [40](#)
- L_obb_filament_start_y
 - definitions.h, [40](#)
- L_obb_filament_start_z
 - definitions.h, [40](#)
- L_obb_flex_rigid
 - definitions.h, [40](#)
- L_obb_l
 - definitions.h, [40](#)
- L_obb_length
 - definitions.h, [40](#)
- L_obb_length_ref
 - definitions.h, [40](#)
- L_obb_on_grid_lev
 - definitions.h, [40](#)
- L_obb_on_grid_reg
 - definitions.h, [40](#)
- L_obb_r
 - definitions.h, [40](#)
- L_obb_scale_direction
 - definitions.h, [40](#)
- L_obb_w
 - definitions.h, [40](#)
- L_obb_x
 - definitions.h, [40](#)
- L_obb_y
 - definitions.h, [40](#)
- L_obb_z
 - definitions.h, [40](#)
- L_K
 - definitions.h, [40](#)
- L_M
 - definitions.h, [40](#)
- L_N
 - definitions.h, [41](#)
- L_nVels
 - definitions.h, [41](#)
- L_num_markers
 - definitions.h, [41](#)
- L_object_length
 - definitions.h, [41](#)
- L_object_length_ref
 - definitions.h, [41](#)
- L_object_on_grid_lev
 - definitions.h, [41](#)
- L_object_on_grid_reg
 - definitions.h, [41](#)
- L_object_scale_direction
 - definitions.h, [41](#)
- L_out_every
 - definitions.h, [41](#)
- L_out_every_probe
 - definitions.h, [41](#)
- L_output_precision
 - definitions.h, [41](#)
- L_restart_out_every
 - definitions.h, [41](#)
- L_rho_in
 - definitions.h, [41](#)
- L_start_BC
 - definitions.h, [41](#)
- L_start_bfl_x
 - definitions.h, [41](#)
- L_start_bfl_y
 - definitions.h, [41](#)

- L_start_ibb_x
 - definitions.h, [42](#)
- L_start_ibb_y
 - definitions.h, [42](#)
- L_start_object_x
 - definitions.h, [42](#)
- L_start_object_y
 - definitions.h, [42](#)
- L_u_0x
 - definitions.h, [42](#)
- L_u_0y
 - definitions.h, [42](#)
- L_u_0z
 - definitions.h, [42](#)
- L_u_max
 - definitions.h, [42](#)
- L_u_ref
 - definitions.h, [42](#)
- L_wall_thickness
 - definitions.h, [42](#)
- LBM_addSubGrid
 - GridObj, [14](#)
- LBM_boundary
 - GridObj, [14](#)
- LBM_coalesce
 - GridObj, [14](#)
- LBM_collide
 - GridObj, [14](#)
- LBM_explode
 - GridObj, [14](#)
- LBM_forcegrid
 - GridObj, [14](#)
- LBM_init_getInletProfile
 - GridObj, [14](#)
- LBM_initBoundLab
 - GridObj, [14](#)
- LBM_initGrid
 - GridObj, [14](#)
- LBM_initRefinedLab
 - GridObj, [14](#)
- LBM_initRho
 - GridObj, [14](#)
- LBM_initSolidLab
 - GridObj, [14](#)
- LBM_initSubGrid
 - GridObj, [15](#)
- LBM_initVelocity
 - GridObj, [15](#)
- LBM_kbcCollide
 - GridObj, [15](#)
- LBM_macro
 - GridObj, [15](#)
- LBM_multi
 - GridObj, [15](#)
- LBM_stream
 - GridObj, [15](#)
- LUMA_FAILED
 - stdafx.h, [55](#)
- LUMA_VERSION
 - definitions.h, [42](#)
- LatTyp
 - GridObj, [15](#)
- level
 - GridObj, [15](#)
 - MpiManager::buffer_struct, [11](#)
 - MpiManager::phdf5_struct, [34](#)
- linspace
 - GridUtils, [18](#)
- local_area
 - IBMarker, [23](#)
- local_size
 - MpiManager, [29](#)
- local_to_global
 - GridUtils, [18](#)
- logfile
 - GridUtils, [19](#)
- logout
 - MpiManager, [29](#)
- MPI_cartlab
 - MpiManager, [29](#)
- MPI_coords
 - MpiManager, [29](#)
- MPI_dims
 - MpiManager, [29](#)
- main
 - main_lbm.cpp, [50](#)
- main_lbm.cpp, [50](#)
- main, [50](#)
- makeBody
 - IBBody, [20](#), [21](#)
- Marker, [25](#)
 - ~Marker, [25](#)
 - Marker, [25](#)
 - position, [25](#)
 - supp_i, [25](#)
 - supp_j, [25](#)
 - supp_k, [25](#)
 - support_rank, [25](#)
- Marker.h, [50](#)
- markerAdder
 - Body, [10](#)
- MarkerData, [26](#)
 - ~MarkerData, [26](#)
 - i, [26](#)
 - ID, [26](#)
 - j, [26](#)
 - k, [26](#)
 - MarkerData, [26](#)
 - x, [26](#)
 - y, [26](#)
 - z, [26](#)
- markers
 - Body, [10](#)
- matrix_multiply
 - GridUtils, [18](#)
- mpi_buffer_pack

- MpiManager, 28
- Mpi_buffer_pack.cpp, 50
- mpi_buffer_size
 - MpiManager, 28
- mpi_buffer_size_recv
 - MpiManager, 28
- Mpi_buffer_size_recv.cpp, 51
- mpi_buffer_size_send
 - MpiManager, 28
- Mpi_buffer_size_send.cpp, 51
- mpi_buffer_unpack
 - MpiManager, 28
- Mpi_buffer_unpk.cpp, 51
- mpi_buildCommunicators
 - MpiManager, 28
- mpi_communicate
 - MpiManager, 28
- mpi_getOpposite
 - MpiManager, 28
- mpi_gridbuild
 - MpiManager, 28
- mpi_init
 - MpiManager, 28
- mpi_writeout_buf
 - MpiManager, 28
- MpiManager, 27
 - buffer_recv_info, 28
 - buffer_send_info, 28
 - destroyInstance, 28
 - f_buffer_recv, 28
 - f_buffer_send, 28
 - getInstance, 28
 - global_dims, 28
 - global_edge_ind, 29
 - global_edge_pos, 29
 - GridObj, 15
 - Grids, 29
 - local_size, 29
 - logout, 29
 - MPI_cartlab, 29
 - MPI_coords, 29
 - MPI_dims, 29
 - mpi_buffer_pack, 28
 - mpi_buffer_size, 28
 - mpi_buffer_size_recv, 28
 - mpi_buffer_size_send, 28
 - mpi_buffer_unpack, 28
 - mpi_buildCommunicators, 28
 - mpi_communicate, 28
 - mpi_getOpposite, 28
 - mpi_gridbuild, 28
 - mpi_init, 28
 - mpi_writeout_buf, 28
 - my_rank, 29
 - neighbour_coords, 29
 - neighbour_rank, 29
 - num_ranks, 29
 - p_data, 29
 - recv_layer_pos, 29
 - recv_stat, 29
 - send_requests, 29
 - send_stat, 29
 - sender_layer_pos, 29
 - subGrid_comm, 29
 - world_comm, 29
- MpiManager.cpp, 51
- MpiManager.h, 51
 - range_i_left, 52
 - range_i_right, 52
 - range_j_down, 52
 - range_j_up, 52
 - range_k_back, 52
 - range_k_front, 52
- MpiManager::buffer_struct, 10
 - level, 11
 - region, 11
 - size, 11
- MpiManager::layer_edges, 24
 - X, 24
 - Y, 24
 - Z, 24
- MpiManager::phdf5_struct, 33
 - halo_max, 34
 - halo_min, 34
 - i_end, 34
 - i_start, 34
 - j_end, 34
 - j_start, 34
 - k_end, 34
 - k_start, 34
 - level, 34
 - region, 34
 - writable_data_count, 34
- my_rank
 - MpiManager, 29
- nProbes
 - definitions.h, 42
- neighbour_coords
 - MpiManager, 29
- neighbour_rank
 - MpiManager, 29
- nu
 - GridObj, 15
- num_ranks
 - MpiManager, 29
- ObjectManager, 30
 - bfl_build_body, 31
 - destroyInstance, 31
 - getInstance, 31
 - getVoxInd, 31
 - GridObj, 15, 32
 - IBBody, 21
 - IBMarker, 22
 - ibm_apply, 31
 - ibm_banbks, 31

- ibm_banded, 31
- ibm_bicgstab, 31
- ibm_build_body, 31
- ibm_computeforce, 31
- ibm_deltakernel, 31
- ibm_findepsilon, 31
- ibm_findsupport, 31
- ibm_initialise, 31
- ibm_interpol, 31
- ibm_jacowire, 31
- ibm_move_bodies, 32
- ibm_position_update, 32
- ibm_position_update_grp, 32
- ibm_spread, 32
- io_readInCloud, 32
- io_restart, 32
- io_vtk_IBwriter, 32
- io_write_body_pos, 32
- io_write_lift_drag, 32
- ObjectManager.cpp, 52
- ObjectManager.h, 52
 - eBBBCloud, 53
 - eBFLCloud, 53
 - eIBBCloud, 53
 - eObjectType, 53
- ObjectManager_init_bflbody.cpp, 53
- ObjectManager_init_ibmbody.cpp, 53
- ObjectManager_ops_ibm.cpp, 53
- ObjectManager_ops_ibmflex.cpp, 53
 - SWAP, 54
 - TINY, 54
- ObjectManager_ops_io.cpp, 54
- omega
 - GridObj, 15
- onespace
 - GridUtils, 18
- operator()
 - IVector, 24
- p_data
 - MpiManager, 29
- PCpts, 32
 - ~PCpts, 33
 - PCpts, 33
 - x, 33
 - y, 33
 - z, 33
- PCpts.h, 54
- path_str
 - GridUtils, 19
- position
 - Marker, 25
- position_old
 - IBMarker, 23
- Q
 - BFLBody, 8
- range_i_left
 - MpiManager.h, 52
- range_i_right
 - MpiManager.h, 52
- range_j_down
 - MpiManager.h, 52
- range_j_up
 - MpiManager.h, 52
- range_k_back
 - MpiManager.h, 52
- range_k_front
 - MpiManager.h, 52
- recv_layer_pos
 - MpiManager, 29
- recv_stat
 - MpiManager, 29
- RefXend
 - definitions.h, 42
- RefXstart
 - definitions.h, 42
- RefYend
 - definitions.h, 42
- RefYstart
 - definitions.h, 42
- RefZend
 - definitions.h, 42
- RefZstart
 - definitions.h, 43
- region
 - MpiManager::buffer_struct, 11
 - MpiManager::phdf5_struct, 34
- SWAP
 - ObjectManager_ops_ibmflex.cpp, 54
- send_requests
 - MpiManager, 29
- send_stat
 - MpiManager, 29
- sender_layer_pos
 - MpiManager, 29
- size
 - MpiManager::buffer_struct, 11
- spacing
 - Body, 10
- stdafx.cpp, 54
- stdafx.h, 54
 - is_nan, 55
 - LUMA_FAILED, 55
- stridedCopy
 - GridUtils, 18
- subGrid_comm
 - MpiManager, 29
- subtract
 - GridUtils, 18
- supp_i
 - Marker, 25
- supp_j
 - Marker, 25
- supp_k
 - Marker, 25

support_rank
 Marker, [25](#)

t
 GridObj, [15](#)

TINY
 ObjectManager_ops_ibmflex.cpp, [54](#)

targetver.h, [55](#)

tension
 IBBody, [21](#)

timeav_mpi_overhead
 GridObj, [15](#)

timeav_timestep
 GridObj, [15](#)

upToZero
 GridUtils, [18](#)

vecmultiply
 GridUtils, [18](#)

vecnorm
 GridUtils, [18](#), [19](#)

w
 Gen_init_globalvars.cpp, [43](#)
 globalvars.h, [44](#)

world_comm
 MpiManager, [29](#)

writable_data_count
 MpiManager::phdf5_struct, [34](#)

X
 MpiManager::layer_edges, [24](#)

x
 MarkerData, [26](#)
 PCpts, [33](#)

XInd
 GridObj, [15](#)

XPos
 GridObj, [15](#)

xProbeLims
 definitions.h, [43](#)

Y
 MpiManager::layer_edges, [24](#)

y
 MarkerData, [26](#)
 PCpts, [33](#)

YInd
 GridObj, [15](#)

YPos
 GridObj, [15](#)

yProbeLims
 definitions.h, [43](#)

Z
 MpiManager::layer_edges, [24](#)

z
 MarkerData, [26](#)
 PCpts, [33](#)

ZInd
 GridObj, [15](#)

ZPos
 GridObj, [15](#)

zProbeLims
 definitions.h, [43](#)