

Qualia

0.2

Generated by Doxygen 1.8.3.1

Thu Jul 4 2013 13:44:02

Contents

1	Main Page	1
2	Hierarchical Index	3
2.1	Class Hierarchy	3
3	Class Index	5
3.1	Class List	5
4	File Index	7
4.1	File List	7
5	Class Documentation	11
5.1	Action Class Reference	11
5.1.1	Detailed Description	12
5.1.2	Constructor & Destructor Documentation	12
5.1.2.1	Action	12
5.1.2.2	~Action	12
5.1.3	Member Function Documentation	12
5.1.3.1	conflated	12
5.1.3.2	copyFrom	13
5.1.3.3	dim	13
5.1.3.4	hasNext	13
5.1.3.5	loadData	13
5.1.3.6	nActions	13
5.1.3.7	nConflated	13
5.1.3.8	next	13
5.1.3.9	operator[]	13
5.1.3.10	reset	13
5.1.3.11	saveData	13
5.1.3.12	setConflated	13
5.1.3.13	undefined	13
5.1.4	Member Data Documentation	14
5.1.4.1	_undefined	14

5.1.4.2	actions	14
5.1.4.3	properties	14
5.2	ActionProperties Class Reference	14
5.2.1	Detailed Description	14
5.2.2	Constructor & Destructor Documentation	15
5.2.2.1	ActionProperties	15
5.2.2.2	~ActionProperties	15
5.2.3	Member Function Documentation	15
5.2.3.1	dim	15
5.2.3.2	equals	15
5.2.3.3	nActions	15
5.2.3.4	nConflated	15
5.2.3.5	random	15
5.2.4	Member Data Documentation	15
5.2.4.1	_dim	15
5.2.4.2	_nActions	15
5.2.4.3	_nConflated	15
5.3	Agent Class Reference	15
5.3.1	Detailed Description	16
5.3.2	Constructor & Destructor Documentation	16
5.3.2.1	~Agent	16
5.3.3	Member Function Documentation	16
5.3.3.1	end	16
5.3.3.2	init	17
5.3.3.3	load	17
5.3.3.4	save	17
5.3.3.5	start	17
5.3.3.6	step	17
5.4	Alloc Class Reference	17
5.4.1	Detailed Description	18
5.4.2	Member Function Documentation	18
5.4.2.1	free	18
5.4.2.2	init	18
5.4.2.3	instance	18
5.4.2.4	malloc	18
5.4.2.5	realloc	18
5.4.3	Member Data Documentation	18
5.4.3.1	inst	18
5.5	Allocator Class Reference	18
5.5.1	Detailed Description	19

5.5.2	Constructor & Destructor Documentation	19
5.5.2.1	~Allocator	19
5.5.3	Member Function Documentation	19
5.5.3.1	free	19
5.5.3.2	malloc	19
5.5.3.3	realloc	19
5.6	BatchRLTrainer Class Reference	20
5.6.1	Detailed Description	21
5.6.2	Constructor & Destructor Documentation	21
5.6.2.1	BatchRLTrainer	21
5.6.2.2	~BatchRLTrainer	21
5.6.3	Member Function Documentation	21
5.6.3.1	_doTrainEpisode	21
5.6.3.2	init	21
5.6.4	Member Data Documentation	22
5.6.4.1	_lastObservation	22
5.6.4.2	_lastOrNextAction	22
5.6.4.3	_observation	22
5.6.4.4	gamma	22
5.6.4.5	maxExamples	22
5.6.4.6	maxTrainingIterationsPerEpisode	22
5.6.4.7	targets	22
5.7	BinaryChromosome Class Reference	22
5.7.1	Constructor & Destructor Documentation	24
5.7.1.1	BinaryChromosome	24
5.7.1.2	~BinaryChromosome	24
5.7.2	Member Function Documentation	24
5.7.2.1	_codeCrossoverOnePoint	24
5.7.2.2	_codeCrossoverTwoPoint	24
5.7.2.3	copyFrom	24
5.7.2.4	crossoverOnePoint	24
5.7.2.5	crossoverTwoPoint	24
5.7.2.6	equals	24
5.7.2.7	getGeneValue	24
5.7.2.8	init	24
5.7.2.9	initializeRandom	24
5.7.2.10	mutate	24
5.7.2.11	mutateFlip	24
5.7.2.12	setGeneValue	24
5.7.3	Member Data Documentation	25

5.7.3.1	code	25
5.7.3.2	info	25
5.8	BinaryChromosomeProperties Class Reference	25
5.8.1	Constructor & Destructor Documentation	26
5.8.1.1	BinaryChromosomeProperties	26
5.8.2	Member Function Documentation	26
5.8.2.1	bitSize	26
5.8.2.2	byteSize	26
5.8.2.3	equals	26
5.8.2.4	geneSize	26
5.8.2.5	getStartBitPosition	26
5.8.2.6	nGenes	26
5.8.3	Member Data Documentation	26
5.8.3.1	_geneSizes	26
5.8.3.2	_nGenes	26
5.8.3.3	evaluator	27
5.8.3.4	initializer	27
5.8.3.5	mutator	27
5.9	BoolCmdOption Class Reference	27
5.9.1	Detailed Description	28
5.9.2	Constructor & Destructor Documentation	28
5.9.2.1	BoolCmdOption	28
5.9.2.2	~BoolCmdOption	28
5.9.3	Member Function Documentation	28
5.9.3.1	initValue	28
5.9.3.2	load	28
5.9.3.3	read	28
5.9.3.4	save	28
5.9.4	Member Data Documentation	29
5.9.4.1	init_value	29
5.9.4.2	ptr	29
5.10	Chromosome Class Reference	29
5.10.1	Constructor & Destructor Documentation	29
5.10.1.1	Chromosome	29
5.10.1.2	~Chromosome	29
5.10.2	Member Function Documentation	29
5.10.2.1	copyFrom	29
5.10.2.2	equals	30
5.10.2.3	init	30
5.10.2.4	mutate	30

5.11 CmdLine Class Reference	30
5.11.1 Detailed Description	31
5.11.2 Constructor & Destructor Documentation	31
5.11.2.1 CmdLine	31
5.11.2.2 ~CmdLine	31
5.11.3 Member Function Documentation	31
5.11.3.1 addBCmdArg	32
5.11.3.2 addBCmdOption	32
5.11.3.3 addCmdOption	32
5.11.3.4 addICmdArg	32
5.11.3.5 addICmdOption	32
5.11.3.6 addMasterSwitch	32
5.11.3.7 addRCmdArg	32
5.11.3.8 addRCmdOption	32
5.11.3.9 addSCmdArg	32
5.11.3.10 addSCmdOption	32
5.11.3.11 addText	32
5.11.3.12 getPath	32
5.11.3.13 getXFile	32
5.11.3.14 help	33
5.11.3.15 info	33
5.11.3.16 load	33
5.11.3.17 read	33
5.11.3.18 save	33
5.11.3.19 setWorkingDirectory	33
5.11.3.20 writeLog	33
5.11.4 Member Data Documentation	33
5.11.4.1 argc	33
5.11.4.2 argv	33
5.11.4.3 associated_files	33
5.11.4.4 cmd_options	33
5.11.4.5 master_switch	33
5.11.4.6 n_associated_files	33
5.11.4.7 n_cmd_options	33
5.11.4.8 n_master_switches	33
5.11.4.9 program_name	33
5.11.4.10 text_info	33
5.11.4.11 working_directory	34
5.11.4.12 write_log	34
5.12 CmdOption Class Reference	34

5.12.1 Detailed Description	35
5.12.2 Constructor & Destructor Documentation	35
5.12.2.1 CmdOption	35
5.12.2.2 ~CmdOption	35
5.12.3 Member Function Documentation	35
5.12.3.1 initValue	35
5.12.3.2 isArgument	35
5.12.3.3 isCurrent	35
5.12.3.4 isMasterSwitch	35
5.12.3.5 isOption	35
5.12.3.6 isText	35
5.12.3.7 load	36
5.12.3.8 printValue	36
5.12.3.9 read	36
5.12.3.10 save	36
5.12.4 Member Data Documentation	36
5.12.4.1 help	36
5.12.4.2 is_argument	36
5.12.4.3 is_master_switch	36
5.12.4.4 is_option	36
5.12.4.5 is_setted	36
5.12.4.6 is_text	36
5.12.4.7 name	36
5.12.4.8 needsSave	36
5.12.4.9 type_name	36
5.13 DataSet Class Reference	37
5.13.1 Detailed Description	37
5.13.2 Constructor & Destructor Documentation	38
5.13.2.1 DataSet	38
5.13.2.2 ~DataSet	38
5.13.3 Member Function Documentation	38
5.13.3.1 dim	38
5.13.3.2 init	38
5.13.3.3 nExamples	38
5.13.3.4 reset	38
5.13.3.5 setExample	38
5.13.4 Member Data Documentation	38
5.13.4.1 _dim	38
5.13.4.2 _nExamples	38
5.13.4.3 example	38

5.14 DataSetTrainer Class Reference	39
5.14.1 Detailed Description	40
5.14.2 Constructor & Destructor Documentation	40
5.14.2.1 DataSetTrainer	40
5.14.2.2 ~DataSetTrainer	40
5.14.3 Member Function Documentation	40
5.14.3.1 _doTrainEpisode	40
5.14.3.2 stop	40
5.14.3.3 train	40
5.14.3.4 trainEpisode	40
5.15 DiskXFile Class Reference	41
5.15.1 Detailed Description	42
5.15.2 Constructor & Destructor Documentation	42
5.15.2.1 DiskXFile	42
5.15.2.2 DiskXFile	43
5.15.2.3 ~DiskXFile	43
5.15.3 Member Function Documentation	43
5.15.3.1 eof	43
5.15.3.2 flush	43
5.15.3.3 gets	43
5.15.3.4 isBigEndianProcessor	43
5.15.3.5 isLittleEndianProcessor	43
5.15.3.6 isNativeMode	43
5.15.3.7 printf	43
5.15.3.8 read	43
5.15.3.9 reverseMemory	43
5.15.3.10 rewind	43
5.15.3.11 scanf	44
5.15.3.12 seek	44
5.15.3.13 setBigEndianMode	44
5.15.3.14 setLittleEndianMode	44
5.15.3.15 setNativeMode	44
5.15.3.16 tell	44
5.15.3.17 write	44
5.15.4 Member Data Documentation	44
5.15.4.1 buffer_block	44
5.15.4.2 buffer_block_size	44
5.15.4.3 file	44
5.15.4.4 is_native_mode	44
5.15.4.5 is_opened	44

5.15.4.6	its_a_pipe	44
5.16	DiskXFileDataSet Class Reference	45
5.16.1	Constructor & Destructor Documentation	46
5.16.1.1	DiskXFileDataSet	46
5.16.1.2	~DiskXFileDataSet	46
5.16.2	Member Function Documentation	46
5.16.2.1	init	46
5.16.2.2	reset	46
5.16.2.3	setExample	46
5.16.3	Member Data Documentation	46
5.16.3.1	_diskFile	46
5.16.3.2	ascii	46
5.17	Environment Class Reference	47
5.17.1	Detailed Description	47
5.17.2	Constructor & Destructor Documentation	47
5.17.2.1	~Environment	47
5.17.3	Member Function Documentation	47
5.17.3.1	init	47
5.17.3.2	start	47
5.17.3.3	step	48
5.18	FileExportEnvironment Class Reference	48
5.18.1	Detailed Description	49
5.18.2	Constructor & Destructor Documentation	49
5.18.2.1	FileExportEnvironment	49
5.18.2.2	~FileExportEnvironment	49
5.18.3	Member Function Documentation	49
5.18.3.1	init	49
5.18.3.2	start	49
5.18.3.3	step	49
5.18.4	Member Data Documentation	50
5.18.4.1	actionDim	50
5.18.4.2	env	50
5.18.4.3	file	50
5.18.4.4	observationDim	50
5.19	Function Class Reference	50
5.19.1	Detailed Description	51
5.19.2	Constructor & Destructor Documentation	51
5.19.2.1	Function	51
5.19.2.2	~Function	51
5.19.3	Member Function Documentation	51

5.19.3.1	getOutput	51
5.19.3.2	getOutputs	51
5.19.3.3	init	51
5.19.3.4	load	51
5.19.3.5	nInputs	51
5.19.3.6	nOutputs	52
5.19.3.7	propagate	52
5.19.3.8	save	52
5.19.3.9	setInput	52
5.19.3.10	setInputs	52
5.20	GradientFunction Class Reference	52
5.20.1	Detailed Description	54
5.20.2	Constructor & Destructor Documentation	54
5.20.2.1	GradientFunction	54
5.20.2.2	~GradientFunction	54
5.20.3	Member Function Documentation	54
5.20.3.1	backpropagate	54
5.20.3.2	clearDelta	54
5.20.3.3	load	54
5.20.3.4	nParams	54
5.20.3.5	save	54
5.20.3.6	update	55
5.20.4	Member Data Documentation	55
5.20.4.1	dWeights	55
5.20.4.2	weights	55
5.21	IntCmdOption Class Reference	55
5.21.1	Detailed Description	56
5.21.2	Constructor & Destructor Documentation	56
5.21.2.1	IntCmdOption	56
5.21.2.2	~IntCmdOption	56
5.21.3	Member Function Documentation	57
5.21.3.1	initValue	57
5.21.3.2	load	57
5.21.3.3	printValue	57
5.21.3.4	read	57
5.21.3.5	save	57
5.21.4	Member Data Documentation	57
5.21.4.1	init_value	57
5.21.4.2	ptr	57
5.22	NeuralNetwork::Layer Struct Reference	57

5.22.1	Detailed Description	58
5.22.2	Member Data Documentation	58
5.22.2.1	dWeight	58
5.22.2.2	error	58
5.22.2.3	linear	58
5.22.2.4	n	58
5.22.2.5	output	58
5.22.2.6	weight	58
5.23	MapperBasicEnvironment Class Reference	58
5.23.1	Constructor & Destructor Documentation	59
5.23.1.1	MapperBasicEnvironment	59
5.23.1.2	~MapperBasicEnvironment	59
5.23.2	Member Function Documentation	59
5.23.2.1	addSignals	59
5.23.2.2	readInputs	59
5.23.2.3	writeOutputs	60
5.23.3	Member Data Documentation	60
5.23.3.1	actionDim	60
5.23.3.2	currentObservation	60
5.24	MapperConnector Class Reference	60
5.24.1	Member Typedef Documentation	61
5.24.1.1	SignalDataMap	61
5.24.2	Constructor & Destructor Documentation	61
5.24.2.1	MapperConnector	61
5.24.2.2	~MapperConnector	61
5.24.3	Member Function Documentation	61
5.24.3.1	addInput	61
5.24.3.2	addOutput	61
5.24.3.3	createConnections	61
5.24.3.4	devDbCallback	61
5.24.3.5	init	61
5.24.3.6	linkDbCallback	61
5.24.3.7	logout	61
5.24.3.8	readInput	61
5.24.3.9	readInput	61
5.24.3.10	sendAllOutputs	61
5.24.3.11	updateInput	61
5.24.3.12	waitForBlockingInputs	61
5.24.3.13	writeOutput	62
5.24.3.14	writeOutput	62

5.24.4	Member Data Documentation	62
5.24.4.1	admin	62
5.24.4.2	autoConnect	62
5.24.4.3	db	62
5.24.4.4	dev	62
5.24.4.5	deviceName	62
5.24.4.6	initialPort	62
5.24.4.7	inputData	62
5.24.4.8	mon	62
5.24.4.9	nLinked	62
5.24.4.10	outputData	62
5.24.4.11	peerDeviceName	62
5.25	MapperEnvironment Class Reference	62
5.25.1	Constructor & Destructor Documentation	64
5.25.1.1	MapperEnvironment	64
5.25.1.2	~MapperEnvironment	64
5.25.2	Member Function Documentation	64
5.25.2.1	addSignals	64
5.25.2.2	init	64
5.25.2.3	readInputs	64
5.25.2.4	start	64
5.25.2.5	step	64
5.25.2.6	writeOutputs	64
5.25.3	Member Data Documentation	64
5.25.3.1	connector	64
5.26	MemoryDataSet Class Reference	65
5.26.1	Detailed Description	65
5.26.2	Constructor & Destructor Documentation	66
5.26.2.1	MemoryDataSet	66
5.26.2.2	~MemoryDataSet	66
5.26.3	Member Function Documentation	66
5.26.3.1	setExample	66
5.26.4	Member Data Documentation	66
5.26.4.1	data	66
5.27	MovingAverage Class Reference	66
5.27.1	Constructor & Destructor Documentation	66
5.27.1.1	MovingAverage	66
5.27.2	Member Function Documentation	67
5.27.2.1	get	67
5.27.2.2	reset	67

5.27.2.3	update	67
5.27.3	Member Data Documentation	67
5.27.3.1	_alpha	67
5.27.3.2	_value	67
5.28	MultiAgent Class Reference	67
5.28.1	Constructor & Destructor Documentation	68
5.28.1.1	MultiAgent	68
5.28.1.2	~MultiAgent	69
5.28.2	Member Function Documentation	69
5.28.2.1	appendAgentAction	69
5.28.2.2	combineAgentActions	69
5.28.2.3	end	69
5.28.2.4	extractAgentObservation	69
5.28.2.5	init	69
5.28.2.6	start	69
5.28.2.7	step	69
5.28.3	Member Data Documentation	69
5.28.3.1	agents	69
5.28.3.2	nAgents	70
5.28.3.3	ownsAgents	70
5.29	MultiDataSet Class Reference	70
5.29.1	Detailed Description	71
5.29.2	Constructor & Destructor Documentation	71
5.29.2.1	MultiDataSet	71
5.29.2.2	~MultiDataSet	71
5.29.3	Member Function Documentation	71
5.29.3.1	init	71
5.29.3.2	reset	71
5.29.3.3	setExample	71
5.29.4	Member Data Documentation	71
5.29.4.1	_dataSets	71
5.29.4.2	_nDataSets	71
5.30	NeuralNetwork Class Reference	72
5.30.1	Constructor & Destructor Documentation	74
5.30.1.1	NeuralNetwork	74
5.30.1.2	~NeuralNetwork	74
5.30.2	Member Function Documentation	74
5.30.2.1	_allocateLayer	74
5.30.2.2	_backpropagateLayer	74
5.30.2.3	_deallocate	74

5.30.2.4	<code>_deallocateLayer</code>	74
5.30.2.5	<code>_propagateLayer</code>	74
5.30.2.6	<code>backpropagate</code>	74
5.30.2.7	<code>getCurrentLearningRate</code>	74
5.30.2.8	<code>getOutput</code>	74
5.30.2.9	<code>getOutputs</code>	74
5.30.2.10	<code>init</code>	74
5.30.2.11	<code>load</code>	74
5.30.2.12	<code>nHidden</code>	75
5.30.2.13	<code>nInputs</code>	75
5.30.2.14	<code>nOutputs</code>	75
5.30.2.15	<code>nParams</code>	75
5.30.2.16	<code>propagate</code>	75
5.30.2.17	<code>save</code>	75
5.30.2.18	<code>setInput</code>	75
5.30.2.19	<code>setInputs</code>	75
5.30.2.20	<code>update</code>	75
5.30.3	Member Data Documentation	76
5.30.3.1	<code>_learningRateDiv</code>	76
5.30.3.2	<code>_nParams</code>	76
5.30.3.3	<code>decreaseConstant</code>	76
5.30.3.4	<code>hiddenLayer</code>	76
5.30.3.5	<code>inputLayer</code>	76
5.30.3.6	<code>learningRate</code>	76
5.30.3.7	<code>outputLayer</code>	76
5.30.3.8	<code>weightDecay</code>	76
5.31	Observation Class Reference	76
5.31.1	Detailed Description	77
5.31.2	Constructor & Destructor Documentation	78
5.31.2.1	<code>Observation</code>	78
5.31.2.2	<code>~Observation</code>	78
5.31.3	Member Function Documentation	78
5.31.3.1	<code>copyFrom</code>	78
5.31.3.2	<code>dim</code>	78
5.31.3.3	<code>loadData</code>	78
5.31.3.4	<code>operator[]</code>	78
5.31.3.5	<code>saveData</code>	78
5.31.4	Member Data Documentation	78
5.31.4.1	<code>_dim</code>	78
5.31.4.2	<code>observations</code>	78

5.31.4.3	terminal	78
5.32	ObservationAction Struct Reference	79
5.32.1	Member Data Documentation	79
5.32.1.1	action	79
5.32.1.2	observation	79
5.33	OscAgent Class Reference	79
5.33.1	Constructor & Destructor Documentation	81
5.33.1.1	OscAgent	81
5.33.1.2	~OscAgent	81
5.33.2	Member Function Documentation	81
5.33.2.1	getPath	81
5.33.2.2	handlerInit	81
5.33.2.3	handlerStartStep	81
5.33.2.4	init	81
5.33.2.5	readAction	81
5.33.2.6	start	81
5.33.2.7	step	81
5.33.3	Member Data Documentation	82
5.33.3.1	actionBuffer	82
5.33.3.2	actionDim	82
5.33.3.3	actionTypes	82
5.33.3.4	id	82
5.33.3.5	locked	82
5.33.3.6	observationBufferDim	82
5.33.3.7	observationDim	82
5.34	OscBasicAgent Class Reference	82
5.34.1	Constructor & Destructor Documentation	83
5.34.1.1	OscBasicAgent	83
5.34.1.2	~OscBasicAgent	83
5.34.2	Member Function Documentation	83
5.34.2.1	readAction	83
5.34.3	Member Data Documentation	83
5.34.3.1	currentAction	83
5.35	OscBasicEnvironment Class Reference	84
5.35.1	Constructor & Destructor Documentation	85
5.35.1.1	OscBasicEnvironment	85
5.35.1.2	~OscBasicEnvironment	85
5.35.2	Member Function Documentation	85
5.35.2.1	readObservation	85
5.35.3	Member Data Documentation	85

5.35.3.1	currentObservation	85
5.36	OscEnvironment Class Reference	85
5.36.1	Constructor & Destructor Documentation	87
5.36.1.1	OscEnvironment	87
5.36.1.2	~OscEnvironment	87
5.36.2	Member Function Documentation	87
5.36.2.1	getPath	87
5.36.2.2	handlerInit	87
5.36.2.3	handlerStartStep	87
5.36.2.4	init	87
5.36.2.5	readObservation	87
5.36.2.6	start	87
5.36.2.7	step	87
5.36.3	Member Data Documentation	88
5.36.3.1	actionDim	88
5.36.3.2	id	88
5.36.3.3	locked	88
5.36.3.4	observationBuffer	88
5.36.3.5	observationBufferDim	88
5.36.3.6	observationDim	88
5.36.3.7	observationTypes	88
5.37	OscManager Class Reference	88
5.37.1	Constructor & Destructor Documentation	88
5.37.1.1	OscManager	88
5.37.1.2	~OscManager	88
5.37.2	Member Function Documentation	88
5.37.2.1	client	89
5.37.2.2	error	89
5.37.2.3	initOsc	89
5.37.2.4	repeatChar	89
5.37.2.5	server	89
5.37.3	Member Data Documentation	89
5.37.3.1	_client	89
5.37.3.2	_server	89
5.38	OscRLEnvironment Class Reference	89
5.38.1	Constructor & Destructor Documentation	90
5.38.1.1	OscRLEnvironment	90
5.38.1.2	~OscRLEnvironment	90
5.38.2	Member Function Documentation	90
5.38.2.1	readObservation	90

5.38.3	Member Data Documentation	90
5.38.3.1	currentObservation	90
5.39	Policy Class Reference	91
5.39.1	Constructor & Destructor Documentation	92
5.39.1.1	Policy	92
5.39.1.2	~Policy	92
5.39.2	Member Function Documentation	92
5.39.2.1	chooseAction	92
5.39.2.2	init	92
5.39.2.3	setAgent	92
5.39.3	Member Data Documentation	92
5.39.3.1	agent	92
5.40	Print Class Reference	92
5.40.1	Constructor & Destructor Documentation	93
5.40.1.1	Print	93
5.40.2	Member Function Documentation	93
5.40.2.1	clearWriteError	93
5.40.2.2	getWriteError	93
5.40.2.3	print	94
5.40.2.4	print	94
5.40.2.5	print	94
5.40.2.6	print	94
5.40.2.7	print	94
5.40.2.8	print	94
5.40.2.9	print	94
5.40.2.10	print	94
5.40.2.11	print	94
5.40.2.12	print	94
5.40.2.13	printFloat	94
5.40.2.14	println	94
5.40.2.15	println	94
5.40.2.16	println	94
5.40.2.17	println	94
5.40.2.18	println	94
5.40.2.19	println	94
5.40.2.20	println	94
5.40.2.21	println	94
5.40.2.22	println	94
5.40.2.23	println	94
5.40.2.24	println	94

5.40.2.25	printNumber	94
5.40.2.26	setWriteError	94
5.40.2.27	write	94
5.40.2.28	write	94
5.40.2.29	write	94
5.40.3	Member Data Documentation	94
5.40.3.1	write_error	95
5.41	Printable Class Reference	95
5.41.1	Detailed Description	95
5.41.2	Member Function Documentation	95
5.41.2.1	printTo	95
5.42	QFunction Class Reference	95
5.42.1	Constructor & Destructor Documentation	97
5.42.1.1	QFunction	97
5.42.1.2	~QFunction	97
5.42.2	Member Function Documentation	97
5.42.2.1	backpropagate	97
5.42.2.2	getMaxAction	97
5.42.2.3	getOutput	97
5.42.2.4	getValue	98
5.42.2.5	init	98
5.42.2.6	load	98
5.42.2.7	nInputs	98
5.42.2.8	nOutputs	98
5.42.2.9	nParams	98
5.42.2.10	propagate	98
5.42.2.11	save	98
5.42.2.12	setInput	98
5.42.2.13	update	98
5.42.3	Member Data Documentation	99
5.42.3.1	actionProperties	99
5.42.3.2	function	99
5.42.3.3	input	99
5.42.3.4	observationDim	99
5.43	QLearningAgent Class Reference	99
5.43.1	Detailed Description	100
5.43.2	Constructor & Destructor Documentation	100
5.43.2.1	QLearningAgent	100
5.43.2.2	~QLearningAgent	100
5.43.3	Member Function Documentation	100

5.43.3.1	end	100
5.43.3.2	init	101
5.43.3.3	load	101
5.43.3.4	save	101
5.43.3.5	start	101
5.43.3.6	step	101
5.43.4	Member Data Documentation	101
5.43.4.1	currentAction	101
5.43.4.2	isLearning	101
5.43.4.3	lastAction	101
5.43.4.4	lastObservation	101
5.43.4.5	observationDim	101
5.43.4.6	policy	101
5.43.4.7	qFunction	101
5.43.4.8	trainer	102
5.44	QLearningEDecreasingPolicy Class Reference	102
5.44.1	Constructor & Destructor Documentation	103
5.44.1.1	QLearningEDecreasingPolicy	103
5.44.2	Member Function Documentation	103
5.44.2.1	chooseAction	103
5.44.2.2	getCurrentEpsilon	104
5.44.2.3	init	104
5.44.3	Member Data Documentation	104
5.44.3.1	_epsilonDiv	104
5.44.3.2	decreaseConstant	104
5.45	QLearningEGreedyPolicy Class Reference	104
5.45.1	Detailed Description	105
5.45.2	Constructor & Destructor Documentation	105
5.45.2.1	QLearningEGreedyPolicy	105
5.45.2.2	~QLearningEGreedyPolicy	105
5.45.3	Member Function Documentation	105
5.45.3.1	chooseAction	105
5.45.4	Member Data Documentation	106
5.45.4.1	epsilon	106
5.46	QLearningSoftmaxPolicy Class Reference	106
5.46.1	Detailed Description	107
5.46.2	Constructor & Destructor Documentation	107
5.46.2.1	QLearningSoftmaxPolicy	107
5.46.2.2	~QLearningSoftmaxPolicy	107
5.46.3	Member Function Documentation	107

5.46.3.1	chooseAction	107
5.46.4	Member Data Documentation	107
5.46.4.1	epsilon	107
5.46.4.2	temperature	107
5.47	Qualia Class Reference	108
5.47.1	Constructor & Destructor Documentation	109
5.47.1.1	Qualia	109
5.47.1.2	~Qualia	109
5.47.2	Member Function Documentation	109
5.47.2.1	episode	109
5.47.2.2	init	109
5.47.2.3	start	109
5.47.2.4	step	109
5.47.3	Member Data Documentation	109
5.47.3.1	agent	109
5.47.3.2	environment	109
5.47.3.3	lastAction	109
5.47.3.4	nEpisodes	109
5.47.3.5	nSteps	109
5.47.3.6	observationAction	109
5.48	RealCmdOption Class Reference	110
5.48.1	Detailed Description	111
5.48.2	Constructor & Destructor Documentation	111
5.48.2.1	RealCmdOption	111
5.48.2.2	~RealCmdOption	111
5.48.3	Member Function Documentation	111
5.48.3.1	initValue	111
5.48.3.2	load	111
5.48.3.3	printValue	111
5.48.3.4	read	111
5.48.3.5	save	111
5.48.4	Member Data Documentation	111
5.48.4.1	init_value	111
5.48.4.2	ptr	111
5.49	Reward Class Reference	112
5.49.1	Detailed Description	112
5.49.2	Constructor & Destructor Documentation	112
5.49.2.1	Reward	112
5.49.2.2	~Reward	112
5.49.3	Member Function Documentation	112

5.49.3.1	reward	113
5.49.3.2	setEnvironment	113
5.49.4	Member Data Documentation	113
5.49.4.1	environment	113
5.50	RewardEnvironment Class Reference	113
5.50.1	Detailed Description	114
5.50.2	Constructor & Destructor Documentation	114
5.50.2.1	RewardEnvironment	114
5.50.2.2	~RewardEnvironment	114
5.50.3	Member Function Documentation	115
5.50.3.1	doAction	115
5.50.3.2	doStart	115
5.50.3.3	start	115
5.50.3.4	step	115
5.50.4	Member Data Documentation	115
5.50.4.1	lastObservation	115
5.50.4.2	reward	115
5.51	RLObservation Class Reference	115
5.51.1	Detailed Description	116
5.51.2	Constructor & Destructor Documentation	116
5.51.2.1	RLObservation	116
5.51.3	Member Function Documentation	116
5.51.3.1	copyFrom	116
5.51.3.2	loadData	116
5.51.3.3	saveData	117
5.51.4	Member Data Documentation	117
5.51.4.1	reward	117
5.52	RLQualia Class Reference	117
5.52.1	Constructor & Destructor Documentation	118
5.52.1.1	RLQualia	118
5.52.2	Member Function Documentation	118
5.52.2.1	init	118
5.52.2.2	start	118
5.52.2.3	step	119
5.52.3	Member Data Documentation	119
5.52.3.1	totalReward	119
5.53	MapperConnector::SignalData Struct Reference	119
5.53.1	Constructor & Destructor Documentation	119
5.53.1.1	SignalData	119
5.53.1.2	~SignalData	119

5.53.2	Member Data Documentation	119
5.53.2.1	data	119
5.53.2.2	flag	119
5.53.2.3	isBlocking	119
5.53.2.4	n	119
5.53.2.5	sig	119
5.54	StaticAllocator Class Reference	120
5.54.1	Detailed Description	121
5.54.2	Constructor & Destructor Documentation	121
5.54.2.1	StaticAllocator	121
5.54.3	Member Function Documentation	121
5.54.3.1	free	121
5.54.3.2	freeAll	121
5.54.3.3	malloc	121
5.54.3.4	realloc	121
5.54.4	Member Data Documentation	121
5.54.4.1	buffer	121
5.54.4.2	bufferIdx	121
5.54.4.3	bufferSize	121
5.54.4.4	lastLeak	121
5.54.4.5	nLeaks	122
5.55	StochasticDataSetTrainer Class Reference	122
5.55.1	Detailed Description	123
5.55.2	Constructor & Destructor Documentation	123
5.55.2.1	StochasticDataSetTrainer	123
5.55.2.2	~StochasticDataSetTrainer	123
5.55.3	Member Function Documentation	123
5.55.3.1	_doTrainEpisode	123
5.55.3.2	trainExample	124
5.56	Stream Class Reference	124
5.56.1	Constructor & Destructor Documentation	125
5.56.1.1	Stream	125
5.56.2	Member Function Documentation	125
5.56.2.1	available	125
5.56.2.2	find	125
5.56.2.3	find	125
5.56.2.4	findUntil	125
5.56.2.5	findUntil	125
5.56.2.6	flush	125
5.56.2.7	parseFloat	125

5.56.2.8	parseFloat	125
5.56.2.9	parseInt	125
5.56.2.10	parseInt	125
5.56.2.11	peek	125
5.56.2.12	peekNextDigit	126
5.56.2.13	read	126
5.56.2.14	readBytes	126
5.56.2.15	readBytesUntil	126
5.56.2.16	readString	126
5.56.2.17	readStringUntil	126
5.56.2.18	setTimeout	126
5.56.2.19	timedPeek	126
5.56.2.20	timedRead	126
5.56.3	Member Data Documentation	126
5.56.3.1	_startMillis	126
5.56.3.2	_timeout	126
5.57	StringCmdOption Class Reference	126
5.57.1	Detailed Description	127
5.57.2	Constructor & Destructor Documentation	127
5.57.2.1	StringCmdOption	127
5.57.2.2	~StringCmdOption	127
5.57.3	Member Function Documentation	128
5.57.3.1	initValue	128
5.57.3.2	load	128
5.57.3.3	printValue	128
5.57.3.4	read	128
5.57.3.5	save	128
5.57.4	Member Data Documentation	128
5.57.4.1	init_value	128
5.57.4.2	ptr	128
5.58	SubDataSet Class Reference	128
5.58.1	Constructor & Destructor Documentation	130
5.58.1.1	SubDataSet	130
5.58.1.2	~SubDataSet	130
5.58.2	Member Function Documentation	130
5.58.2.1	getShuffledDataSet	130
5.58.2.2	getStandardDataSets	130
5.58.2.3	init	130
5.58.2.4	reset	130
5.58.2.5	setExample	130

5.58.3	Member Data Documentation	130
5.58.3.1	_dataSet	130
5.58.3.2	_indices	130
5.59	TDTrainer Class Reference	130
5.59.1	Detailed Description	132
5.59.2	Constructor & Destructor Documentation	132
5.59.2.1	TDTrainer	132
5.59.2.2	~TDTrainer	132
5.59.3	Member Function Documentation	132
5.59.3.1	init	132
5.59.3.2	step	132
5.59.4	Member Data Documentation	132
5.59.4.1	actionDim	132
5.59.4.2	bufferAction	132
5.59.4.3	eTraces	132
5.59.4.4	gamma	132
5.59.4.5	lambda	132
5.59.4.6	observationDim	133
5.59.4.7	offPolicy	133
5.60	Trainer Class Reference	133
5.60.1	Detailed Description	134
5.60.2	Constructor & Destructor Documentation	134
5.60.2.1	Trainer	134
5.60.2.2	~Trainer	134
5.60.3	Member Function Documentation	134
5.60.3.1	init	135
5.60.3.2	nEpisodes	135
5.60.4	Member Data Documentation	135
5.60.4.1	_function	135
5.60.4.2	_nEpisodes	135
5.61	TupleDataSet Class Reference	135
5.61.1	Detailed Description	137
5.61.2	Constructor & Destructor Documentation	137
5.61.2.1	TupleDataSet	137
5.61.2.2	~TupleDataSet	137
5.61.3	Member Function Documentation	137
5.61.3.1	actionDim	137
5.61.3.2	init	137
5.61.3.3	observationDim	137
5.61.3.4	reset	137

5.61.3.5	setExample	137
5.61.3.6	tupleFromExample	137
5.61.3.7	tupleToExample	137
5.61.4	Member Data Documentation	137
5.61.4.1	currentExampleIndex	137
5.61.4.2	file	137
5.61.4.3	lastAction	137
5.61.4.4	lastObservation	137
5.61.4.5	observation	138
5.62	XFile Class Reference	138
5.62.1	Detailed Description	139
5.62.2	Constructor & Destructor Documentation	139
5.62.2.1	XFile	139
5.62.2.2	~XFile	139
5.62.3	Member Function Documentation	139
5.62.3.1	eof	139
5.62.3.2	flush	139
5.62.3.3	gets	139
5.62.3.4	read	139
5.62.3.5	rewind	139
5.62.3.6	seek	139
5.62.3.7	size	140
5.62.3.8	taggedRead	140
5.62.3.9	taggedWrite	140
5.62.3.10	tell	140
5.62.3.11	write	140
5.63	XFileDataSet Class Reference	140
5.63.1	Detailed Description	141
5.63.2	Constructor & Destructor Documentation	141
5.63.2.1	XFileDataSet	141
5.63.2.2	~XFileDataSet	141
5.63.3	Member Function Documentation	141
5.63.3.1	init	141
5.63.3.2	reset	142
5.63.3.3	setExample	142
5.63.4	Member Data Documentation	142
5.63.4.1	_currentExampleIndex	142
5.63.4.2	_file	142

6.1	src/mainpage.dox File Reference	143
6.2	src/qualia/compat/Arduino.cpp File Reference	143
6.3	src/qualia/compat/Arduino.h File Reference	143
6.3.1	Macro Definition Documentation	145
6.3.1.1	abs	145
6.3.1.2	bit	145
6.3.1.3	bitClear	145
6.3.1.4	bitRead	145
6.3.1.5	bitSet	145
6.3.1.6	bitWrite	145
6.3.1.7	constrain	145
6.3.1.8	DEG_TO_RAD	145
6.3.1.9	degrees	145
6.3.1.10	false	145
6.3.1.11	HALF_PI	145
6.3.1.12	highByte	145
6.3.1.13	lowByte	145
6.3.1.14	max	145
6.3.1.15	min	145
6.3.1.16	PI	145
6.3.1.17	RAD_TO_DEG	145
6.3.1.18	radians	145
6.3.1.19	round	145
6.3.1.20	sq	145
6.3.1.21	true	146
6.3.1.22	TWO_PI	146
6.3.2	Typedef Documentation	146
6.3.2.1	boolean	146
6.3.2.2	byte	146
6.3.2.3	word	146
6.3.3	Function Documentation	146
6.3.3.1	delay	146
6.3.3.2	delayMicroseconds	146
6.3.3.3	micros	146
6.3.3.4	millis	146
6.4	src/qualia/compat/binary.h File Reference	146
6.4.1	Macro Definition Documentation	156
6.4.1.1	B0	156
6.4.1.2	B00	156
6.4.1.3	B000	156

6.4.1.4	B0000	156
6.4.1.5	B00000	156
6.4.1.6	B000000	156
6.4.1.7	B0000000	156
6.4.1.8	B00000000	156
6.4.1.9	B00000001	156
6.4.1.10	B0000001	156
6.4.1.11	B00000010	156
6.4.1.12	B00000011	156
6.4.1.13	B000001	156
6.4.1.14	B0000010	156
6.4.1.15	B00000100	156
6.4.1.16	B00000101	156
6.4.1.17	B0000011	156
6.4.1.18	B00000110	156
6.4.1.19	B00000111	157
6.4.1.20	B00001	157
6.4.1.21	B000010	157
6.4.1.22	B0000100	157
6.4.1.23	B00001000	157
6.4.1.24	B00001001	157
6.4.1.25	B0000101	157
6.4.1.26	B00001010	157
6.4.1.27	B00001011	157
6.4.1.28	B000011	157
6.4.1.29	B0000110	157
6.4.1.30	B00001100	157
6.4.1.31	B00001101	157
6.4.1.32	B0000111	157
6.4.1.33	B00001110	157
6.4.1.34	B00001111	157
6.4.1.35	B0001	157
6.4.1.36	B00010	157
6.4.1.37	B000100	157
6.4.1.38	B0001000	157
6.4.1.39	B00010000	157
6.4.1.40	B00010001	157
6.4.1.41	B0001001	157
6.4.1.42	B00010010	157
6.4.1.43	B00010011	157

6.4.1.44	B000101	157
6.4.1.45	B0001010	157
6.4.1.46	B00010100	157
6.4.1.47	B00010101	158
6.4.1.48	B0001011	158
6.4.1.49	B00010110	158
6.4.1.50	B00010111	158
6.4.1.51	B00011	158
6.4.1.52	B000110	158
6.4.1.53	B0001100	158
6.4.1.54	B00011000	158
6.4.1.55	B00011001	158
6.4.1.56	B0001101	158
6.4.1.57	B00011010	158
6.4.1.58	B00011011	158
6.4.1.59	B000111	158
6.4.1.60	B0001110	158
6.4.1.61	B00011100	158
6.4.1.62	B00011101	158
6.4.1.63	B0001111	158
6.4.1.64	B00011110	158
6.4.1.65	B00011111	158
6.4.1.66	B001	158
6.4.1.67	B0010	158
6.4.1.68	B00100	158
6.4.1.69	B001000	158
6.4.1.70	B0010000	158
6.4.1.71	B00100000	158
6.4.1.72	B00100001	158
6.4.1.73	B0010001	158
6.4.1.74	B00100010	158
6.4.1.75	B00100011	159
6.4.1.76	B001001	159
6.4.1.77	B0010010	159
6.4.1.78	B00100100	159
6.4.1.79	B00100101	159
6.4.1.80	B0010011	159
6.4.1.81	B00100110	159
6.4.1.82	B00100111	159
6.4.1.83	B00101	159

6.4.1.84 B001010	159
6.4.1.85 B0010100	159
6.4.1.86 B00101000	159
6.4.1.87 B00101001	159
6.4.1.88 B0010101	159
6.4.1.89 B00101010	159
6.4.1.90 B00101011	159
6.4.1.91 B001011	159
6.4.1.92 B0010110	159
6.4.1.93 B00101100	159
6.4.1.94 B00101101	159
6.4.1.95 B0010111	159
6.4.1.96 B00101110	159
6.4.1.97 B00101111	159
6.4.1.98 B0011	159
6.4.1.99 B00110	159
6.4.1.100 B001100	159
6.4.1.101 B0011000	159
6.4.1.102 B00110000	159
6.4.1.103 B00110001	160
6.4.1.104 B0011001	160
6.4.1.105 B00110010	160
6.4.1.106 B00110011	160
6.4.1.107 B001101	160
6.4.1.108 B0011010	160
6.4.1.109 B00110100	160
6.4.1.110 B00110101	160
6.4.1.111 B0011011	160
6.4.1.112 B00110110	160
6.4.1.113 B00110111	160
6.4.1.114 B00111	160
6.4.1.115 B001110	160
6.4.1.116 B0011100	160
6.4.1.117 B00111000	160
6.4.1.118 B00111001	160
6.4.1.119 B0011101	160
6.4.1.120 B00111010	160
6.4.1.121 B00111011	160
6.4.1.122 B001111	160
6.4.1.123 B0011110	160

6.4.1.124 B00111100	160
6.4.1.125 B00111101	160
6.4.1.126 B0011111	160
6.4.1.127 B00111110	160
6.4.1.128 B00111111	160
6.4.1.129 B01	160
6.4.1.130 B010	160
6.4.1.131 B0100	161
6.4.1.132 B01000	161
6.4.1.133 B010000	161
6.4.1.134 B0100000	161
6.4.1.135 B01000000	161
6.4.1.136 B01000001	161
6.4.1.137 B0100001	161
6.4.1.138 B01000010	161
6.4.1.139 B01000011	161
6.4.1.140 B010001	161
6.4.1.141 B0100010	161
6.4.1.142 B01000100	161
6.4.1.143 B01000101	161
6.4.1.144 B0100011	161
6.4.1.145 B01000110	161
6.4.1.146 B01000111	161
6.4.1.147 B01001	161
6.4.1.148 B010010	161
6.4.1.149 B0100100	161
6.4.1.150 B01001000	161
6.4.1.151 B01001001	161
6.4.1.152 B0100101	161
6.4.1.153 B01001010	161
6.4.1.154 B01001011	161
6.4.1.155 B010011	161
6.4.1.156 B0100110	161
6.4.1.157 B01001100	161
6.4.1.158 B01001101	161
6.4.1.159 B0100111	162
6.4.1.160 B01001110	162
6.4.1.161 B01001111	162
6.4.1.162 B0101	162
6.4.1.163 B01010	162

6.4.1.164 B010100	162
6.4.1.165 B0101000	162
6.4.1.166 B01010000	162
6.4.1.167 B01010001	162
6.4.1.168 B0101001	162
6.4.1.169 B01010010	162
6.4.1.170 B01010011	162
6.4.1.171 B010101	162
6.4.1.172 B0101010	162
6.4.1.173 B01010100	162
6.4.1.174 B01010101	162
6.4.1.175 B0101011	162
6.4.1.176 B01010110	162
6.4.1.177 B01010111	162
6.4.1.178 B01011	162
6.4.1.179 B010110	162
6.4.1.180 B0101100	162
6.4.1.181 B01011000	162
6.4.1.182 B01011001	162
6.4.1.183 B0101101	162
6.4.1.184 B01011010	162
6.4.1.185 B01011011	162
6.4.1.186 B010111	162
6.4.1.187 B0101110	163
6.4.1.188 B01011100	163
6.4.1.189 B01011101	163
6.4.1.190 B0101111	163
6.4.1.191 B01011110	163
6.4.1.192 B01011111	163
6.4.1.193 B011	163
6.4.1.194 B0110	163
6.4.1.195 B01100	163
6.4.1.196 B011000	163
6.4.1.197 B0110000	163
6.4.1.198 B01100000	163
6.4.1.199 B01100001	163
6.4.1.200 B0110001	163
6.4.1.201 B01100010	163
6.4.1.202 B01100011	163
6.4.1.203 B011001	163

6.4.1.204 B0110010	163
6.4.1.205 B01100100	163
6.4.1.206 B01100101	163
6.4.1.207 B0110011	163
6.4.1.208 B01100110	163
6.4.1.209 B01100111	163
6.4.1.210 B01101	163
6.4.1.211 B011010	163
6.4.1.212 B0110100	163
6.4.1.213 B01101000	163
6.4.1.214 B01101001	163
6.4.1.215 B0110101	164
6.4.1.216 B01101010	164
6.4.1.217 B01101011	164
6.4.1.218 B011011	164
6.4.1.219 B0110110	164
6.4.1.220 B01101100	164
6.4.1.221 B01101101	164
6.4.1.222 B0110111	164
6.4.1.223 B01101110	164
6.4.1.224 B01101111	164
6.4.1.225 B0111	164
6.4.1.226 B01110	164
6.4.1.227 B011100	164
6.4.1.228 B0111000	164
6.4.1.229 B01110000	164
6.4.1.230 B01110001	164
6.4.1.231 B0111001	164
6.4.1.232 B01110010	164
6.4.1.233 B01110011	164
6.4.1.234 B011101	164
6.4.1.235 B0111010	164
6.4.1.236 B01110100	164
6.4.1.237 B01110101	164
6.4.1.238 B0111011	164
6.4.1.239 B01110110	164
6.4.1.240 B01110111	164
6.4.1.241 B01111	164
6.4.1.242 B011110	164
6.4.1.243 B0111100	165

6.4.1.244 B01111000	165
6.4.1.245 B01111001	165
6.4.1.246 B0111101	165
6.4.1.247 B01111010	165
6.4.1.248 B01111011	165
6.4.1.249 B011111	165
6.4.1.250 B0111110	165
6.4.1.251 B01111100	165
6.4.1.252 B01111101	165
6.4.1.253 B0111111	165
6.4.1.254 B01111110	165
6.4.1.255 B01111111	165
6.4.1.256 B1	165
6.4.1.257 B10	165
6.4.1.258 B100	165
6.4.1.259 B1000	165
6.4.1.260 B10000	165
6.4.1.261 B100000	165
6.4.1.262 B1000000	165
6.4.1.263 B10000000	165
6.4.1.264 B10000001	165
6.4.1.265 B1000001	165
6.4.1.266 B10000010	165
6.4.1.267 B10000011	165
6.4.1.268 B100001	165
6.4.1.269 B1000010	165
6.4.1.270 B10000100	165
6.4.1.271 B10000101	166
6.4.1.272 B1000011	166
6.4.1.273 B10000110	166
6.4.1.274 B10000111	166
6.4.1.275 B10001	166
6.4.1.276 B100010	166
6.4.1.277 B1000100	166
6.4.1.278 B10001000	166
6.4.1.279 B10001001	166
6.4.1.280 B1000101	166
6.4.1.281 B10001010	166
6.4.1.282 B10001011	166
6.4.1.283 B100011	166

6.4.1.284 B1000110	166
6.4.1.285 B10001100	166
6.4.1.286 B10001101	166
6.4.1.287 B1000111	166
6.4.1.288 B10001110	166
6.4.1.289 B10001111	166
6.4.1.290 B1001	166
6.4.1.291 B10010	166
6.4.1.292 B100100	166
6.4.1.293 B1001000	166
6.4.1.294 B10010000	166
6.4.1.295 B10010001	166
6.4.1.296 B1001001	166
6.4.1.297 B10010010	166
6.4.1.298 B10010011	166
6.4.1.299 B100101	167
6.4.1.300 B1001010	167
6.4.1.301 B10010100	167
6.4.1.302 B10010101	167
6.4.1.303 B1001011	167
6.4.1.304 B10010110	167
6.4.1.305 B10010111	167
6.4.1.306 B10011	167
6.4.1.307 B100110	167
6.4.1.308 B1001100	167
6.4.1.309 B10011000	167
6.4.1.310 B10011001	167
6.4.1.311 B1001101	167
6.4.1.312 B10011010	167
6.4.1.313 B10011011	167
6.4.1.314 B100111	167
6.4.1.315 B1001110	167
6.4.1.316 B10011100	167
6.4.1.317 B10011101	167
6.4.1.318 B1001111	167
6.4.1.319 B10011110	167
6.4.1.320 B10011111	167
6.4.1.321 B101	167
6.4.1.322 B1010	167
6.4.1.323 B10100	167

6.4.1.324 B101000	167
6.4.1.325 B1010000	167
6.4.1.326 B10100000	167
6.4.1.327 B10100001	168
6.4.1.328 B1010001	168
6.4.1.329 B10100010	168
6.4.1.330 B10100011	168
6.4.1.331 B101001	168
6.4.1.332 B1010010	168
6.4.1.333 B10100100	168
6.4.1.334 B10100101	168
6.4.1.335 B1010011	168
6.4.1.336 B10100110	168
6.4.1.337 B10100111	168
6.4.1.338 B10101	168
6.4.1.339 B101010	168
6.4.1.340 B1010100	168
6.4.1.341 B10101000	168
6.4.1.342 B10101001	168
6.4.1.343 B1010101	168
6.4.1.344 B10101010	168
6.4.1.345 B10101011	168
6.4.1.346 B101011	168
6.4.1.347 B1010110	168
6.4.1.348 B10101100	168
6.4.1.349 B10101101	168
6.4.1.350 B1010111	168
6.4.1.351 B10101110	168
6.4.1.352 B10101111	168
6.4.1.353 B1011	168
6.4.1.354 B10110	168
6.4.1.355 B101100	169
6.4.1.356 B1011000	169
6.4.1.357 B10110000	169
6.4.1.358 B10110001	169
6.4.1.359 B1011001	169
6.4.1.360 B10110010	169
6.4.1.361 B10110011	169
6.4.1.362 B101101	169
6.4.1.363 B1011010	169

6.4.1.364 B10110100	169
6.4.1.365 B10110101	169
6.4.1.366 B1011011	169
6.4.1.367 B10110110	169
6.4.1.368 B10110111	169
6.4.1.369 B10111	169
6.4.1.370 B101110	169
6.4.1.371 B1011100	169
6.4.1.372 B10111000	169
6.4.1.373 B10111001	169
6.4.1.374 B1011101	169
6.4.1.375 B10111010	169
6.4.1.376 B10111011	169
6.4.1.377 B101111	169
6.4.1.378 B1011110	169
6.4.1.379 B10111100	169
6.4.1.380 B10111101	169
6.4.1.381 B1011111	169
6.4.1.382 B10111110	169
6.4.1.383 B10111111	170
6.4.1.384 B11	170
6.4.1.385 B110	170
6.4.1.386 B1100	170
6.4.1.387 B11000	170
6.4.1.388 B110000	170
6.4.1.389 B1100000	170
6.4.1.390 B11000000	170
6.4.1.391 B11000001	170
6.4.1.392 B1100001	170
6.4.1.393 B11000010	170
6.4.1.394 B11000011	170
6.4.1.395 B110001	170
6.4.1.396 B1100010	170
6.4.1.397 B11000100	170
6.4.1.398 B11000101	170
6.4.1.399 B1100011	170
6.4.1.400 B11000110	170
6.4.1.401 B11000111	170
6.4.1.402 B11001	170
6.4.1.403 B110010	170

6.4.1.404 B1100100	170
6.4.1.405 B11001000	170
6.4.1.406 B11001001	170
6.4.1.407 B1100101	170
6.4.1.408 B11001010	170
6.4.1.409 B11001011	170
6.4.1.410 B110011	170
6.4.1.411 B1100110	171
6.4.1.412 B11001100	171
6.4.1.413 B11001101	171
6.4.1.414 B1100111	171
6.4.1.415 B11001110	171
6.4.1.416 B11001111	171
6.4.1.417 B1101	171
6.4.1.418 B11010	171
6.4.1.419 B110100	171
6.4.1.420 B1101000	171
6.4.1.421 B11010000	171
6.4.1.422 B11010001	171
6.4.1.423 B1101001	171
6.4.1.424 B11010010	171
6.4.1.425 B11010011	171
6.4.1.426 B110101	171
6.4.1.427 B1101010	171
6.4.1.428 B11010100	171
6.4.1.429 B11010101	171
6.4.1.430 B1101011	171
6.4.1.431 B11010110	171
6.4.1.432 B11010111	171
6.4.1.433 B11011	171
6.4.1.434 B110110	171
6.4.1.435 B1101100	171
6.4.1.436 B11011000	171
6.4.1.437 B11011001	171
6.4.1.438 B1101101	171
6.4.1.439 B11011010	172
6.4.1.440 B11011011	172
6.4.1.441 B110111	172
6.4.1.442 B1101110	172
6.4.1.443 B11011100	172

6.4.1.444 B11011101	172
6.4.1.445 B1101111	172
6.4.1.446 B11011110	172
6.4.1.447 B11011111	172
6.4.1.448 B111	172
6.4.1.449 B1110	172
6.4.1.450 B11100	172
6.4.1.451 B111000	172
6.4.1.452 B1110000	172
6.4.1.453 B11100000	172
6.4.1.454 B11100001	172
6.4.1.455 B1110001	172
6.4.1.456 B11100010	172
6.4.1.457 B11100011	172
6.4.1.458 B111001	172
6.4.1.459 B1110010	172
6.4.1.460 B11100100	172
6.4.1.461 B11100101	172
6.4.1.462 B1110011	172
6.4.1.463 B11100110	172
6.4.1.464 B11100111	172
6.4.1.465 B11101	172
6.4.1.466 B111010	172
6.4.1.467 B1110100	173
6.4.1.468 B11101000	173
6.4.1.469 B11101001	173
6.4.1.470 B1110101	173
6.4.1.471 B11101010	173
6.4.1.472 B11101011	173
6.4.1.473 B111011	173
6.4.1.474 B1110110	173
6.4.1.475 B11101100	173
6.4.1.476 B11101101	173
6.4.1.477 B1110111	173
6.4.1.478 B11101110	173
6.4.1.479 B11101111	173
6.4.1.480 B1111	173
6.4.1.481 B11110	173
6.4.1.482 B111100	173
6.4.1.483 B1111000	173

6.4.1.484 B11110000	173
6.4.1.485 B11110001	173
6.4.1.486 B1111001	173
6.4.1.487 B11110010	173
6.4.1.488 B11110011	173
6.4.1.489 B111101	173
6.4.1.490 B1111010	173
6.4.1.491 B11110100	173
6.4.1.492 B11110101	173
6.4.1.493 B1111011	173
6.4.1.494 B11110110	173
6.4.1.495 B11110111	174
6.4.1.496 B11111	174
6.4.1.497 B111110	174
6.4.1.498 B1111100	174
6.4.1.499 B11111000	174
6.4.1.500 B11111001	174
6.4.1.501 B1111101	174
6.4.1.502 B11111010	174
6.4.1.503 B11111011	174
6.4.1.504 B111111	174
6.4.1.505 B1111110	174
6.4.1.506 B11111100	174
6.4.1.507 B11111101	174
6.4.1.508 B1111111	174
6.4.1.509 B11111110	174
6.4.1.510 B11111111	174
6.5 src/qualia/compat/new.cpp File Reference	174
6.6 src/qualia/compat/new.h File Reference	175
6.7 src/qualia/compat/Print.cpp File Reference	175
6.8 src/qualia/compat/Print.h File Reference	176
6.8.1 Macro Definition Documentation	177
6.8.1.1 BIN	177
6.8.1.2 DEC	177
6.8.1.3 HEX	177
6.8.1.4 OCT	177
6.9 src/qualia/compat/Printable.h File Reference	178
6.10 src/qualia/compat/Stream.cpp File Reference	178
6.10.1 Macro Definition Documentation	179
6.10.1.1 NO_SKIP_CHAR	179

6.10.1.2	PARSE_TIMEOUT	179
6.11	src/qualia/compat/Stream.h File Reference	179
6.12	src/qualia/compat/WMath.cpp File Reference	180
6.12.1	Function Documentation	180
6.12.1.1	makeWord	180
6.12.1.2	makeWord	180
6.12.1.3	map	180
6.12.1.4	random	180
6.12.1.5	random	180
6.12.1.6	randomSeed	180
6.13	src/qualia/compat/WString.cpp File Reference	180
6.13.1	Function Documentation	181
6.13.1.1	operator+	181
6.13.1.2	operator+	181
6.13.1.3	operator+	181
6.13.1.4	operator+	181
6.13.1.5	operator+	181
6.13.1.6	operator+	181
6.13.1.7	operator+	181
6.13.1.8	operator+	181
6.14	src/qualia/compat/WString.h File Reference	181
6.15	src/qualia/computer/CmdLine.cpp File Reference	182
6.16	src/qualia/computer/CmdLine.h File Reference	183
6.17	src/qualia/computer/CmdOption.cpp File Reference	184
6.18	src/qualia/computer/CmdOption.h File Reference	184
6.19	src/qualia/computer/DiskXFile.cpp File Reference	186
6.20	src/qualia/computer/DiskXFile.h File Reference	186
6.21	src/qualia/computer/DiskXFileDataSet.cpp File Reference	187
6.21.1	Macro Definition Documentation	188
6.21.1.1	REAL_FORMAT	188
6.22	src/qualia/computer/DiskXFileDataSet.h File Reference	188
6.23	src/qualia/computer/random_shuffle.cpp File Reference	189
6.23.1	Function Documentation	189
6.23.1.1	randomShuffle	189
6.23.1.2	randomShuffle	189
6.23.1.3	randomShuffledIndices	189
6.24	src/qualia/computer/random_shuffle.h File Reference	189
6.24.1	Function Documentation	190
6.24.1.1	randomShuffle	190
6.24.1.2	randomShuffle	190

6.24.1.3	randomShuffledIndices	191
6.25	src/qualia/computer/string_utils.cpp File Reference	191
6.25.1	Function Documentation	191
6.25.1.1	strBaseName	191
6.25.1.2	strConcat	191
6.25.1.3	strRemoveSuffix	191
6.26	src/qualia/computer/string_utils.h File Reference	192
6.26.1	Function Documentation	192
6.26.1.1	strBaseName	192
6.26.1.2	strConcat	193
6.26.1.3	strRemoveSuffix	193
6.27	src/qualia/computer/SubDataSet.cpp File Reference	193
6.28	src/qualia/computer/SubDataSet.h File Reference	193
6.29	src/qualia/core/Action.cpp File Reference	194
6.30	src/qualia/core/Action.h File Reference	195
6.30.1	Typedef Documentation	196
6.30.1.1	action_dim_t	196
6.30.1.2	action_t	196
6.31	src/qualia/core/Agent.h File Reference	196
6.32	src/qualia/core/Allocator.cpp File Reference	197
6.32.1	Function Documentation	197
6.32.1.1	operator new	197
6.33	src/qualia/core/Allocator.h File Reference	197
6.33.1	Function Documentation	198
6.33.1.1	operator new	198
6.34	src/qualia/core/common.h File Reference	198
6.34.1	Macro Definition Documentation	199
6.34.1.1	bitFlip	199
6.34.1.2	INF	199
6.34.1.3	is_arduino	199
6.34.1.4	is_avr	199
6.34.1.5	is_computer	199
6.34.1.6	RANDOM_MAX	199
6.34.2	Typedef Documentation	199
6.34.2.1	real	199
6.35	src/qualia/core/Environment.h File Reference	200
6.36	src/qualia/core/error.cpp File Reference	200
6.37	src/qualia/core/error.h File Reference	201
6.37.1	Macro Definition Documentation	202
6.37.1.1	__DEBUG_DUMMY_INSTRUCTION	202

6.37.1.2	__DEBUG_TRIGGER_ASSERT	202
6.37.1.3	__STRING	202
6.37.1.4	DEBUG_ERROR	202
6.37.1.5	DEBUG_LEVEL	202
6.37.1.6	DEBUG_LEVEL_ERROR	202
6.37.1.7	DEBUG_LEVEL_NODEBUG	202
6.37.1.8	DEBUG_LEVEL_NOTICE	203
6.37.1.9	DEBUG_LEVEL_WARNING	203
6.37.1.10	DEBUG_NOTICE	203
6.37.1.11	DEBUG_WARNING	203
6.37.1.12	Q_ASSERT_ERROR	203
6.37.1.13	Q_ASSERT_ERROR_MESSAGE	203
6.37.1.14	Q_ASSERT_NOTICE	203
6.37.1.15	Q_ASSERT_NOTICE_MESSAGE	203
6.37.1.16	Q_ASSERT_WARNING	203
6.37.1.17	Q_ASSERT_WARNING_MESSAGE	203
6.37.1.18	Q_ERROR	203
6.37.1.19	Q_MESSAGE	203
6.37.1.20	Q_NOTICE	203
6.37.1.21	Q_PRINT	203
6.37.1.22	Q_WARNING	203
6.38	src/qualia/core/FileExportEnvironment.cpp File Reference	204
6.39	src/qualia/core/FileExportEnvironment.h File Reference	204
6.40	src/qualia/core/MultiAgent.cpp File Reference	205
6.41	src/qualia/core/MultiAgent.h File Reference	205
6.42	src/qualia/core/Observation.cpp File Reference	206
6.43	src/qualia/core/Observation.h File Reference	207
6.43.1	Typedef Documentation	208
6.43.1.1	observation_t	208
6.44	src/qualia/core/Qualia.cpp File Reference	208
6.45	src/qualia/core/Qualia.h File Reference	208
6.46	src/qualia/core/StaticAllocator.cpp File Reference	209
6.47	src/qualia/core/StaticAllocator.h File Reference	210
6.48	src/qualia/core/XFile.cpp File Reference	211
6.49	src/qualia/core/XFile.h File Reference	211
6.49.1	Macro Definition Documentation	212
6.49.1.1	SEEK_CUR	212
6.49.1.2	SEEK_END	212
6.49.1.3	SEEK_SET	212
6.50	src/qualia/ga/BinaryChromosome.cpp File Reference	212

6.51	src/qualia/ga/BinaryChromosome.h File Reference	213
6.52	src/qualia/ga/Chromosome.h File Reference	214
6.52.1	Typedef Documentation	215
6.52.1.1	AsexualCrossover	215
6.52.1.2	Comparator	215
6.52.1.3	Evaluator	215
6.52.1.4	Initializer	215
6.52.1.5	Mutator	215
6.52.1.6	SexualCrossover	215
6.53	src/qualia/learning/DataSet.cpp File Reference	215
6.54	src/qualia/learning/DataSet.h File Reference	216
6.55	src/qualia/learning/DataSetTrainer.cpp File Reference	216
6.56	src/qualia/learning/DataSetTrainer.h File Reference	217
6.57	src/qualia/learning/Function.cpp File Reference	218
6.58	src/qualia/learning/Function.h File Reference	218
6.59	src/qualia/learning/GradientFunction.h File Reference	219
6.60	src/qualia/learning/MemoryDataSet.cpp File Reference	220
6.61	src/qualia/learning/MemoryDataSet.h File Reference	221
6.62	src/qualia/learning/MultiDataSet.cpp File Reference	222
6.63	src/qualia/learning/MultiDataSet.h File Reference	222
6.64	src/qualia/learning/NeuralNetwork.cpp File Reference	223
6.65	src/qualia/learning/NeuralNetwork.h File Reference	224
6.66	src/qualia/learning/StochasticDataSetTrainer.cpp File Reference	225
6.67	src/qualia/learning/StochasticDataSetTrainer.h File Reference	226
6.68	src/qualia/learning/Trainer.cpp File Reference	226
6.69	src/qualia/learning/Trainer.h File Reference	227
6.70	src/qualia/learning/XFileDataSet.cpp File Reference	228
6.71	src/qualia/learning/XFileDataSet.h File Reference	228
6.72	src/qualia/plugins/mapper/MapperBasicEnvironment.cpp File Reference	229
6.73	src/qualia/plugins/mapper/MapperBasicEnvironment.h File Reference	230
6.74	src/qualia/plugins/mapper/MapperConnector.cpp File Reference	231
6.75	src/qualia/plugins/mapper/MapperConnector.h File Reference	231
6.76	src/qualia/plugins/mapper/MapperEnvironment.cpp File Reference	232
6.77	src/qualia/plugins/mapper/MapperEnvironment.h File Reference	233
6.78	src/qualia/plugins/osc/OscAgent.cpp File Reference	234
6.79	src/qualia/plugins/osc/OscAgent.h File Reference	234
6.80	src/qualia/plugins/osc/OscBasicAgent.cpp File Reference	235
6.81	src/qualia/plugins/osc/OscBasicAgent.h File Reference	236
6.82	src/qualia/plugins/osc/OscBasicEnvironment.cpp File Reference	237
6.83	src/qualia/plugins/osc/OscBasicEnvironment.h File Reference	237

6.84	src/qualia/plugins/osc/OscEnvironment.cpp File Reference	238
6.85	src/qualia/plugins/osc/OscEnvironment.h File Reference	239
6.86	src/qualia/plugins/osc/OscManager.cpp File Reference	240
6.87	src/qualia/plugins/osc/OscManager.h File Reference	240
6.88	src/qualia/plugins/osc/OscRLEnvironment.cpp File Reference	241
6.89	src/qualia/plugins/osc/OscRLEnvironment.h File Reference	242
6.90	src/qualia/rl/BatchRLTrainer.cpp File Reference	243
6.91	src/qualia/rl/BatchRLTrainer.h File Reference	243
6.92	src/qualia/rl/Policy.h File Reference	244
6.93	src/qualia/rl/QFunction.cpp File Reference	245
6.94	src/qualia/rl/QFunction.h File Reference	246
6.95	src/qualia/rl/QLearningAgent.cpp File Reference	247
6.96	src/qualia/rl/QLearningAgent.h File Reference	248
6.97	src/qualia/rl/QLearningEDecreasingPolicy.cpp File Reference	249
6.98	src/qualia/rl/QLearningEDecreasingPolicy.h File Reference	249
6.99	src/qualia/rl/QLearningEGreedyPolicy.cpp File Reference	250
6.100	src/qualia/rl/QLearningEGreedyPolicy.h File Reference	250
6.101	src/qualia/rl/QLearningSoftmaxPolicy.cpp File Reference	251
6.102	src/qualia/rl/QLearningSoftmaxPolicy.h File Reference	252
6.103	src/qualia/rl/Reward.h File Reference	253
6.104	src/qualia/rl/RewardEnvironment.cpp File Reference	254
6.105	src/qualia/rl/RewardEnvironment.h File Reference	255
6.106	src/qualia/rl/RLObservation.cpp File Reference	255
6.107	src/qualia/rl/RLObservation.h File Reference	256
6.108	src/qualia/rl/RLQualia.cpp File Reference	257
6.109	src/qualia/rl/RLQualia.h File Reference	257
6.110	src/qualia/rl/TDTrainer.cpp File Reference	258
6.111	src/qualia/rl/TDTrainer.h File Reference	259
6.112	src/qualia/rl/TupleDataSet.cpp File Reference	260
6.113	src/qualia/rl/TupleDataSet.h File Reference	260
6.114	src/qualia/util/bitarray.cpp File Reference	261
6.114.1	Function Documentation	262
6.114.1.1	arrayBitClear	262
6.114.1.2	arrayBitFlip	262
6.114.1.3	arrayBitRead	262
6.114.1.4	arrayBitSet	262
6.114.1.5	arrayBitWrite	262
6.114.1.6	arrayBlockCopy	262
6.114.1.7	arrayBlockWrite	262
6.114.1.8	initPointerAndPositionForBitOperation	262

6.115src/qualia/util/bitarray.h File Reference	262
6.115.1 Macro Definition Documentation	263
6.115.1.1 BITARRAY_BIT_IN_BYTE	263
6.115.1.2 BITARRAY_BIT_TO_BYTE	264
6.115.2 Function Documentation	264
6.115.2.1 arrayBitClear	264
6.115.2.2 arrayBitFlip	264
6.115.2.3 arrayBitRead	264
6.115.2.4 arrayBitSet	264
6.115.2.5 arrayBitWrite	264
6.115.2.6 arrayBlockCopy	264
6.115.2.7 arrayBlockWrite	264
6.115.2.8 initPointerAndPositionForBitOperation	264
6.116src/qualia/util/log_add.cpp File Reference	264
6.116.1 Macro Definition Documentation	264
6.116.1.1 MINUS_LOG_THRESHOLD	265
6.116.2 Function Documentation	265
6.116.2.1 logAdd	265
6.116.2.2 logSub	265
6.117src/qualia/util/log_add.h File Reference	265
6.117.1 Macro Definition Documentation	266
6.117.1.1 log1p	266
6.117.1.2 LOG_2_PI	266
6.117.1.3 LOG_ONE	266
6.117.1.4 LOG_ZERO	266
6.117.2 Function Documentation	266
6.117.2.1 logAdd	266
6.117.2.2 logSub	266
6.118src/qualia/util/map.cpp File Reference	267
6.118.1 Function Documentation	267
6.118.1.1 mapReal	267
6.119src/qualia/util/map.h File Reference	267
6.119.1 Function Documentation	268
6.119.1.1 mapReal	268
6.120src/qualia/util/MovingAverage.h File Reference	268
6.121src/qualia/util/random.cpp File Reference	269
6.121.1 Function Documentation	269
6.121.1.1 randomUniform	269
6.121.1.2 randomUniform	269
6.121.1.3 randomUniform	269

6.122src/qualia/util/random.h File Reference	269
6.122.1 Function Documentation	270
6.122.1.1 randomUniform	270
6.122.1.2 randomUniform	270
6.122.1.3 randomUniform	270
6.123src/qualia/util/random_dist.cpp File Reference	271
6.123.1 Function Documentation	271
6.123.1.1 bernouilli	271
6.123.1.2 cauchy	271
6.123.1.3 exponential	271
6.123.1.4 geometric	271
6.124src/qualia/util/random_dist.h File Reference	272
6.124.1 Function Documentation	272
6.124.1.1 bernouilli	272
6.124.1.2 cauchy	273
6.124.1.3 exponential	273
6.124.1.4 geometric	273
6.125src/qualia/util/random_normal.cpp File Reference	273
6.125.1 Function Documentation	273
6.125.1.1 randomNormal	273
6.126src/qualia/util/random_normal.h File Reference	274
6.126.1 Function Documentation	274
6.126.1.1 randomNormal	274

Chapter 1

Main Page

[Qualia](#) is a free (FLOSS) software library for agent-based systems written in simple C++. It is meant to run on embedded devices as well as on general purpose computers, with a specific focus on AVR microcontrollers such as the ones used on Arduino. It makes it easy for designers, artists and researchers to experiment with different sorts of agents (reinforcement learning, finite state machines, cellular automata, etc) and environments (simulation, physical/embedded, network-based, etc).

Key concepts

An [Agent](#) is a device/being/thing that takes actions in response to observations. In [Qualia](#), agents will typically be platform-agnostic.

An [Environment](#) is a component that gives back observations in response to actions. In [Qualia](#), environments typically implement the platform-specific behaviors of actually "doing" the action (eg. by asking a microcontroller to send a signal to one of its pins) and "reading" the observation (eg. by getting them from a sensor).

A [Qualia](#) is simply the combination of an agent and an environment, managing the interaction between them both and thus creating the behavioral loop.

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Action	11
ActionProperties	14
Agent	15
MultiAgent	67
OscAgent	79
OscBasicAgent	82
QLearningAgent	99
Alloc	17
Allocator	18
StaticAllocator	120
BinaryChromosomeProperties	25
Chromosome	29
BinaryChromosome	22
CmdLine	30
CmdOption	34
BoolCmdOption	27
IntCmdOption	55
RealCmdOption	110
StringCmdOption	126
DataSet	37
MemoryDataSet	65
MultiDataSet	70
SubDataSet	128
TupleDataSet	135
XFileDataSet	140
DiskXFileDataSet	45
Environment	47
FileExportEnvironment	48
MapperEnvironment	62
MapperBasicEnvironment	58
OscEnvironment	85
OscBasicEnvironment	84
OscRLEnvironment	89
RewardEnvironment	113
Function	50
GradientFunction	52

NeuralNetwork	72
QFunction	95
NeuralNetwork::Layer	57
MapperConnector	60
MovingAverage	66
Observation	76
RLObservation	115
ObservationAction	79
OscManager	88
Policy	91
QLearningEGreedyPolicy	104
QLearningEDecreasingPolicy	102
QLearningSoftmaxPolicy	106
Print	92
Stream	124
Printable	95
Qualia	108
RLQualia	117
Reward	112
MapperConnector::SignalData	119
Trainer	133
DataSetTrainer	39
BatchRLTrainer	20
StochasticDataSetTrainer	122
TDTrainer	130
XFile	138
DiskXFile	41

Chapter 3

Class Index

3.1 Class List

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

Action	11
ActionProperties	14
Agent	15
Alloc	17
Allocator	18
BatchRLTrainer	20
BinaryChromosome	22
BinaryChromosomeProperties	25
BoolCmdOption	27
Chromosome	29
CmdLine	30
CmdOption	34
DataSet	37
DataSetTrainer	39
DiskXFile	41
DiskXFileDataSet	45
Environment	47
FileExportEnvironment	48
Function	50
GradientFunction	52
IntCmdOption	55
NeuralNetwork::Layer	
Layer structure	57
MapperBasicEnvironment	58
MapperConnector	60
MapperEnvironment	62
MemoryDataSet	65
MovingAverage	66
MultiAgent	67
MultiDataSet	70
NeuralNetwork	72
Observation	76
ObservationAction	79
OscAgent	79
OscBasicAgent	82
OscBasicEnvironment	84
OscEnvironment	85
OscManager	88

OscRLEnvironment	89
Policy	91
Print	92
Printable	95
QFunction	95
QLearningAgent	99
QLearningEDecreasingPolicy	102
QLearningEGreedyPolicy	104
QLearningSoftmaxPolicy	106
Qualia	108
RealCmdOption	110
Reward	112
RewardEnvironment	113
RLObservation	115
RLQualia	117
MapperConnector::SignalData	119
StaticAllocator	120
StochasticDataSetTrainer	122
Stream	124
StringCmdOption	126
SubDataSet	128
TDTrainer	130
Trainer	133
TupleDataSet	135
XFile	138
XFileDataSet	140

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

src/qualia/compat/Arduino.cpp	143
src/qualia/compat/Arduino.h	143
src/qualia/compat/binary.h	146
src/qualia/compat/new.cpp	174
src/qualia/compat/new.h	175
src/qualia/compat/Print.cpp	175
src/qualia/compat/Print.h	176
src/qualia/compat/Printable.h	178
src/qualia/compat/Stream.cpp	178
src/qualia/compat/Stream.h	179
src/qualia/compat/WMath.cpp	180
src/qualia/compat/WString.cpp	180
src/qualia/compat/WString.h	181
src/qualia/computer/CmdLine.cpp	182
src/qualia/computer/CmdLine.h	183
src/qualia/computer/CmdOption.cpp	184
src/qualia/computer/CmdOption.h	184
src/qualia/computer/DiskXFile.cpp	186
src/qualia/computer/DiskXFile.h	186
src/qualia/computer/DiskXFileDataSet.cpp	187
src/qualia/computer/DiskXFileDataSet.h	188
src/qualia/computer/random_shuffle.cpp	189
src/qualia/computer/random_shuffle.h	189
src/qualia/computer/string_utils.cpp	191
src/qualia/computer/string_utils.h	192
src/qualia/computer/SubDataSet.cpp	193
src/qualia/computer/SubDataSet.h	193
src/qualia/core/Action.cpp	194
src/qualia/core/Action.h	195
src/qualia/core/Agent.h	196
src/qualia/core/Allocator.cpp	197
src/qualia/core/Allocator.h	197
src/qualia/core/common.h	198
src/qualia/core/Environment.h	200
src/qualia/core/error.cpp	200
src/qualia/core/error.h	201
src/qualia/core/FileExportEnvironment.cpp	204
src/qualia/core/FileExportEnvironment.h	204

src/qualia/core/MultiAgent.cpp	205
src/qualia/core/MultiAgent.h	205
src/qualia/core/Observation.cpp	206
src/qualia/core/Observation.h	207
src/qualia/core/Qualia.cpp	208
src/qualia/core/Qualia.h	208
src/qualia/core/StaticAllocator.cpp	209
src/qualia/core/StaticAllocator.h	210
src/qualia/core/XFile.cpp	211
src/qualia/core/XFile.h	211
src/qualia/ga/BinaryChromosome.cpp	212
src/qualia/ga/BinaryChromosome.h	213
src/qualia/ga/Chromosome.h	214
src/qualia/learning/DataSet.cpp	215
src/qualia/learning/DataSet.h	216
src/qualia/learning/DataSetTrainer.cpp	216
src/qualia/learning/DataSetTrainer.h	217
src/qualia/learning/Function.cpp	218
src/qualia/learning/Function.h	218
src/qualia/learning/GradientFunction.h	219
src/qualia/learning/MemoryDataSet.cpp	220
src/qualia/learning/MemoryDataSet.h	221
src/qualia/learning/MultiDataSet.cpp	222
src/qualia/learning/MultiDataSet.h	222
src/qualia/learning/NeuralNetwork.cpp	223
src/qualia/learning/NeuralNetwork.h	224
src/qualia/learning/StochasticDataSetTrainer.cpp	225
src/qualia/learning/StochasticDataSetTrainer.h	226
src/qualia/learning/Trainer.cpp	226
src/qualia/learning/Trainer.h	227
src/qualia/learning/XFileDataSet.cpp	228
src/qualia/learning/XFileDataSet.h	228
src/qualia/plugins/mapper/MapperBasicEnvironment.cpp	229
src/qualia/plugins/mapper/MapperBasicEnvironment.h	230
src/qualia/plugins/mapper/MapperConnector.cpp	231
src/qualia/plugins/mapper/MapperConnector.h	231
src/qualia/plugins/mapper/MapperEnvironment.cpp	232
src/qualia/plugins/mapper/MapperEnvironment.h	233
src/qualia/plugins/osc/OscAgent.cpp	234
src/qualia/plugins/osc/OscAgent.h	234
src/qualia/plugins/osc/OscBasicAgent.cpp	235
src/qualia/plugins/osc/OscBasicAgent.h	236
src/qualia/plugins/osc/OscBasicEnvironment.cpp	237
src/qualia/plugins/osc/OscBasicEnvironment.h	237
src/qualia/plugins/osc/OscEnvironment.cpp	238
src/qualia/plugins/osc/OscEnvironment.h	239
src/qualia/plugins/osc/OscManager.cpp	240
src/qualia/plugins/osc/OscManager.h	240
src/qualia/plugins/osc/OscRLEnvironment.cpp	241
src/qualia/plugins/osc/OscRLEnvironment.h	242
src/qualia/rl/BatchRLTrainer.cpp	243
src/qualia/rl/BatchRLTrainer.h	243
src/qualia/rl/Policy.h	244
src/qualia/rl/QFunction.cpp	245
src/qualia/rl/QFunction.h	246
src/qualia/rl/QLearningAgent.cpp	247
src/qualia/rl/QLearningAgent.h	248
src/qualia/rl/QLearningEDecreasingPolicy.cpp	249

src/qualia/rl/QLearningEDecreasingPolicy.h	249
src/qualia/rl/QLearningEGreedyPolicy.cpp	250
src/qualia/rl/QLearningEGreedyPolicy.h	250
src/qualia/rl/QLearningSoftmaxPolicy.cpp	251
src/qualia/rl/QLearningSoftmaxPolicy.h	252
src/qualia/rl/Reward.h	253
src/qualia/rl/RewardEnvironment.cpp	254
src/qualia/rl/RewardEnvironment.h	255
src/qualia/rl/RLObservation.cpp	255
src/qualia/rl/RLObservation.h	256
src/qualia/rl/RLQualia.cpp	257
src/qualia/rl/RLQualia.h	257
src/qualia/rl/TDTrainer.cpp	258
src/qualia/rl/TDTrainer.h	259
src/qualia/rl/TupleDataSet.cpp	260
src/qualia/rl/TupleDataSet.h	260
src/qualia/util/bitarray.cpp	261
src/qualia/util/bitarray.h	262
src/qualia/util/log_add.cpp	264
src/qualia/util/log_add.h	265
src/qualia/util/map.cpp	267
src/qualia/util/map.h	267
src/qualia/util/MovingAverage.h	268
src/qualia/util/random.cpp	269
src/qualia/util/random.h	269
src/qualia/util/random_dist.cpp	271
src/qualia/util/random_dist.h	272
src/qualia/util/random_normal.cpp	273
src/qualia/util/random_normal.h	274

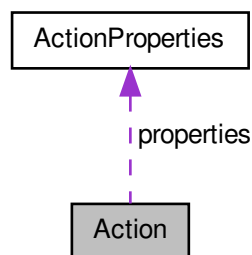
Chapter 5

Class Documentation

5.1 Action Class Reference

```
#include <Action.h>
```

Collaboration diagram for Action:



Public Member Functions

- **Action** (**ActionProperties** ***properties**)
Class constructor.
- virtual **~Action** ()
Class destructor.
- **action_dim_t** & **operator[]** (int i) const
Access operator.
- virtual **action_t** **conflated** () const
Returns the conflated representation of current action.
- virtual **Action** & **setConflated** (**action_t** action)
Sets the action to the given conflated representation action.
- bool **undefined** () const
*Returns true if this action is undefined (see **Action::reset()**).*
- virtual **Action** & **reset** ()
- virtual bool **hasNext** ()

True iff iterator has a next action.

- virtual [Action](#) & [next](#) ()
- virtual [Action](#) & [copyFrom](#) (const [Action](#) &src)

Copies data from src.

- unsigned int [dim](#) () const

Dimension of actions.

- unsigned long [nConflated](#) () const

Number of conflated actions.

- unsigned int [nActions](#) (int i) const

Number of possible actions in dimension i.

- virtual void [saveData](#) ([XFile](#) *file) const

Save data to file (in binary format).

- virtual void [loadData](#) ([XFile](#) *file)

Loads data from file (in binary format). Dimension needs to be known in advance.

Public Attributes

- [ActionProperties](#) * [properties](#)

Pointer to the action properties (read-only).

- [action_dim_t](#) * [actions](#)

The array containing the actions (of size [dim\(\)](#)).

- bool [_undefined](#)

Reset flag.

5.1.1 Detailed Description

Represent an agent action. Actions in [Qualia](#) are always discrete. A specific action has two representations:

1. A multi-dimensional array of integers (each of type [action_dim_t](#))
2. A single integer (conflated representation, of type [action_t](#))

Meta-informations about the actions (dimensions and number of actions per dimension) are contained within an [ActionProperties](#) object that can be shared accross many actions.

5.1.2 Constructor & Destructor Documentation

5.1.2.1 [Action::Action](#) ([ActionProperties](#) * [properties](#))

Class constructor.

5.1.2.2 [Action::~~Action](#) () [virtual]

Class destructor.

5.1.3 Member Function Documentation

5.1.3.1 [action_t](#) [Action::conflated](#) () const [virtual]

Returns the conflated representation of current action.

5.1.3.2 Action & Action::copyFrom (const Action & src) [virtual]

Copies data from *src*.

5.1.3.3 unsigned int Action::dim () const [inline]

Dimension of actions.

5.1.3.4 bool Action::hasNext () [virtual]

True iff iterator has a next action.

5.1.3.5 void Action::loadData (XFile * file) [virtual]

Loads data from *file* (in binary format). Dimension needs to be known in advance.

5.1.3.6 unsigned int Action::nActions (int i) const [inline]

Number of possible actions in dimension *i*.

5.1.3.7 unsigned long Action::nConflated () const [inline]

Number of conflated actions.

5.1.3.8 Action & Action::next () [virtual]

Resets iterator. Typically equivalent to setConflated(getConflated()+1) (although subclasses can change that behavior).

5.1.3.9 action_dim_t& Action::operator[] (int i) const [inline]

Access operator.

5.1.3.10 Action & Action::reset () [virtual]

Resets iterator. Notice that this sets the action to be undefined until the first call to [next\(\)](#).

5.1.3.11 void Action::saveData (XFile * file) const [virtual]

Save data to *file* (in binary format).

5.1.3.12 Action & Action::setConflated (action_t action) [virtual]

Sets the action to the given conflated representation *action*.

5.1.3.13 bool Action::undefined () const [inline]

Returns true if this action is undefined (see [Action::reset\(\)](#)).

5.1.4 Member Data Documentation

5.1.4.1 `bool Action::_undefined`

Reset flag.

5.1.4.2 `action_dim_t* Action::actions`

The array containing the actions (of size `dim()`).

5.1.4.3 `ActionProperties* Action::properties`

Pointer to the action properties (read-only).

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

- `src/qualia/core/Action.h`
- `src/qualia/core/Action.cpp`

5.2 ActionProperties Class Reference

```
#include <Action.h>
```

Public Member Functions

- `ActionProperties` (unsigned int `dim`, const unsigned int *`nActions`)
Class constructor.
- virtual `~ActionProperties` ()
Class destructor.
- unsigned int `dim` () const
Dimension of actions.
- unsigned long `nConflated` () const
Number of conflated actions.
- unsigned int `nActions` (int i) const
Number of possible actions in dimension #i#.
- bool `equals` (const `ActionProperties` &p) const
Comparison method.
- `action_t random` () const
Returns a random action.

Public Attributes

- unsigned int `_dim`
- unsigned int * `_nActions`
- unsigned long `_nConflated`

5.2.1 Detailed Description

Represents a set of properties (basically, action dimension and number of possible actions per dimension) from which specific actions are drawn.

5.2.2 Constructor & Destructor Documentation

5.2.2.1 `ActionProperties::ActionProperties (unsigned int dim, const unsigned int * nActions)`

Class constructor.

5.2.2.2 `ActionProperties::~~ActionProperties () [virtual]`

Class destructor.

5.2.3 Member Function Documentation

5.2.3.1 `unsigned int ActionProperties::dim () const [inline]`

Dimension of actions.

5.2.3.2 `bool ActionProperties::equals (const ActionProperties & p) const`

Comparison method.

5.2.3.3 `unsigned int ActionProperties::nActions (int i) const [inline]`

Number of possible actions in dimension *i*.

5.2.3.4 `unsigned long ActionProperties::nConflated () const [inline]`

Number of conflated actions.

5.2.3.5 `action_t ActionProperties::random () const [inline]`

Returns a random action.

5.2.4 Member Data Documentation

5.2.4.1 `unsigned int ActionProperties::_dim`

5.2.4.2 `unsigned int* ActionProperties::_nActions`

5.2.4.3 `unsigned long ActionProperties::_nConflated`

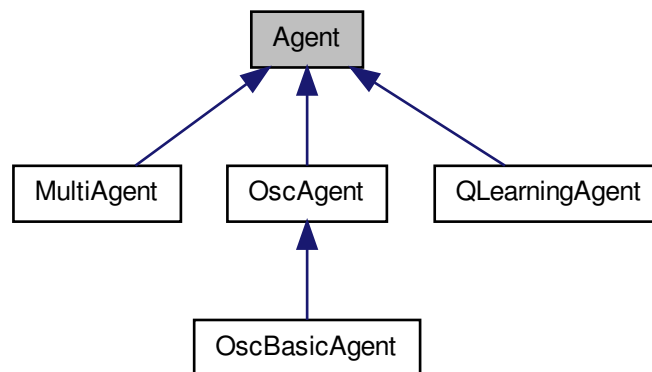
The documentation for this class was generated from the following files:

- `src/qualia/core/Action.h`
- `src/qualia/core/Action.cpp`

5.3 Agent Class Reference

```
#include <Agent.h>
```

Inheritance diagram for Agent:



Public Member Functions

- virtual `~Agent()`
- virtual void `init()`
Initializes the agent.
- virtual `Action * start (const Observation *observation)=0`
Chooses the first action in the episode.
- virtual `Action * step (const Observation *observation)=0`
Performs an episode step.
- virtual void `end (const Observation *observation)`
Performs the last episode step.
- virtual void `save (XFile *file)`
Saves the agent.
- virtual void `load (XFile *file)`
Loads an agent.

5.3.1 Detailed Description

Provides an interface for an agent acting within an environment in the [Qualia](#) framework. An agent is a device/being/thing that takes actions in response to observations. In [Qualia](#), agents will typically be platform-agnostic.

5.3.2 Constructor & Destructor Documentation

5.3.2.1 `virtual Agent::~Agent()` `[inline], [virtual]`

5.3.3 Member Function Documentation

5.3.3.1 `virtual void Agent::end (const Observation * observation)` `[inline], [virtual]`

Performs the last episode step.

Reimplemented in [QLearningAgent](#), and [MultiAgent](#).

5.3.3.2 `virtual void Agent::init () [inline],[virtual]`

Initializes the agent.

Reimplemented in [QLearningAgent](#), [MultiAgent](#), and [OscAgent](#).

5.3.3.3 `virtual void Agent::load (XFile * file) [inline],[virtual]`

Loads an agent.

Reimplemented in [QLearningAgent](#).

5.3.3.4 `virtual void Agent::save (XFile * file) [inline],[virtual]`

Saves the agent.

Reimplemented in [QLearningAgent](#).

5.3.3.5 `virtual Action* Agent::start (const Observation * observation) [pure virtual]`

Chooses the first action in the episode.

Implemented in [QLearningAgent](#), [MultiAgent](#), and [OscAgent](#).

5.3.3.6 `virtual Action* Agent::step (const Observation * observation) [pure virtual]`

Performs an episode step.

Implemented in [QLearningAgent](#), [MultiAgent](#), and [OscAgent](#).

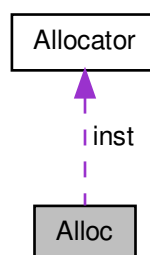
The documentation for this class was generated from the following file:

- [src/qualia/core/Agent.h](#)

5.4 Alloc Class Reference

```
#include <Allocator.h>
```

Collaboration diagram for Alloc:



Static Public Member Functions

- static void * [malloc](#) (size_t size)
- static void * [realloc](#) (void *ptr, size_t size)
- static void [free](#) (void *ptr)
- static void [init](#) ([Allocator](#) *alloc)
- static [Allocator](#) * [instance](#) ()

Static Private Attributes

- static [Allocator](#) * [inst](#) = 0

5.4.1 Detailed Description

Static class for memory management. In [Qualia](#), all dynamic memory allocation calls should use the [Alloc](#) static methods instead of the ones provided in the standard library. That allows code to be easily ported to AVR platform where static allocation is usually better (see [StaticAllocator](#)).

Example use (see class [StaticAllocator](#)).

```
#if is_computer()
Allocator allocator; // use standard allocator
#else
unsigned char STATIC_BUFFER[1000];
StaticAllocator allocator(&STATIC_BUFFER); // use static allocator
#endif

Alloc::init(&allocator);
int* val = (int*) Alloc::malloc(10*sizeof(int));
MyObject* obj = new(Alloc::instance()) MyObject;
...
Alloc::free(val);
```

5.4.2 Member Function Documentation

5.4.2.1 void [Alloc::free](#) (void * *ptr*) [static]

5.4.2.2 void [Alloc::init](#) ([Allocator](#) * *alloc*) [static]

5.4.2.3 static [Allocator](#)* [Alloc::instance](#) () [inline],[static]

5.4.2.4 void * [Alloc::malloc](#) (size_t *size*) [static]

5.4.2.5 void * [Alloc::realloc](#) (void * *ptr*, size_t *size*) [static]

5.4.3 Member Data Documentation

5.4.3.1 [Allocator](#) * [Alloc::inst](#) = 0 [static],[private]

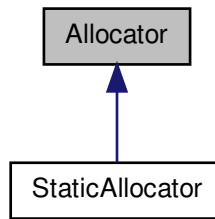
The documentation for this class was generated from the following files:

- src/qualia/core/[Allocator.h](#)
- src/qualia/core/[Allocator.cpp](#)

5.5 Allocator Class Reference

```
#include <Allocator.h>
```

Inheritance diagram for Allocator:



Public Member Functions

- virtual [~Allocator](#) ()
- virtual void * [malloc](#) (size_t size)
- virtual void * [realloc](#) (void *ptr, size_t size)
- virtual void [free](#) (void *ptr)

5.5.1 Detailed Description

Allows for different implementation of memory allocation methods. This is the parent class, which contains basic implementations of malloc/realloc/free functions from the standard library. Subclasses may override these functions.

5.5.2 Constructor & Destructor Documentation

5.5.2.1 virtual Allocator::~Allocator () [inline],[virtual]

5.5.3 Member Function Documentation

5.5.3.1 void Allocator::free (void * ptr) [virtual]

Reimplemented in [StaticAllocator](#).

5.5.3.2 void * Allocator::malloc (size_t size) [virtual]

Reimplemented in [StaticAllocator](#).

5.5.3.3 void * Allocator::realloc (void * ptr, size_t size) [virtual]

Reimplemented in [StaticAllocator](#).

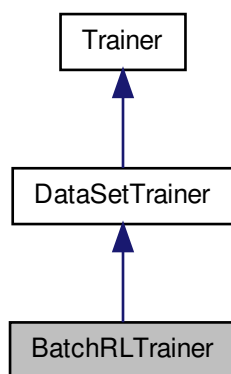
The documentation for this class was generated from the following files:

- src/qualia/core/[Allocator.h](#)
- src/qualia/core/[Allocator.cpp](#)

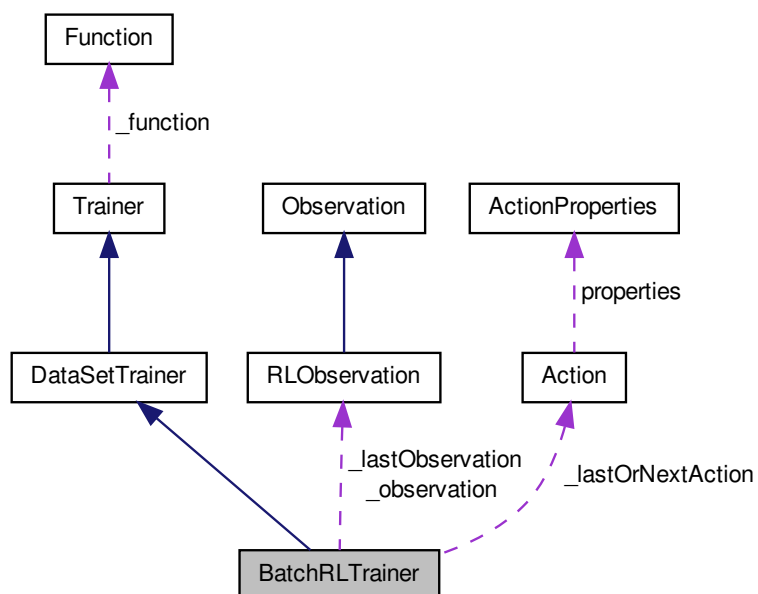
5.6 BatchRLTrainer Class Reference

```
#include <BatchRLTrainer.h>
```

Inheritance diagram for BatchRLTrainer:



Collaboration diagram for BatchRLTrainer:



Public Member Functions

- [BatchRLTrainer](#) ([QFunction](#) *qFunction, int [maxExamples](#), float [gamma](#), int [maxTrainingIterationsPerEpisode](#)=-1)
- virtual [~BatchRLTrainer](#) ()
- virtual void [init](#) ()
- virtual void [_doTrainEpisode](#) ([DataSet](#) *data)

This method should be overridden by subclasses to train a single episode over the dataset.

Public Attributes

- [real](#) * [targets](#)
- int [maxExamples](#)
- float [gamma](#)
- int [maxTrainingIterationsPerEpisode](#)
- [RLObservation](#) [_lastObservation](#)
- [Action](#) [_lastOrNextAction](#)
- [RLObservation](#) [_observation](#)

5.6.1 Detailed Description

This class implements the *Batch reinforcement learning algorithm*. It trains a [QFunction](#) on a [DataSet](#) containing (s, a, r, s') tuples.

Reference: S. Lange, T. Gabel and M. Riedmiller. *Batch Reinforcement Learning*. In Wiering, M. and van Otterlo, M., editor, Reinforcement Learning: State of the Art, Springer, in press, 2011

See Also

[TupleDataSet](#)

5.6.2 Constructor & Destructor Documentation

5.6.2.1 [BatchRLTrainer::BatchRLTrainer](#) ([QFunction](#) * *qFunction*, int *maxExamples*, float *gamma*, int *maxTrainingIterationsPerEpisode* = -1)

5.6.2.2 [BatchRLTrainer::~~BatchRLTrainer](#) () [virtual]

5.6.3 Member Function Documentation

5.6.3.1 void [BatchRLTrainer::_doTrainEpisode](#) ([DataSet](#) * *data*) [virtual]

This method should be overridden by subclasses to train a single episode over the dataset.

Implements [DataSetTrainer](#).

5.6.3.2 void [BatchRLTrainer::init](#) () [virtual]

Reimplemented from [Trainer](#).

5.6.4 Member Data Documentation

5.6.4.1 `RLObservation BatchRLTrainer::_lastObservation`

5.6.4.2 `Action BatchRLTrainer::_lastOrNextAction`

5.6.4.3 `RLObservation BatchRLTrainer::_observation`

5.6.4.4 `float BatchRLTrainer::gamma`

5.6.4.5 `int BatchRLTrainer::maxExamples`

5.6.4.6 `int BatchRLTrainer::maxTrainingIterationsPerEpisode`

5.6.4.7 `real* BatchRLTrainer::targets`

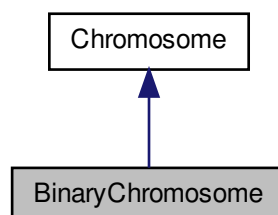
The documentation for this class was generated from the following files:

- `src/qualia/rl/`[BatchRLTrainer.h](#)
- `src/qualia/rl/`[BatchRLTrainer.cpp](#)

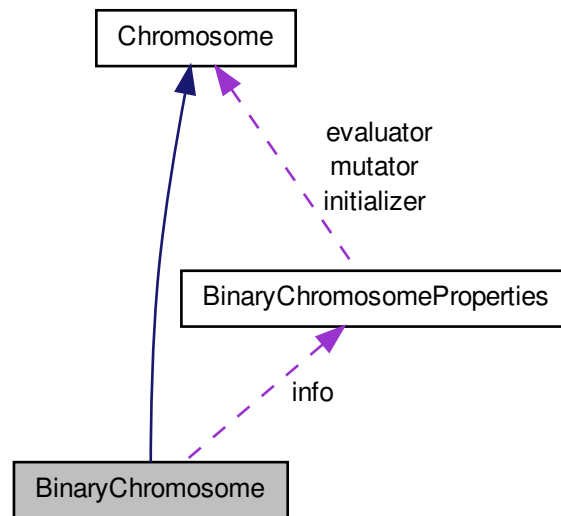
5.7 BinaryChromosome Class Reference

```
#include <BinaryChromosome.h>
```

Inheritance diagram for BinaryChromosome:



Collaboration diagram for BinaryChromosome:



Public Member Functions

- [BinaryChromosome](#) ([BinaryChromosomeProperties](#) *[info](#))
- virtual [~BinaryChromosome](#) ()
- virtual void [copyFrom](#) (const [Chromosome](#) &c)
- virtual void [init](#) ()
- virtual void [mutate](#) (float p)
- virtual bool [equals](#) (const [Chromosome](#) &c) const
- uint64_t [getGeneValue](#) (int gene) const

Returns the value of gene #gene# as a 64-bit integer.

- void [setGeneValue](#) (int gene, uint64_t value)

Sets the value of gene #gene# to #value# (max. 64-bit).

Static Public Member Functions

- static void [initializeRandom](#) ([Chromosome](#) &chromosome)
- static void [mutateFlip](#) ([Chromosome](#) &chromosome, float probability)
- static void [crossoverOnePoint](#) (const [Chromosome](#) &parent1, const [Chromosome](#) &parent2, [Chromosome](#) *offspring1, [Chromosome](#) *offspring2)
- static void [crossoverTwoPoint](#) (const [Chromosome](#) &parent1, const [Chromosome](#) &parent2, [Chromosome](#) *offspring1, [Chromosome](#) *offspring2)
- static void [_codeCrossoverOnePoint](#) (uint8_t *offspringCode, const uint8_t *parentCode1, const uint8_t *parentCode2, unsigned int point, unsigned int codeSize)
- static void [_codeCrossoverTwoPoint](#) (uint8_t *offspringCode, const uint8_t *parentCode1, const uint8_t *parentCode2, unsigned int point1, unsigned int point2, unsigned int codeSize)

Public Attributes

- [BinaryChromosomeProperties](#) * *info*
- `uint8_t` * *code*

5.7.1 Constructor & Destructor Documentation

5.7.1.1 `BinaryChromosome::BinaryChromosome (BinaryChromosomeProperties * info)`

5.7.1.2 `BinaryChromosome::~~BinaryChromosome ()` `[virtual]`

5.7.2 Member Function Documentation

5.7.2.1 `void BinaryChromosome::codeCrossoverOnePoint (uint8_t * offspringCode, const uint8_t * parentCode1, const uint8_t * parentCode2, unsigned int point, unsigned int codeSize)` `[static]`

5.7.2.2 `void BinaryChromosome::codeCrossoverTwoPoint (uint8_t * offspringCode, const uint8_t * parentCode1, const uint8_t * parentCode2, unsigned int point1, unsigned int point2, unsigned int codeSize)` `[static]`

5.7.2.3 `void BinaryChromosome::copyFrom (const Chromosome & c)` `[virtual]`

Reimplemented from [Chromosome](#).

5.7.2.4 `void BinaryChromosome::crossoverOnePoint (const Chromosome & parent1, const Chromosome & parent2, Chromosome * offspring1, Chromosome * offspring2)` `[static]`

5.7.2.5 `void BinaryChromosome::crossoverTwoPoint (const Chromosome & parent1, const Chromosome & parent2, Chromosome * offspring1, Chromosome * offspring2)` `[static]`

5.7.2.6 `bool BinaryChromosome::equals (const Chromosome & c)` `const` `[virtual]`

Implements [Chromosome](#).

5.7.2.7 `uint64_t BinaryChromosome::getGeneValue (int gene)` `const`

Returns the value of gene #gene# as a 64-bit integer.

5.7.2.8 `void BinaryChromosome::init ()` `[virtual]`

Reimplemented from [Chromosome](#).

5.7.2.9 `void BinaryChromosome::initializeRandom (Chromosome & chromosome)` `[static]`

5.7.2.10 `void BinaryChromosome::mutate (float p)` `[virtual]`

Reimplemented from [Chromosome](#).

5.7.2.11 `void BinaryChromosome::mutateFlip (Chromosome & chromosome, float probability)` `[static]`

5.7.2.12 `void BinaryChromosome::setGeneValue (int gene, uint64_t value)`

Sets the value of gene #gene# to #value# (max. 64-bit).

5.7.3 Member Data Documentation

5.7.3.1 `uint8_t*` BinaryChromosome::code

5.7.3.2 BinaryChromosomeProperties* BinaryChromosome::info

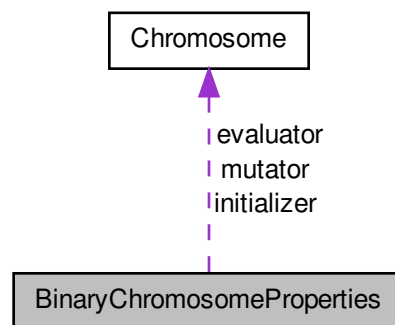
The documentation for this class was generated from the following files:

- `src/qualia/ga/BinaryChromosome.h`
- `src/qualia/ga/BinaryChromosome.cpp`

5.8 BinaryChromosomeProperties Class Reference

```
#include <BinaryChromosome.h>
```

Collaboration diagram for BinaryChromosomeProperties:



Public Member Functions

- `BinaryChromosomeProperties` (unsigned int `nGenes`, const `uint8_t *`geneSizes, `Initializer` initializer=0, `Mutator` mutator=0)
- unsigned int `nGenes` () const
Returns the number of genes in the chromosome.
- `uint8_t` `geneSize` (int i) const
Returns the size (in bits) of gene #i#.
- unsigned int `bitSize` () const
Returns the total size (in bits) of the chromosome.
- unsigned int `byteSize` () const
Returns the total size (in bytes) of the chromosome.
- int `getStartBitPosition` (int gene) const
Returns the bit at which gene #gene# starts.
- virtual bool `equals` (const `BinaryChromosomeProperties` &info)
Tests equality.

Public Attributes

- unsigned int [_nGenes](#)
The number of genes in the chromosome.
- uint8_t * [_geneSizes](#)
The size (in bits) of each gene.
- [Initializer](#) *initializer*
The initializer, mutator and evaluator functions.
- [Mutator](#) *mutator*
- [Evaluator](#) *evaluator*

5.8.1 Constructor & Destructor Documentation

5.8.1.1 **BinaryChromosomeProperties::BinaryChromosomeProperties** (unsigned int *nGenes*, const uint8_t * *geneSizes*, *Initializer initializer* = 0, *Mutator mutator* = 0)

5.8.2 Member Function Documentation

5.8.2.1 unsigned int **BinaryChromosomeProperties::bitSize** () const

Returns the total size (in bits) of the chromosome.

5.8.2.2 unsigned int **BinaryChromosomeProperties::byteSize** () const [inline]

Returns the total size (in bytes) of the chromosome.

5.8.2.3 bool **BinaryChromosomeProperties::equals** (const **BinaryChromosomeProperties** & *info*) [virtual]

Tests equality.

5.8.2.4 uint8_t **BinaryChromosomeProperties::geneSize** (int *i*) const [inline]

Returns the size (in bits) of gene #i#.

5.8.2.5 int **BinaryChromosomeProperties::getStartBitPosition** (int *gene*) const

Returns the bit at which gene #gene# starts.

5.8.2.6 unsigned int **BinaryChromosomeProperties::nGenes** () const [inline]

Returns the number of genes in the chromosome.

5.8.3 Member Data Documentation

5.8.3.1 uint8_t* **BinaryChromosomeProperties::_geneSizes**

The size (in bits) of each gene.

5.8.3.2 unsigned int **BinaryChromosomeProperties::_nGenes**

The number of genes in the chromosome.

5.8.3.3 Evaluator BinaryChromosomeProperties::evaluator

5.8.3.4 Initializer BinaryChromosomeProperties::initializer

The initializer, mutator and evaluator functions.

5.8.3.5 Mutator BinaryChromosomeProperties::mutator

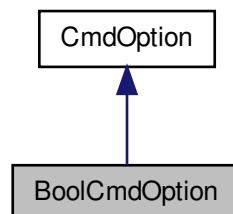
The documentation for this class was generated from the following files:

- [src/qualia/ga/BinaryChromosome.h](#)
- [src/qualia/ga/BinaryChromosome.cpp](#)

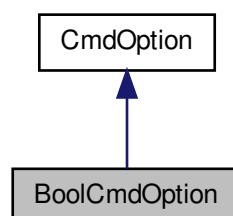
5.9 BoolCmdOption Class Reference

```
#include <CmdOption.h>
```

Inheritance diagram for BoolCmdOption:



Collaboration diagram for BoolCmdOption:



Public Member Functions

- [BoolCmdOption](#) (const char *name_, bool *ptr_, bool init_value_, const char *help_="", bool save_=false)

- virtual void [initValue](#) ()
Initialize the value of the option.
- virtual void [read](#) (int *argc_, char ***argv_)
- virtual void [load](#) ([DiskXFile](#) *file)
- virtual void [save](#) ([DiskXFile](#) *file)
- [~BoolCmdOption](#) ()

Public Attributes

- bool * [ptr](#)
- bool [init_value](#)

5.9.1 Detailed Description

This class defines a bool command-line option.

Author

Ronan Collobert (collober@idiap.ch)

See Also

[CmdLine](#)

5.9.2 Constructor & Destructor Documentation

5.9.2.1 [BoolCmdOption::BoolCmdOption](#) (const char * *name_*, bool * *ptr_*, bool *init_value_*, const char * *help_* = " ", bool *save_* = false)

5.9.2.2 [BoolCmdOption::~~BoolCmdOption](#) ()

5.9.3 Member Function Documentation

5.9.3.1 void [BoolCmdOption::initValue](#) () [virtual]

Initialize the value of the option.

Reimplemented from [CmdOption](#).

5.9.3.2 void [BoolCmdOption::load](#) ([DiskXFile](#) * *file*) [virtual]

Reimplemented from [CmdOption](#).

5.9.3.3 void [BoolCmdOption::read](#) (int * *argc_*, char *** *argv_*) [virtual]

Read the option on the command line. argv_ and argc_ have to point of the next option after that.

Reimplemented from [CmdOption](#).

5.9.3.4 void [BoolCmdOption::save](#) ([DiskXFile](#) * *file*) [virtual]

Reimplemented from [CmdOption](#).

5.9.4 Member Data Documentation

5.9.4.1 `bool BoolCmdOption::init_value`

5.9.4.2 `bool* BoolCmdOption::ptr`

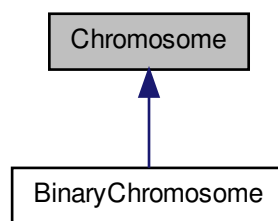
The documentation for this class was generated from the following files:

- `src/qualia/computer/CmdOption.h`
- `src/qualia/computer/CmdOption.cpp`

5.10 Chromosome Class Reference

```
#include <Chromosome.h>
```

Inheritance diagram for Chromosome:



Public Member Functions

- [Chromosome](#) ()
- virtual [~Chromosome](#) ()
- virtual void [copyFrom](#) (const [Chromosome](#) &c)
- virtual void [init](#) ()
- virtual void [mutate](#) (float p)
- virtual bool [equals](#) (const [Chromosome](#) &g) const =0

5.10.1 Constructor & Destructor Documentation

5.10.1.1 `Chromosome::Chromosome ()` [`inline`]

5.10.1.2 `virtual Chromosome::~~Chromosome ()` [`inline`], [`virtual`]

5.10.2 Member Function Documentation

5.10.2.1 `virtual void Chromosome::copyFrom (const Chromosome &c)` [`inline`], [`virtual`]

Reimplemented in [BinaryChromosome](#).

5.10.2.2 `virtual bool Chromosome::equals (const Chromosome & g) const` `[pure virtual]`

Implemented in [BinaryChromosome](#).

5.10.2.3 `virtual void Chromosome::init ()` `[inline],[virtual]`

Reimplemented in [BinaryChromosome](#).

5.10.2.4 `virtual void Chromosome::mutate (float p)` `[inline],[virtual]`

Reimplemented in [BinaryChromosome](#).

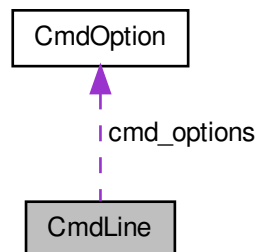
The documentation for this class was generated from the following file:

- `src/qualia/ga/Chromosome.h`

5.11 CmdLine Class Reference

```
#include <CmdLine.h>
```

Collaboration diagram for CmdLine:



Public Member Functions

- [CmdLine](#) ()
- `int read (int argc_, char **argv_)`
- `void help ()`
- `void addICmdOption (const char *name, int *ptr, int init_value, const char *help="", bool save_it=false)`
- `void addBCmdOption (const char *name, bool *ptr, bool init_value, const char *help="", bool save_it=false)`
- `void addRCmdOption (const char *name, real *ptr, real init_value, const char *help="", bool save_it=false)`
- `void addSCmdOption (const char *name, char **ptr, const char *init_value, const char *help="", bool save_it=false)`
- `void addICmdArg (const char *name, int *ptr, const char *help="", bool save_it=false)`
- `void addBCmdArg (const char *name, bool *ptr, const char *help="", bool save_it=false)`
- `void addRCmdArg (const char *name, real *ptr, const char *help="", bool save_it=false)`
- `void addSCmdArg (const char *name, char **ptr, const char *help="", bool save_it=false)`
- `void addText (const char *text)`
Add a text line in the help message.

- void [info](#) (const char *text)
Add a text at the beginnig of the help.
- void [addMasterSwitch](#) (const char *text)
- void [setWorkingDirectory](#) (const char *dirname)
- char * [getPath](#) (const char *filename)
- [DiskXFile](#) * [getXFile](#) (const char *filename)
- virtual void [load](#) ([DiskXFile](#) *file)
Load the object from a file pointer ({not the options})
- virtual void [save](#) ([DiskXFile](#) *file)
Save the object to a file pointer ({not the options})
- void [addCmdOption](#) ([CmdOption](#) *option)
- void [writeLog](#) ([DiskXFile](#) *file, bool write_associated_files)
- virtual [~CmdLine](#) ()

Public Attributes

- char * [program_name](#)
- bool [write_log](#)
- int [n_master_switches](#)
- int * [n_cmd_options](#)
- [CmdOption](#) *** [cmd_options](#)
- char * [text_info](#)
- char * [working_directory](#)
- char ** [associated_files](#)
- int [n_associated_files](#)
- int [master_switch](#)
- char ** [argv](#)
- int [argc](#)

5.11.1 Detailed Description

This class provides a useful interface for the user, to easily read some arguments/options from the command-line.

Note that here, we make a difference between: {itemize} { options} which are not required. { arguments} which are required. {itemize}

Options: {tabular}{|c|} "write log" & bool & Should I output the cmd.log file ? & [true] {tabular}

Author

Ronan Collobert (collober@idiap.ch)

See Also

[CmdOption](#)

5.11.2 Constructor & Destructor Documentation

5.11.2.1 [CmdLine::CmdLine](#) ()

5.11.2.2 [CmdLine::~~CmdLine](#) () [\[virtual\]](#)

5.11.3 Member Function Documentation

5.11.3.1 `void CmdLine::addBCmdArg (const char * name, bool * ptr, const char * help = " ", bool save_it = false)`

5.11.3.2 `void CmdLine::addBCmdOption (const char * name, bool * ptr, bool init_value, const char * help = " ", bool save_it = false)`

5.11.3.3 `void CmdLine::addCmdOption (CmdOption * option)`

Add an option to the command line. Use this method if the wrappers that are provided are not sufficient.

5.11.3.4 `void CmdLine::addICmdArg (const char * name, int * ptr, const char * help = " ", bool save_it = false)`

Functions for adding an argument. The argument will be setted to #value# in the command-line by writing "#value#" { after } all the options. If there are N arguments, you have to write "#value1# #value2# #value3# ... #valueN#" to set them in the command-line.

5.11.3.5 `void CmdLine::addICmdOption (const char * name, int * ptr, int init_value, const char * help = " ", bool save_it = false)`

Functions for adding options. The calling order of the following functions will define the text order associated when you will call [help\(\)](#)).

Add an option (Int, Bool, Real, String). {itemize} #name# the name of the option (must be unique). #ptr# is the pointer on the optional variable. #init_value# is the initialization value. [help](#)# is the help text for this option. {itemize}

The option will be setted to #value# in the command-line by printing "#name# #value#"

5.11.3.6 `void CmdLine::addMasterSwitch (const char * text)`

Add a master switch. It creates an another type of command line. If the #text# is the first argument of the user command line, only the options corresponding to this new command line will be considered.

5.11.3.7 `void CmdLine::addRCmdArg (const char * name, real * ptr, const char * help = " ", bool save_it = false)`

5.11.3.8 `void CmdLine::addRCmdOption (const char * name, real * ptr, real init_value, const char * help = " ", bool save_it = false)`

5.11.3.9 `void CmdLine::addSCmdArg (const char * name, char ** ptr, const char * help = " ", bool save_it = false)`

5.11.3.10 `void CmdLine::addSCmdOption (const char * name, char ** ptr, const char * init_value, const char * help = " ", bool save_it = false)`

5.11.3.11 `void CmdLine::addText (const char * text)`

Add a text line in the help message.

5.11.3.12 `char * CmdLine::getPath (const char * filename)`

Get a full path. It adds the [working_directory](#)# before the #filename#. This path will be deleted by [CmdLine](#).

5.11.3.13 `DiskXFile * CmdLine::getXFile (const char * filename)`

Get a [DiskXFile](#). It adds the [working_directory](#)# before the #filename# and opens the file. This [XFile](#) will be deleted by [CmdLine](#).

5.11.3.14 void CmdLine::help ()

[Print](#) the help. Call this function { after} adding options/arguments that you need, with the help of the following functions.

5.11.3.15 void CmdLine::info (const char * text)

Add a text at the beginnig of the help.

5.11.3.16 void CmdLine::load (DiskXFile * file) [virtual]

Load the object from a file pointer ({not the options})

5.11.3.17 int CmdLine::read (int argc_, char ** argv_)

Read the command-line. Call this function { after} adding options/arguments that you need, with the help of the following functions.

5.11.3.18 void CmdLine::save (DiskXFile * file) [virtual]

Save the object to a file pointer ({not the options})

5.11.3.19 void CmdLine::setWorkingDirectory (const char * dirname)

Set the working directory. Use it with [getPath\(\)](#)# and [getXFile\(\)](#)#.

5.11.3.20 void CmdLine::writeLog (DiskXFile * file, bool write_associated_files)

Write a log in #file#. If desired, the associated files can be printed.

5.11.4 Member Data Documentation

5.11.4.1 int CmdLine::argc

5.11.4.2 char** CmdLine::argv

5.11.4.3 char** CmdLine::associated_files

5.11.4.4 CmdOption*** CmdLine::cmd_options

5.11.4.5 int CmdLine::master_switch

5.11.4.6 int CmdLine::n_associated_files

5.11.4.7 int* CmdLine::n_cmd_options

5.11.4.8 int CmdLine::n_master_switches

5.11.4.9 char* CmdLine::program_name

5.11.4.10 char* CmdLine::text_info

5.11.4.11 `char* CmdLine::working_directory`

5.11.4.12 `bool CmdLine::write_log`

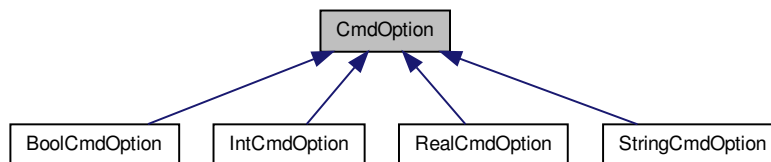
The documentation for this class was generated from the following files:

- [src/qualia/computer/CmdLine.h](#)
- [src/qualia/computer/CmdLine.cpp](#)

5.12 CmdOption Class Reference

```
#include <CmdOption.h>
```

Inheritance diagram for CmdOption:



Public Member Functions

- [CmdOption](#) (const char *name_, const char *type_name_, const char *help_="", bool save_[=false](#))
- virtual void [initValue](#) ()
Initialize the value of the option.
- virtual void [printValue](#) (DiskXFile *file_)
If [is_setted](#) is true, print the current value, else the init value.
- virtual void [read](#) (int *argc_, char ***argv_)
- bool [isCurrent](#) (int *argc_, char ***argv_)
- bool [isOption](#) (bool set_[=false](#))
- bool [isArgument](#) (bool set_[=false](#))
- bool [isText](#) (bool set_[=false](#))
- bool [isMasterSwitch](#) (bool set_[=false](#))
- virtual void [load](#) (DiskXFile *file)
- virtual void [save](#) (DiskXFile *file)
- virtual [~CmdOption](#) ()

Public Attributes

- char * [name](#)
Name of the option.
- char * [type_name](#)
Type name of the option.
- char * [help](#)
- bool [needsSave](#)
- bool [is_setted](#)

Private Attributes

- bool [is_option](#)
- bool [is_argument](#)
- bool [is_text](#)
- bool [is_master_switch](#)

5.12.1 Detailed Description

This class defines an option for the command line. If you need special command line arguments/options, you have to create a new children of this class.

Author

Ronan Collobert (collober@idiap.ch)

See Also

[CmdLine](#)

5.12.2 Constructor & Destructor Documentation

5.12.2.1 `CmdOption::CmdOption (const char * name_, const char * type_name_, const char * help_ = "", bool save_ = false)`

5.12.2.2 `CmdOption::~~CmdOption () [virtual]`

5.12.3 Member Function Documentation

5.12.3.1 `void CmdOption::initValue () [virtual]`

Initialize the value of the option.

Reimplemented in [StringCmdOption](#), [BoolCmdOption](#), [RealCmdOption](#), and [IntCmdOption](#).

5.12.3.2 `bool CmdOption::isArgument (bool set_ = false)`

Returns true if it's a required argument. If #set_# is true, set it to a required argument.

5.12.3.3 `bool CmdOption::isCurrent (int * argc_, char *** argv_)`

5.12.3.4 `bool CmdOption::isMasterSwitch (bool set_ = false)`

Returns true if it's a master switch. If #set_# is true, set it to a master switch.

5.12.3.5 `bool CmdOption::isOption (bool set_ = false)`

Returns true if it's an optional argument. If #set_# is true, set it to an optional argument.

5.12.3.6 `bool CmdOption::isText (bool set_ = false)`

Returns true if it's just text to be displayed in the command line. If #set_# is true, set it to text mode.

5.12.3.7 `virtual void CmdOption::load (DiskXFile * file) [inline],[virtual]`

Reimplemented in [StringCmdOption](#), [BoolCmdOption](#), [RealCmdOption](#), and [IntCmdOption](#).

5.12.3.8 `void CmdOption::printValue (DiskXFile * file_) [virtual]`

If `is_setted#` is true, print the current value, else the init value.

Reimplemented in [StringCmdOption](#), [RealCmdOption](#), and [IntCmdOption](#).

5.12.3.9 `void CmdOption::read (int * argc_, char *** argv_) [virtual]`

Read the option on the command line. `argv_` and `argc_` have to point of the next option after that.

Reimplemented in [StringCmdOption](#), [BoolCmdOption](#), [RealCmdOption](#), and [IntCmdOption](#).

5.12.3.10 `virtual void CmdOption::save (DiskXFile * file) [inline],[virtual]`

Reimplemented in [StringCmdOption](#), [BoolCmdOption](#), [RealCmdOption](#), and [IntCmdOption](#).

5.12.4 Member Data Documentation

5.12.4.1 `char* CmdOption::help`

An help string. Cannot be NULL.

5.12.4.2 `bool CmdOption::is_argument [private]`

5.12.4.3 `bool CmdOption::is_master_switch [private]`

5.12.4.4 `bool CmdOption::is_option [private]`

5.12.4.5 `bool CmdOption::is_setted`

True is the option has been setted after reading the command-line.

5.12.4.6 `bool CmdOption::is_text [private]`

5.12.4.7 `char* CmdOption::name`

Name of the option.

5.12.4.8 `bool CmdOption::needsSave`

True is the option has to be saved when saving the command line.

5.12.4.9 `char* CmdOption::type_name`

Type name of the option.

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

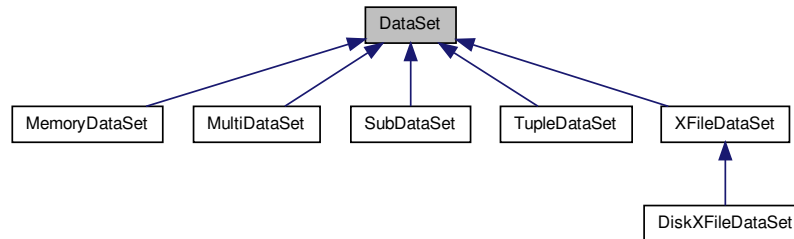
- [src/qualia/computer/CmdOption.h](#)

- [src/qualia/computer/CmdOption.cpp](#)

5.13 DataSet Class Reference

```
#include <DataSet.h>
```

Inheritance diagram for DataSet:



Public Member Functions

- [DataSet](#) (int [nExamples](#)=0, int [dim](#)=0)
Constructor.
- virtual [~DataSet](#) ()
- int [dim](#) () const
Returns the dimension of each example.
- int [nExamples](#) () const
Returns the number of examples.
- virtual void [init](#) ()
- virtual void [reset](#) ()
To be called before every episode.
- virtual void [setExample](#) (int t)=0
Sets this->example to given index t.

Public Attributes

- int [_nExamples](#)
Number of examples in the dataset.
- int [_dim](#)
Dimension of the example.
- [real](#) * [example](#)
Current example.

5.13.1 Detailed Description

Provides an interface to manipulate all kind of data.

5.13.2 Constructor & Destructor Documentation

5.13.2.1 `DataSet::DataSet (int nExamples = 0, int dim = 0)`

Constructor.

5.13.2.2 `DataSet::~~DataSet () [virtual]`

5.13.3 Member Function Documentation

5.13.3.1 `int DataSet::dim () const [inline]`

Returns the dimension of each example.

5.13.3.2 `void DataSet::init () [virtual]`

To be called before training. Should allocate example, among other things. Default version allocates example to the size of *dim*.

Reimplemented in [TupleDataSet](#), [XFileDataSet](#), [MultiDataSet](#), [DiskXFileDataSet](#), and [SubDataSet](#).

5.13.3.3 `int DataSet::nExamples () const [inline]`

Returns the number of examples.

5.13.3.4 `virtual void DataSet::reset () [inline], [virtual]`

To be called before every episode.

Reimplemented in [TupleDataSet](#), [MultiDataSet](#), [XFileDataSet](#), [DiskXFileDataSet](#), and [SubDataSet](#).

5.13.3.5 `virtual void DataSet::setExample (int t) [pure virtual]`

Sets this->example to given index *t*.

Implemented in [TupleDataSet](#), [MultiDataSet](#), [XFileDataSet](#), [SubDataSet](#), [DiskXFileDataSet](#), and [MemoryDataSet](#).

5.13.4 Member Data Documentation

5.13.4.1 `int DataSet::_dim`

Dimension of the example.

5.13.4.2 `int DataSet::_nExamples`

Number of examples in the dataset.

5.13.4.3 `real* DataSet::example`

Current example.

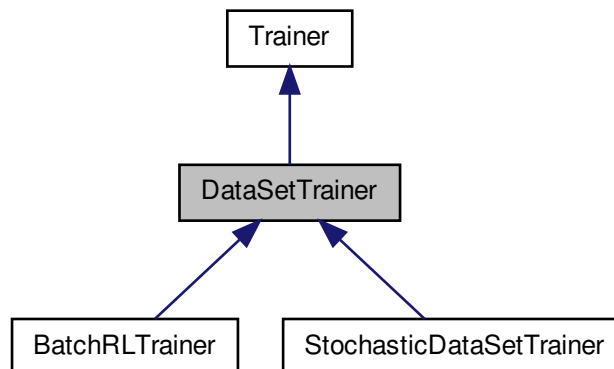
The documentation for this class was generated from the following files:

- [src/qualia/learning/DataSet.h](#)
- [src/qualia/learning/DataSet.cpp](#)

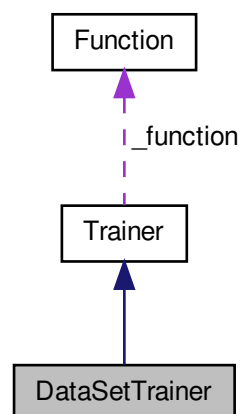
5.14 DataSetTrainer Class Reference

```
#include <DataSetTrainer.h>
```

Inheritance diagram for DataSetTrainer:



Collaboration diagram for DataSetTrainer:



Public Member Functions

- [DataSetTrainer](#) ([Function](#) *function)

Constructor.

- virtual `~DataSetTrainer()`
- virtual void `train(DataSet *data, int maxEpisodes=10)`
- virtual void `trainEpisode(DataSet *data)`

Trains one episode over the dataset.

- virtual bool `stop()`
- virtual void `_doTrainEpisode(DataSet *data)=0`

This method should be overridden by subclasses to train a single episode over the dataset.

Additional Inherited Members

5.14.1 Detailed Description

Trains a function over a dataset.

5.14.2 Constructor & Destructor Documentation

5.14.2.1 DataSetTrainer::DataSetTrainer (Function * function)

Constructor.

5.14.2.2 DataSetTrainer::~~DataSetTrainer () [virtual]

5.14.3 Member Function Documentation

5.14.3.1 virtual void DataSetTrainer::_doTrainEpisode (DataSet * data) [pure virtual]

This method should be overridden by subclasses to train a single episode over the dataset.

Implemented in [BatchRLTrainer](#), and [StochasticDataSetTrainer](#).

5.14.3.2 virtual bool DataSetTrainer::stop () [inline],[virtual]

Should return true iff the training is ended (can be used to provide early-stopping capability). This method is meant to be overridden by subclasses.

5.14.3.3 void DataSetTrainer::train (DataSet * data, int maxEpisodes = 10) [virtual]

Trains the function over a dataset. Training will stop when the `stop()` method returns true OR when the number of episodes reaches `maxEpisodes`. If `maxEpisodes <= 0` is specified then it will not be considered (ie. it will only stop when `stop()` returns true).

5.14.3.4 void DataSetTrainer::trainEpisode (DataSet * data) [virtual]

Trains one episode over the dataset.

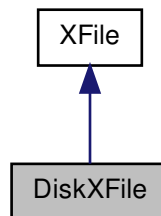
The documentation for this class was generated from the following files:

- [src/qualia/learning/DataSetTrainer.h](#)
- [src/qualia/learning/DataSetTrainer.cpp](#)

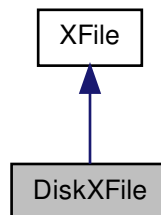
5.15 DiskXFile Class Reference

```
#include <DiskXFile.h>
```

Inheritance diagram for DiskXFile:



Collaboration diagram for DiskXFile:



Public Member Functions

- [DiskXFile](#) (const char *file_name, const char *open_flags)
Open "file_name" with the flags #open_flags#.
- [DiskXFile](#) (FILE *file_)
Use the given file...
- virtual int [read](#) (void *ptr, int block_size, int n_blocks)
- virtual int [write](#) (const void *ptr, int block_size, int n_blocks)
- virtual int [eof](#) ()
Are we at the end ?
- virtual int [flush](#) ()
Flush the file.
- virtual int [seek](#) (long offset, int whence)
Seek.
- virtual long [tell](#) ()
Tell me where am I...
- virtual void [rewind](#) ()

Rewind.

- virtual int `printf` (const char *format,...)
- virtual int `scanf` (const char *format, void *ptr)
- virtual char * `gets` (char *dest, int size_)

Print some text.

- virtual `~DiskXFile` ()

Static Public Member Functions

- static bool `isLittleEndianProcessor` ()
Returns true# if the processor uses the little endian coding format.
- static bool `isBigEndianProcessor` ()
Returns true# if the processor uses the big endian coding format.
- static bool `isNativeMode` ()
Returns true# if we'll load/save using the native mode.
- static void `setNativeMode` ()
- static void `setLittleEndianMode` ()
- static void `setBigEndianMode` ()

Public Attributes

- FILE * `file`
- bool `is_opened`
- bool `its_a_pipe`

Private Member Functions

- void `reverseMemory` (void *ptr_, int block_size, int n_blocks)

Private Attributes

- char * `buffer_block`
- int `buffer_block_size`

Static Private Attributes

- static bool `is_native_mode` = true

5.15.1 Detailed Description

A file on the disk.

Author

Ronan Collobert (collober@idiap.ch)

5.15.2 Constructor & Destructor Documentation

5.15.2.1 `DiskXFile::DiskXFile (const char * file_name, const char * open_flags)`

Open "file_name" with the flags #open_flags#.

5.15.2.2 DiskXFile::DiskXFile (FILE * *file_*)

Use the given file...

5.15.2.3 DiskXFile::~~DiskXFile () [virtual]

5.15.3 Member Function Documentation

5.15.3.1 int DiskXFile::eof () [virtual]

Are we at the end ?

Implements [XFile](#).

5.15.3.2 int DiskXFile::flush () [virtual]

Flush the file.

Implements [XFile](#).

5.15.3.3 char * DiskXFile::gets (char * *dest*, int *size_*) [virtual]

[Print](#) some text.

Scan some text. Get one line (read at most *#size_#* characters).

Implements [XFile](#).

5.15.3.4 bool DiskXFile::isBigEndianProcessor () [static]

Returns [true#](#) if the processor uses the big endian coding format.

5.15.3.5 bool DiskXFile::isLittleEndianProcessor () [static]

Returns [true#](#) if the processor uses the little endian coding format.

5.15.3.6 bool DiskXFile::isNativeMode () [static]

Returns [true#](#) if we'll load/save using the native mode.

5.15.3.7 int DiskXFile::printf (const char * *format*, ...) [virtual]5.15.3.8 int DiskXFile::read (void * *ptr*, int *block_size*, int *n_blocks*) [virtual]

Read something. Returns the number of blocks read or a value < 0 if there was an error.

Implements [XFile](#).

5.15.3.9 void DiskXFile::reverseMemory (void * *ptr_*, int *block_size*, int *n_blocks*) [private]

5.15.3.10 void DiskXFile::rewind () [virtual]

Rewind.

Implements [XFile](#).

5.15.3.11 `int DiskXFile::scanf (const char * format, void * ptr)` [virtual]

5.15.3.12 `int DiskXFile::seek (long offset, int whence)` [virtual]

Seek.

Implements [XFile](#).

5.15.3.13 `void DiskXFile::setBigEndianMode ()` [static]

We'll load/save using big endian mode. It means that if the computer doesn't use Big Endian, data will be converted.

5.15.3.14 `void DiskXFile::setLittleEndianMode ()` [static]

We'll load/save using little endian mode. It means that if the computer doesn't use Little Endian, data will be converted.

5.15.3.15 `void DiskXFile::setNativeMode ()` [static]

We'll load/save using native mode. We use little endian iff the computer uses little endian. We use big endian iff the computer uses big endian.

5.15.3.16 `long DiskXFile::tell ()` [virtual]

Tell me where am I...

Implements [XFile](#).

5.15.3.17 `int DiskXFile::write (const void * ptr, int block_size, int n_blocks)` [virtual]

Write. Returns the number of blocks written or a value < 0 if there was an error.

Implements [XFile](#).

5.15.4 Member Data Documentation

5.15.4.1 `char* DiskXFile::buffer_block` [private]

5.15.4.2 `int DiskXFile::buffer_block_size` [private]

5.15.4.3 `FILE* DiskXFile::file`

5.15.4.4 `bool DiskXFile::is_native_mode = true` [static], [private]

5.15.4.5 `bool DiskXFile::is_opened`

5.15.4.6 `bool DiskXFile::its_a_pipe`

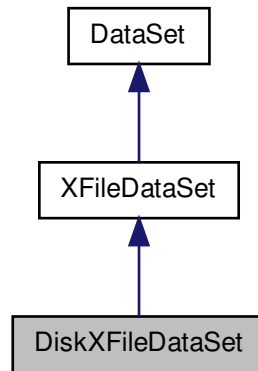
The documentation for this class was generated from the following files:

- [src/qualia/computer/DiskXFile.h](#)
- [src/qualia/computer/DiskXFile.cpp](#)

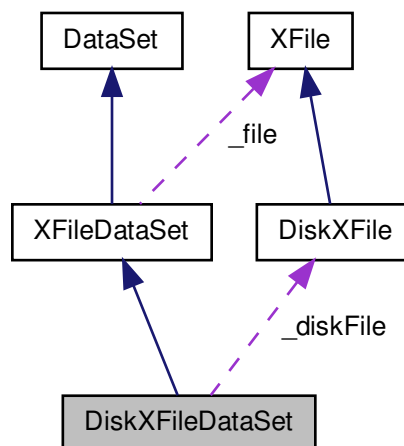
5.16 DiskXFileDataSet Class Reference

```
#include <DiskXFileDataSet.h>
```

Inheritance diagram for DiskXFileDataSet:



Collaboration diagram for DiskXFileDataSet:



Public Member Functions

- `DiskXFileDataSet` (`DiskXFile` *diskFile, bool useAscii=false)
- virtual `~DiskXFileDataSet` ()
- virtual void `init` ()
- virtual void `reset` ()

To be called before every episode.

- virtual void [setExample](#) (int t)

Sets this->example to given index t.

Public Attributes

- [DiskXFile](#) * [_diskFile](#)

The [DiskXFile](#) where the dataset is stored.

- bool [ascii](#)

Ascii or binary mode.

5.16.1 Constructor & Destructor Documentation

5.16.1.1 [DiskXFileDataSet::DiskXFileDataSet](#) ([DiskXFile](#) * *diskFile*, bool *useAscii* = false)

5.16.1.2 [DiskXFileDataSet::~DiskXFileDataSet](#) () [virtual]

5.16.2 Member Function Documentation

5.16.2.1 void [DiskXFileDataSet::init](#) () [virtual]

To be called before training. Should allocate example, among other things. Default version allocates example to the size of *dim*.

Reimplemented from [XFileDataSet](#).

5.16.2.2 void [DiskXFileDataSet::reset](#) () [virtual]

To be called before every episode.

Reimplemented from [XFileDataSet](#).

5.16.2.3 void [DiskXFileDataSet::setExample](#) (int t) [virtual]

Sets this->example to given index *t*.

Reimplemented from [XFileDataSet](#).

5.16.3 Member Data Documentation

5.16.3.1 [DiskXFile](#)* [DiskXFileDataSet::_diskFile](#)

The [DiskXFile](#) where the dataset is stored.

5.16.3.2 bool [DiskXFileDataSet::ascii](#)

Ascii or binary mode.

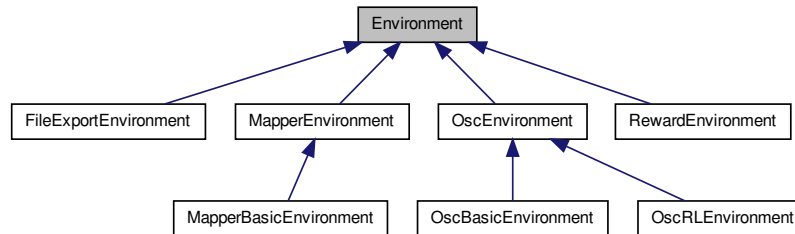
The documentation for this class was generated from the following files:

- src/qualia/computer/[DiskXFileDataSet.h](#)
- src/qualia/computer/[DiskXFileDataSet.cpp](#)

5.17 Environment Class Reference

```
#include <Environment.h>
```

Inheritance diagram for Environment:



Public Member Functions

- virtual [~Environment](#) ()
- virtual void [init](#) ()
Initializes the environment.
- virtual [Observation](#) * [start](#) ()=0
Sends the first observation.
- virtual [Observation](#) * [step](#) (const [Action](#) *action)=0
Performs action action and returns observation.

5.17.1 Detailed Description

Class interface for an environment in the [Qualia](#) framework. An environment is a component that gives back observations in response to actions. In [Qualia](#), environments typically implement the platform-specific behaviors of actually "doing" the action (eg. by asking a microcontroller to send a signal to one of its pins) and "reading" the observation (eg. by getting them from a sensor).

5.17.2 Constructor & Destructor Documentation

5.17.2.1 virtual [Environment::~Environment](#) () [[inline](#)], [[virtual](#)]

5.17.3 Member Function Documentation

5.17.3.1 virtual void [Environment::init](#) () [[inline](#)], [[virtual](#)]

Initializes the environment.

Reimplemented in [OscEnvironment](#), [FileExportEnvironment](#), and [MapperEnvironment](#).

5.17.3.2 virtual [Observation](#)* [Environment::start](#) () [[pure virtual](#)]

Sends the first observation.

Implemented in [OscEnvironment](#), [FileExportEnvironment](#), [MapperEnvironment](#), and [RewardEnvironment](#).

5.17.3.3 `virtual Observation* Environment::step (const Action * action) [pure virtual]`

Performs action *action* and returns observation.

Implemented in [OscEnvironment](#), [FileExportEnvironment](#), [MapperEnvironment](#), and [RewardEnvironment](#).

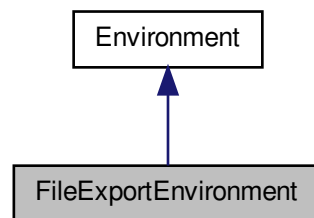
The documentation for this class was generated from the following file:

- `src/qualia/core/Environment.h`

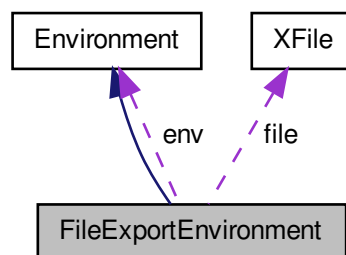
5.18 FileExportEnvironment Class Reference

```
#include <FileExportEnvironment.h>
```

Inheritance diagram for FileExportEnvironment:



Collaboration diagram for FileExportEnvironment:



Public Member Functions

- `FileExportEnvironment (Environment *env, XFile *file, unsigned int observationDim, unsigned int actionDim)`
- `virtual ~FileExportEnvironment ()`
- `virtual void init ()`

Initializes the environment.

- virtual [Observation](#) * [start](#) ()
Sends the first observation.
- virtual [Observation](#) * [step](#) (const [Action](#) *action)
Performs action action and returns observation.

Public Attributes

- [Environment](#) * [env](#)
- [XFile](#) * [file](#)
- unsigned int [observationDim](#)
- unsigned int [actionDim](#)

5.18.1 Detailed Description

Generates a file of data in the [Qualia](#) RAW format: <observation_dim><action_dim><o0><a0><01><a1>...<a{N-1}><oN>

- observation_dim : dimension of observations
- action_dim : dimension of actions
- o0 .. oN : observations (each one can be read by calling [Observation::readData\(\)](#) or appropriate subclass)
- a0 .. a{N-1} : actions (each one can be read by calling [Action::readData\(\)](#) or appropriate subclass)

5.18.2 Constructor & Destructor Documentation

5.18.2.1 [FileExportEnvironment::FileExportEnvironment](#) ([Environment](#) * *env*, [XFile](#) * *file*, unsigned int *observationDim*, unsigned int *actionDim*)

5.18.2.2 [FileExportEnvironment::~FileExportEnvironment](#) () [[virtual](#)]

5.18.3 Member Function Documentation

5.18.3.1 void [FileExportEnvironment::init](#) () [[virtual](#)]

Initializes the environment.

Reimplemented from [Environment](#).

5.18.3.2 [Observation](#) * [FileExportEnvironment::start](#) () [[virtual](#)]

Sends the first observation.

Implements [Environment](#).

5.18.3.3 [Observation](#) * [FileExportEnvironment::step](#) (const [Action](#) * *action*) [[virtual](#)]

Performs action *action* and returns observation.

Implements [Environment](#).

5.18.4 Member Data Documentation

5.18.4.1 unsigned int FileExportEnvironment::actionDim

5.18.4.2 Environment* FileExportEnvironment::env

5.18.4.3 XFile* FileExportEnvironment::file

5.18.4.4 unsigned int FileExportEnvironment::observationDim

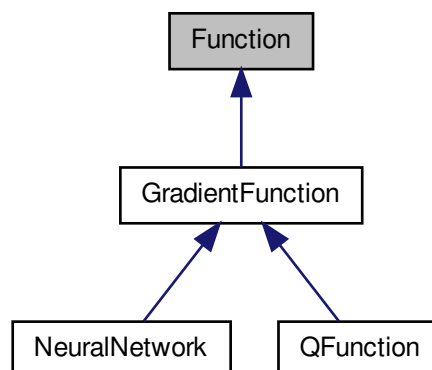
The documentation for this class was generated from the following files:

- [src/qualia/core/FileExportEnvironment.h](#)
- [src/qualia/core/FileExportEnvironment.cpp](#)

5.19 Function Class Reference

```
#include <Function.h>
```

Inheritance diagram for Function:



Public Member Functions

- [Function](#) ()
- virtual [~Function](#) ()
- virtual void [init](#) ()
Initializes the function.
- virtual unsigned [nInputs](#) () const =0
Returns the number of inputs.
- virtual unsigned [nOutputs](#) () const =0
Returns the number of outputs.
- virtual void [setInputs](#) (const [real](#) *input)
Sets the value of the inputs.
- virtual void [getOutputs](#) ([real](#) *output) const

- Get the value of the outputs.*
- virtual void [setInput](#) (int i, [real](#) x)=0
- Sets input i to value x.*
- virtual float [getOutput](#) (int i) const =0
- Get output i.*
- virtual void [propagate](#) ()=0
- Propagates inputs to outputs.*
- virtual void [save](#) ([XFile](#) *file)=0
- Saves to file.*
- virtual void [load](#) ([XFile](#) *file)=0
- Loads from file.*

5.19.1 Detailed Description

An abstract class for a feed-forward function, such as a [NeuralNetwork](#).

5.19.2 Constructor & Destructor Documentation

5.19.2.1 [Function::Function](#) () `[inline]`

5.19.2.2 [virtual Function::~~Function](#) () `[inline],[virtual]`

5.19.3 Member Function Documentation

5.19.3.1 [virtual float Function::getOutput](#) (int i) const `[pure virtual]`

Get output *i*.

Implemented in [NeuralNetwork](#), and [QFunction](#).

5.19.3.2 [void Function::getOutputs](#) ([real](#) * *output*) const `[virtual]`

Get the value of the outputs.

Reimplemented in [NeuralNetwork](#).

5.19.3.3 [virtual void Function::init](#) () `[inline],[virtual]`

Initializes the function.

Reimplemented in [NeuralNetwork](#), and [QFunction](#).

5.19.3.4 [virtual void Function::load](#) ([XFile](#) * *file*) `[pure virtual]`

Loads from file.

Implemented in [NeuralNetwork](#), [GradientFunction](#), and [QFunction](#).

5.19.3.5 [virtual unsigned Function::nInputs](#) () const `[pure virtual]`

Returns the number of inputs.

Implemented in [NeuralNetwork](#), and [QFunction](#).

5.19.3.6 `virtual unsigned Function::nOutputs () const` [pure virtual]

Returns the number of outputs.

Implemented in [NeuralNetwork](#), and [QFunction](#).

5.19.3.7 `virtual void Function::propagate ()` [pure virtual]

Propagates inputs to outputs.

Implemented in [NeuralNetwork](#), and [QFunction](#).

5.19.3.8 `virtual void Function::save (XFile * file)` [pure virtual]

Saves to file.

Implemented in [NeuralNetwork](#), [GradientFunction](#), and [QFunction](#).

5.19.3.9 `virtual void Function::setInput (int i, real x)` [pure virtual]

Sets input *i* to value *x*.

Implemented in [NeuralNetwork](#), and [QFunction](#).

5.19.3.10 `void Function::setInputs (const real * input)` [virtual]

Sets the value of the inputs.

Reimplemented in [NeuralNetwork](#).

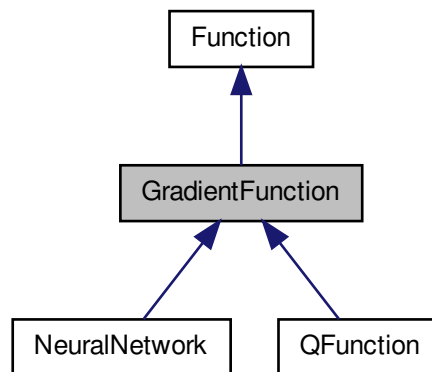
The documentation for this class was generated from the following files:

- [src/qualia/learning/Function.h](#)
- [src/qualia/learning/Function.cpp](#)

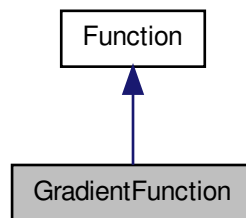
5.20 GradientFunction Class Reference

```
#include <GradientFunction.h>
```

Inheritance diagram for GradientFunction:



Collaboration diagram for GradientFunction:



Public Member Functions

- [GradientFunction](#) ()
- virtual [~GradientFunction](#) ()
- virtual void [clearDelta](#) ()
Clears the derivatives.
- virtual unsigned int [nParams](#) () const =0
Returns the number of parameters.
- virtual void [backpropagate](#) (real *outputError)=0
Backpropagates the error, updating the derivatives.
- virtual void [update](#) ()=0
Updates the weights according to the derivatives.
- virtual void [save](#) (XFile *file)
Saves the model to a file.
- virtual void [load](#) (XFile *file)
Loads the model from a file.

Public Attributes

- [real * weights](#)
The weights (parameters) of the gradient function.
- [real * dWeights](#)
The derivatives of the weights.

5.20.1 Detailed Description

Abstract class for gradient functions, such as a [NeuralNetwork](#). A [GradientFunction](#) has a set of parameters (weights) and error derivatives. It can back-propagate the errors to compute the derivatives and update its weights accordingly.

5.20.2 Constructor & Destructor Documentation

5.20.2.1 `GradientFunction::GradientFunction ()` `[inline]`

5.20.2.2 `virtual GradientFunction::~~GradientFunction ()` `[inline],[virtual]`

5.20.3 Member Function Documentation

5.20.3.1 `virtual void GradientFunction::backpropagate (real * outputError)` `[pure virtual]`

Backpropagates the error, updating the derivatives.

Implemented in [NeuralNetwork](#), and [QFunction](#).

5.20.3.2 `virtual void GradientFunction::clearDelta ()` `[inline],[virtual]`

Clears the derivatives.

5.20.3.3 `virtual void GradientFunction::load (XFile * file)` `[inline],[virtual]`

Loads the model from a file.

Implements [Function](#).

Reimplemented in [NeuralNetwork](#), and [QFunction](#).

5.20.3.4 `virtual unsigned int GradientFunction::nParams () const` `[pure virtual]`

Returns the number of parameters.

Implemented in [NeuralNetwork](#), and [QFunction](#).

5.20.3.5 `virtual void GradientFunction::save (XFile * file)` `[inline],[virtual]`

Saves the model to a file.

Implements [Function](#).

Reimplemented in [NeuralNetwork](#), and [QFunction](#).

5.20.3.6 virtual void GradientFunction::update () [pure virtual]

Updates the weights according to the derivatives.

Implemented in [NeuralNetwork](#), and [QFunction](#).

5.20.4 Member Data Documentation

5.20.4.1 real* GradientFunction::dWeights

The derivatives of the weights.

5.20.4.2 real* GradientFunction::weights

The weights (parameters) of the gradient function.

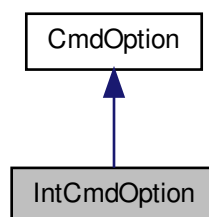
The documentation for this class was generated from the following file:

- [src/qualia/learning/GradientFunction.h](#)

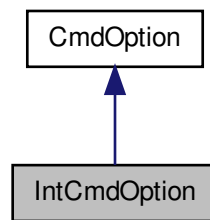
5.21 IntCmdOption Class Reference

```
#include <CmdOption.h>
```

Inheritance diagram for IntCmdOption:



Collaboration diagram for IntCmdOption:



Public Member Functions

- [IntCmdOption](#) (const char *name_, int *ptr_, int init_value_, const char *help_="", bool save_=false)
- virtual void [initValue](#) ()
Initialize the value of the option.
- virtual void [printValue](#) (DiskXFile *file_)
If [is_setted](#)# is true, print the current value, else the init value.
- virtual void [read](#) (int *argc_, char ***argv_)
- virtual void [load](#) (DiskXFile *file)
- virtual void [save](#) (DiskXFile *file)
- [~IntCmdOption](#) ()

Public Attributes

- int * [ptr](#)
- int [init_value](#)

5.21.1 Detailed Description

This class defines a integer command-line option.

Author

Ronan Collobert (collober@idiap.ch)

See Also

[CmdLine](#)

5.21.2 Constructor & Destructor Documentation

5.21.2.1 `IntCmdOption::IntCmdOption (const char * name_, int * ptr_, int init_value_, const char * help_ = " ", bool save_ = false)`

5.21.2.2 `IntCmdOption::~IntCmdOption ()`

5.21.3 Member Function Documentation

5.21.3.1 void IntCmdOption::initValue () [virtual]

Initialize the value of the option.

Reimplemented from [CmdOption](#).

5.21.3.2 void IntCmdOption::load (DiskXFile * file) [virtual]

Reimplemented from [CmdOption](#).

5.21.3.3 void IntCmdOption::printValue (DiskXFile * file_) [virtual]

If [is_setted](#) is true, print the current value, else the init value.

Reimplemented from [CmdOption](#).

5.21.3.4 void IntCmdOption::read (int * argc_, char *** argv_) [virtual]

Read the option on the command line. argv_ and argc_ have to point of the next option after that.

Reimplemented from [CmdOption](#).

5.21.3.5 void IntCmdOption::save (DiskXFile * file) [virtual]

Reimplemented from [CmdOption](#).

5.21.4 Member Data Documentation

5.21.4.1 int IntCmdOption::init_value

5.21.4.2 int* IntCmdOption::ptr

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

- [src/qualia/computer/CmdOption.h](#)
- [src/qualia/computer/CmdOption.cpp](#)

5.22 NeuralNetwork::Layer Struct Reference

[Layer](#) structure.

```
#include <NeuralNetwork.h>
```

Public Attributes

- unsigned int [n](#)
- [real](#) * [output](#)
- [real](#) * [error](#)
- [real](#) * [weight](#)
- [real](#) * [dWeight](#)
- bool [linear](#)

5.22.1 Detailed Description

[Layer](#) structure.

5.22.2 Member Data Documentation

5.22.2.1 `real*` `NeuralNetwork::Layer::dWeight`

5.22.2.2 `real*` `NeuralNetwork::Layer::error`

5.22.2.3 `bool` `NeuralNetwork::Layer::linear`

5.22.2.4 `unsigned int` `NeuralNetwork::Layer::n`

5.22.2.5 `real*` `NeuralNetwork::Layer::output`

5.22.2.6 `real*` `NeuralNetwork::Layer::weight`

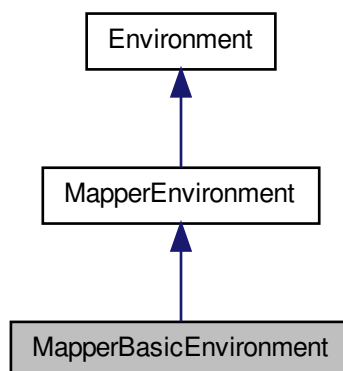
The documentation for this struct was generated from the following file:

- `src/qualia/learning/`[NeuralNetwork.h](#)

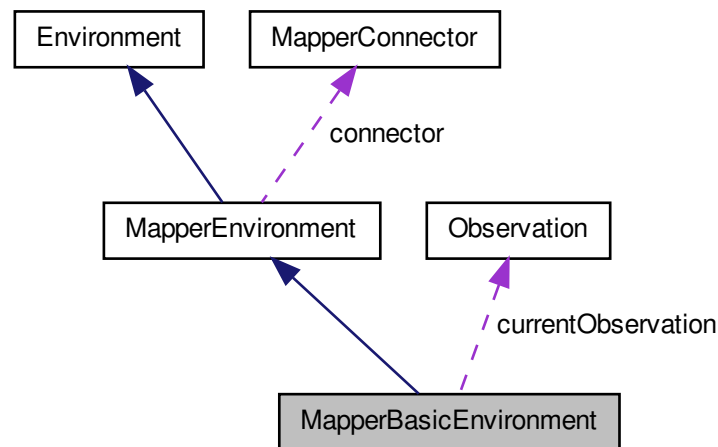
5.23 MapperBasicEnvironment Class Reference

```
#include <MapperBasicEnvironment.h>
```

Inheritance diagram for MapperBasicEnvironment:



Collaboration diagram for MapperBasicEnvironment:



Public Member Functions

- [MapperBasicEnvironment](#) (int observationDim, int actionDim, [MapperConnector](#) *connector)
- virtual [~MapperBasicEnvironment](#) ()
- virtual void [addSignals](#) ()
- virtual void [writeOutputs](#) (const [Action](#) *action)
- virtual [Observation](#) * [readInputs](#) ()

Public Attributes

- [Observation](#) currentObservation
- int actionDim

5.23.1 Constructor & Destructor Documentation

5.23.1.1 [MapperBasicEnvironment::MapperBasicEnvironment](#) (int *observationDim*, int *actionDim*, [MapperConnector](#) * *connector*)

5.23.1.2 [MapperBasicEnvironment::~~MapperBasicEnvironment](#) () [virtual]

5.23.2 Member Function Documentation

5.23.2.1 void [MapperBasicEnvironment::addSignals](#) () [virtual]

Implements [MapperEnvironment](#).

5.23.2.2 [Observation](#) * [MapperBasicEnvironment::readInputs](#) () [virtual]

Implements [MapperEnvironment](#).

5.23.2.3 void MapperBasicEnvironment::writeOutputs (const Action * *action*) [virtual]

Implements [MapperEnvironment](#).

5.23.3 Member Data Documentation

5.23.3.1 int MapperBasicEnvironment::actionDim

5.23.3.2 Observation MapperBasicEnvironment::currentObservation

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

- src/qualia/plugins/mapper/[MapperBasicEnvironment.h](#)
- src/qualia/plugins/mapper/[MapperBasicEnvironment.cpp](#)

5.24 MapperConnector Class Reference

```
#include <MapperConnector.h>
```

Classes

- struct [SignalData](#)

Public Types

- typedef [std::map](#)< [std::string](#), [SignalData](#) * > [SignalDataMap](#)

Public Member Functions

- [MapperConnector](#) (const char *[deviceName](#), const char *[peerDeviceName](#), bool [autoConnect](#)=true, int [initialPort](#)=9000)
- virtual [~MapperConnector](#) ()
- void [init](#) ()
- void [logout](#) ()
- void [createConnections](#) ()
- void [addInput](#) (const char *name, int length, char type, const char *unit, void *minimum, void *maximum, bool blocking=[true](#), float *initialData=0)
- void [addOutput](#) (const char *name, int length, char type, const char *unit, void *minimum, void *maximum, float *initialData=0)
- void [readInput](#) (const char *name, float *data)
- void [readInput](#) (const char *name, int *data)
- void [writeOutput](#) (const char *name, const float *data)
- void [writeOutput](#) (const char *name, const int *data)
- void [waitForBlockingInputs](#) ()
- void [sendAllOutputs](#) ()

Static Public Member Functions

- static void [updateInput](#) (mapper_signal sig, mapper_db_signal props, mapper_timetag_t *timetag, void *value)
- static void [devDbCallback](#) (mapper_db_device record, mapper_db_action_t action, void *user)
- static void [linkDbCallback](#) (mapper_db_link record, mapper_db_action_t action, void *user)

Public Attributes

- char * [deviceName](#)
- char * [peerDeviceName](#)
- bool [autoConnect](#)
- int [initialPort](#)
- [SignalDataMap](#) [inputData](#)
- [SignalDataMap](#) [outputData](#)
- int [nLinked](#)
- mapper_admin [admin](#)
- mapper_device [dev](#)
- mapper_monitor [mon](#)
- mapper_db [db](#)

5.24.1 Member Typedef Documentation

5.24.1.1 typedef std::map<std::string, SignalData*> MapperConnector::SignalDataMap

5.24.2 Constructor & Destructor Documentation

5.24.2.1 MapperConnector::MapperConnector (const char * *deviceName*, const char * *peerDeviceName*, bool *autoConnect* = true, int *initialPort* = 9000)

5.24.2.2 MapperConnector::~MapperConnector () [virtual]

5.24.3 Member Function Documentation

5.24.3.1 void MapperConnector::addInput (const char * *name*, int *length*, char *type*, const char * *unit*, void * *minimum*, void * *maximum*, bool *blocking* = true, float * *initialData* = 0)

5.24.3.2 void MapperConnector::addOutput (const char * *name*, int *length*, char *type*, const char * *unit*, void * *minimum*, void * *maximum*, float * *initialData* = 0)

5.24.3.3 void MapperConnector::createConnections ()

5.24.3.4 void MapperConnector::devDbCallback (mapper_db_device *record*, mapper_db_action_t *action*, void * *user*)
[static]

5.24.3.5 void MapperConnector::init ()

5.24.3.6 void MapperConnector::linkDbCallback (mapper_db_link *record*, mapper_db_action_t *action*, void * *user*)
[static]

5.24.3.7 void MapperConnector::logout ()

5.24.3.8 void MapperConnector::readInput (const char * *name*, float * *data*)

5.24.3.9 void MapperConnector::readInput (const char * *name*, int * *data*)

5.24.3.10 void MapperConnector::sendAllOutputs ()

5.24.3.11 void MapperConnector::updateInput (mapper_signal *sig*, mapper_db_signal *props*, mapper_timetag_t * *timetag*, void * *value*) [static]

5.24.3.12 void MapperConnector::waitForBlockingInputs ()

5.24.3.13 void MapperConnector::writeOutput (const char * *name*, const float * *data*)

5.24.3.14 void MapperConnector::writeOutput (const char * *name*, const int * *data*)

5.24.4 Member Data Documentation

5.24.4.1 mapper_admin MapperConnector::admin

5.24.4.2 bool MapperConnector::autoConnect

5.24.4.3 mapper_db MapperConnector::db

5.24.4.4 mapper_device MapperConnector::dev

5.24.4.5 char* MapperConnector::deviceName

5.24.4.6 int MapperConnector::initialPort

5.24.4.7 SignalDataMap MapperConnector::inputData

5.24.4.8 mapper_monitor MapperConnector::mon

5.24.4.9 int MapperConnector::nLinked

5.24.4.10 SignalDataMap MapperConnector::outputData

5.24.4.11 char* MapperConnector::peerDeviceName

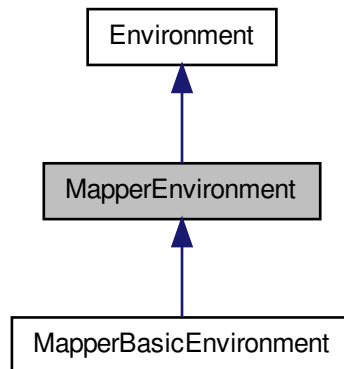
The documentation for this class was generated from the following files:

- src/qualia/plugins/mapper/[MapperConnector.h](#)
- src/qualia/plugins/mapper/[MapperConnector.cpp](#)

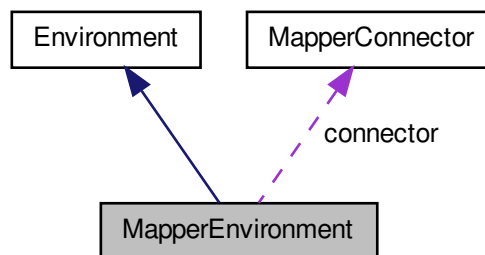
5.25 MapperEnvironment Class Reference

```
#include <MapperEnvironment.h>
```

Inheritance diagram for MapperEnvironment:



Collaboration diagram for MapperEnvironment:



Public Member Functions

- [MapperEnvironment](#) ([MapperConnector](#) *connector)
- virtual [~MapperEnvironment](#) ()
- virtual void [init](#) ()
Initializes the environment.
- virtual [Observation](#) * [start](#) ()
Sends the first observation.
- virtual [Observation](#) * [step](#) (const [Action](#) *action)
Performs action action and returns observation.
- virtual void [addSignals](#) ()=0
- virtual void [writeOutputs](#) (const [Action](#) *action)=0
- virtual [Observation](#) * [readInputs](#) ()=0

Public Attributes

- [MapperConnector](#) * *connector*

5.25.1 Constructor & Destructor Documentation

5.25.1.1 `MapperEnvironment::MapperEnvironment (MapperConnector * connector)`

5.25.1.2 `MapperEnvironment::~MapperEnvironment ()` [virtual]

5.25.2 Member Function Documentation

5.25.2.1 `virtual void MapperEnvironment::addSignals ()` [pure virtual]

Implemented in [MapperBasicEnvironment](#).

5.25.2.2 `void MapperEnvironment::init ()` [virtual]

Initializes the environment.

Reimplemented from [Environment](#).

5.25.2.3 `virtual Observation* MapperEnvironment::readInputs ()` [pure virtual]

Implemented in [MapperBasicEnvironment](#).

5.25.2.4 `Observation * MapperEnvironment::start ()` [virtual]

Sends the first observation.

Implements [Environment](#).

5.25.2.5 `Observation * MapperEnvironment::step (const Action * action)` [virtual]

Performs action *action* and returns observation.

Implements [Environment](#).

5.25.2.6 `virtual void MapperEnvironment::writeOutputs (const Action * action)` [pure virtual]

Implemented in [MapperBasicEnvironment](#).

5.25.3 Member Data Documentation

5.25.3.1 `MapperConnector* MapperEnvironment::connector`

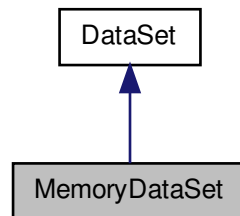
The documentation for this class was generated from the following files:

- `src/qualia/plugins/mapper/MapperEnvironment.h`
- `src/qualia/plugins/mapper/MapperEnvironment.cpp`

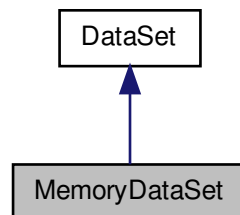
5.26 MemoryDataSet Class Reference

```
#include <MemoryDataSet.h>
```

Inheritance diagram for MemoryDataSet:



Collaboration diagram for MemoryDataSet:



Public Member Functions

- [MemoryDataSet](#) ([DataSet](#) *dataSet)
- virtual [~MemoryDataSet](#) ()
- virtual void [setExample](#) (int t)

Sets this->example to given index t.

Public Attributes

- [real](#) * [data](#)

Pointer to the data.

5.26.1 Detailed Description

A [DataSet](#) that is fully loaded into memory (RAM).

5.26.2 Constructor & Destructor Documentation

5.26.2.1 `MemoryDataSet::MemoryDataSet (DataSet * dataSet)`

5.26.2.2 `MemoryDataSet::~~MemoryDataSet ()` `[virtual]`

5.26.3 Member Function Documentation

5.26.3.1 `void MemoryDataSet::setExample (int t)` `[virtual]`

Sets this->example to given index *t*.

Implements [DataSet](#).

5.26.4 Member Data Documentation

5.26.4.1 `real* MemoryDataSet::data`

Pointer to the data.

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

- [src/qualia/learning/MemoryDataSet.h](#)
- [src/qualia/learning/MemoryDataSet.cpp](#)

5.27 MovingAverage Class Reference

```
#include <MovingAverage.h>
```

Public Member Functions

- [MovingAverage](#) (float startValue, float alphaOrN)
- void [update](#) (float v)
- float [get](#) () const
- void [reset](#) (float startValue)

Private Attributes

- float [_alpha](#)
- float [_value](#)

5.27.1 Constructor & Destructor Documentation

5.27.1.1 `MovingAverage::MovingAverage (float startValue, float alphaOrN)` `[inline]`

Constructs the moving average, starting with #startValue# as its value. The #alphaOrN# argument has two options:

- if ≤ 1 then it's used directly as the alpha value
- if > 1 then it's used as the "number of items that are considered from the past" (*) (*) Of course this is an approximation. It actually sets the alpha value to $2 / (n - 1)$

5.27.2 Member Function Documentation

5.27.2.1 `float MovingAverage::get () const` `[inline]`

Returns the value of the moving average.

5.27.2.2 `void MovingAverage::reset (float startValue)` `[inline]`

Resets the moving average to #startValue#.

5.27.2.3 `void MovingAverage::update (float v)` `[inline]`

Updates the moving average with new value #v#.

5.27.3 Member Data Documentation

5.27.3.1 `float MovingAverage::_alpha` `[private]`

5.27.3.2 `float MovingAverage::_value` `[private]`

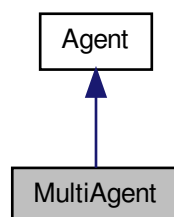
The documentation for this class was generated from the following file:

- `src/qualia/util/MovingAverage.h`

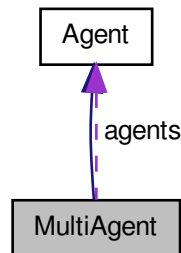
5.28 MultiAgent Class Reference

```
#include <MultiAgent.h>
```

Inheritance diagram for MultiAgent:



Collaboration diagram for MultiAgent:



Public Member Functions

- `MultiAgent` (int `nAgents`, `Agent` **`assign`=0)
- virtual `~MultiAgent` ()
Class destructor (frees the sub-agents if ownsAgents is true).
- virtual void `init` ()
Initializes the agent.
- virtual `Action` * `start` (const `Observation` *`observation`)
Chooses the first action in the episode.
- virtual `Action` * `step` (const `Observation` *`observation`)
Performs an episode step.
- virtual void `end` (const `Observation` *`observation`)
Performs the last episode step.
- virtual `Observation` * `extractAgentObservation` (int `agentIdx`, const `Observation` *`observation`)=0
- virtual void `appendAgentAction` (int `agentIdx`, const `Action` *`agentAction`)=0
- virtual `Action` * `combineAgentActions` ()=0

Public Attributes

- `Agent` ** `agents`
The array of sub-agents.
- int `nAgents`
The number of sub-agents.
- bool `ownsAgents`
True if this `MultiAgent` owns the pointers to the agents (and will thus free them at destruction).

5.28.1 Constructor & Destructor Documentation

5.28.1.1 `MultiAgent::MultiAgent (int nAgents, Agent ** assign = 0)`

Constructor. Creates a multi-agent of `nAgents` sub-agents. Sub-agents can be assigned on-the-spot by specifying `assign` or later on by assigning them eg. `this->agents[i] = new MyAgent()`. If such is the case (ie. if `assign` is NULL), then the `MultiAgent` will also own the agents.

5.28.1.2 MultiAgent::~MultiAgent () [virtual]

Class destructor (frees the sub-agents if ownsAgents is true).

5.28.2 Member Function Documentation

5.28.2.1 virtual void MultiAgent::appendAgentAction (int *agentIdx*, const Action * *agentAction*) [pure virtual]

Appends action from sub-agent *agentIdx* to the action that will later be returned by the call to [combineAgentActions\(\)](#).

5.28.2.2 virtual Action* MultiAgent::combineAgentActions () [pure virtual]

Returns the combine action that is the result of combining all the sub-agent actions through calls to [appendAgentAction\(agentIdx, agentAction\)](#).

5.28.2.3 void MultiAgent::end (const Observation * *observation*) [virtual]

Performs the last episode step.

Reimplemented from [Agent](#).

5.28.2.4 virtual Observation* MultiAgent::extractAgentObservation (int *agentIdx*, const Observation * *observation*) [pure virtual]

Returns an observation suitable for sub-agent *agentIdx* starting from observation *observation* as received by this [MultiAgent](#).

5.28.2.5 void MultiAgent::init () [virtual]

Initializes the agent.

Reimplemented from [Agent](#).

5.28.2.6 Action * MultiAgent::start (const Observation * *observation*) [virtual]

Chooses the first action in the episode.

Implements [Agent](#).

5.28.2.7 Action * MultiAgent::step (const Observation * *observation*) [virtual]

Performs an episode step.

Implements [Agent](#).

5.28.3 Member Data Documentation

5.28.3.1 Agent** MultiAgent::agents

The array of sub-agents.

5.28.3.2 int MultiAgent::nAgents

The number of sub-agents.

5.28.3.3 bool MultiAgent::ownsAgents

True if this [MultiAgent](#) owns the pointers to the agents (and will thus free them at destruction).

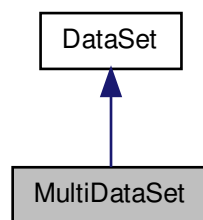
The documentation for this class was generated from the following files:

- [src/qualia/core/MultiAgent.h](#)
- [src/qualia/core/MultiAgent.cpp](#)

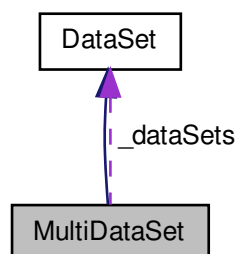
5.29 MultiDataSet Class Reference

```
#include <MultiDataSet.h>
```

Inheritance diagram for MultiDataSet:



Collaboration diagram for MultiDataSet:



Public Member Functions

- [MultiDataSet](#) ([DataSet](#) **dataSets, int nDataSets)

- virtual [~MultiDataSet](#) ()
- virtual void [init](#) ()
- virtual void [reset](#) ()
To be called before every episode.
- virtual void [setExample](#) (int t)
Sets this->example to given index t.

Public Attributes

- [DataSet](#) ** [_dataSets](#)
- int [_nDataSets](#)

5.29.1 Detailed Description

A dataset that is the result of the concatenation of many datasets.

5.29.2 Constructor & Destructor Documentation

5.29.2.1 [MultiDataSet::MultiDataSet](#) ([DataSet](#) ** *dataSets*, int *nDataSets*)

5.29.2.2 [MultiDataSet::~~MultiDataSet](#) () [virtual]

5.29.3 Member Function Documentation

5.29.3.1 void [MultiDataSet::init](#) () [virtual]

To be called before training. Should allocate example, among other things. Default version allocates example to the size of *dim*.

Reimplemented from [DataSet](#).

5.29.3.2 void [MultiDataSet::reset](#) () [virtual]

To be called before every episode.

Reimplemented from [DataSet](#).

5.29.3.3 void [MultiDataSet::setExample](#) (int *t*) [virtual]

Sets this->example to given index *t*.

Implements [DataSet](#).

5.29.4 Member Data Documentation

5.29.4.1 [DataSet](#)** [MultiDataSet::_dataSets](#)

5.29.4.2 int [MultiDataSet::_nDataSets](#)

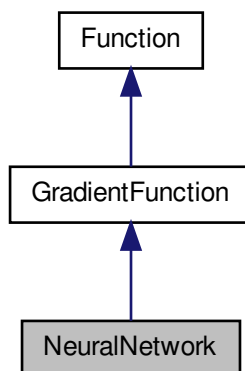
The documentation for this class was generated from the following files:

- [src/qualia/learning/MultiDataSet.h](#)
- [src/qualia/learning/MultiDataSet.cpp](#)

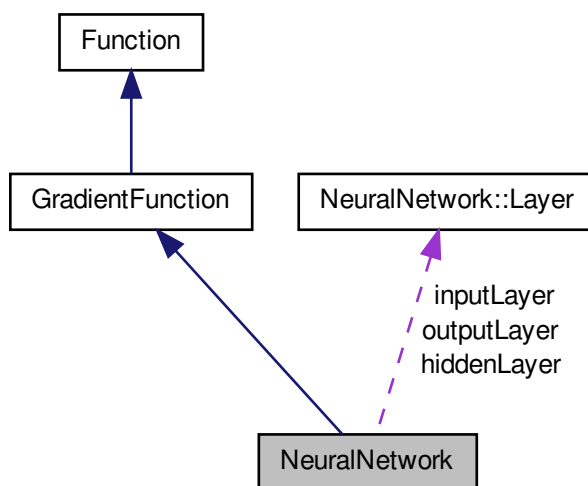
5.30 NeuralNetwork Class Reference

```
#include <NeuralNetwork.h>
```

Inheritance diagram for NeuralNetwork:



Collaboration diagram for NeuralNetwork:



Classes

- struct [Layer](#)
Layer structure.

Public Member Functions

- **NeuralNetwork** (unsigned int **nInputs**, unsigned int **nHiddens**, unsigned int **nOutputs**, float **learningRate**=0.01, float **decreaseConstant**=0, float **weightDecay**=0, bool **linearOutput**=false)
- virtual **~NeuralNetwork** ()
- virtual void **init** ()
Initializes the network (resets the weights, among other things).
- virtual unsigned int **nInputs** () const
Returns the number of inputs.
- virtual unsigned int **nHidden** () const
Returns the number of hidden neurons.
- virtual unsigned int **nOutputs** () const
Returns the number of outputs.
- virtual unsigned int **nParams** () const
Returns the number of parameters.
- virtual float **getCurrentLearningRate** () const
*Returns the current effective learning rate (= learningRate / (1 + t * decreaseConstant)).*
- virtual void **setInput** (int i, **real** x)
Sets input i to value x.
- virtual void **setInputs** (const **real** *inputs)
Sets the value of the inputs.
- virtual **real** **getOutput** (int i) const
Get output i.
- virtual void **getOutputs** (**real** *outputs) const
Get the value of the outputs.
- virtual void **backpropagate** (**real** *outputError)
Backpropagates the error, updating the derivatives.
- virtual void **propagate** ()
Propagates inputs to outputs.
- virtual void **update** ()
Updates the weights according to the derivatives.
- virtual void **save** (XFile *file)
Saves the model to a file.
- virtual void **load** (XFile *file)
Loads the model from a file.
- void **_allocateLayer** (Layer &layer, unsigned int **nInputs**, unsigned int **nOutputs**, unsigned int &k, bool is-Linear=false)
- void **_deallocateLayer** (Layer &layer)
- void **_propagateLayer** (Layer &lower, Layer &upper)
- void **_backpropagateLayer** (Layer &upper, Layer &lower)
- void **_deallocate** ()

Public Attributes

- float **learningRate**
- float **decreaseConstant**
- float **weightDecay**
- float **_learningRateDiv**
- unsigned int **_nParams**
Number of parameters.
- Layer **inputLayer**
The three MLP layers (inputs -> hidden -> outputs).
- Layer **hiddenLayer**
- Layer **outputLayer**

5.30.1 Constructor & Destructor Documentation

5.30.1.1 `NeuralNetwork::NeuralNetwork (unsigned int nInputs, unsigned int nHiddens, unsigned int nOutputs, float learningRate = 0.01, float decreaseConstant = 0, float weightDecay = 0, bool linearOutput = false)`

5.30.1.2 `NeuralNetwork::~~NeuralNetwork () [virtual]`

5.30.2 Member Function Documentation

5.30.2.1 `void NeuralNetwork::_allocateLayer (Layer & layer, unsigned int nInputs, unsigned int nOutputs, unsigned int & k, bool isLinear = false)`

5.30.2.2 `void NeuralNetwork::_backpropagateLayer (Layer & upper, Layer & lower)`

5.30.2.3 `void NeuralNetwork::_deallocate ()`

5.30.2.4 `void NeuralNetwork::_deallocateLayer (Layer & layer)`

5.30.2.5 `void NeuralNetwork::_propagateLayer (Layer & lower, Layer & upper)`

5.30.2.6 `void NeuralNetwork::backpropagate (real * outputError) [virtual]`

Backpropagates the error, updating the derivatives.

Implements [GradientFunction](#).

5.30.2.7 `float NeuralNetwork::getCurrentLearningRate () const [virtual]`

Returns the current effective learning rate (= $\text{learningRate} / (1 + t * \text{decreaseConstant})$).

5.30.2.8 `real NeuralNetwork::getOutput (int i) const [virtual]`

Get output *i*.

Implements [Function](#).

5.30.2.9 `void NeuralNetwork::getOutputs (real * output) const [virtual]`

Get the value of the outputs.

Reimplemented from [Function](#).

5.30.2.10 `void NeuralNetwork::init () [virtual]`

Initializes the network (resets the weights, among other things).

Reimplemented from [Function](#).

5.30.2.11 `void NeuralNetwork::load (XFile * file) [virtual]`

Loads the model from a file.

Reimplemented from [GradientFunction](#).

5.30.2.12 `virtual unsigned int NeuralNetwork::nHidden () const [inline],[virtual]`

Returns the number of hidden neurons.

5.30.2.13 `virtual unsigned int NeuralNetwork::nInputs () const [inline],[virtual]`

Returns the number of inputs.

Implements [Function](#).

5.30.2.14 `virtual unsigned int NeuralNetwork::nOutputs () const [inline],[virtual]`

Returns the number of outputs.

Implements [Function](#).

5.30.2.15 `virtual unsigned int NeuralNetwork::nParams () const [inline],[virtual]`

Returns the number of parameters.

Implements [GradientFunction](#).

5.30.2.16 `void NeuralNetwork::propagate () [virtual]`

Propagates inputs to outputs.

Implements [Function](#).

5.30.2.17 `void NeuralNetwork::save (XFile * file) [virtual]`

Saves the model to a file.

Reimplemented from [GradientFunction](#).

5.30.2.18 `void NeuralNetwork::setInput (int i, real x) [virtual]`

Sets input *i* to value *x*.

Implements [Function](#).

5.30.2.19 `void NeuralNetwork::setInputs (const real * input) [virtual]`

Sets the value of the inputs.

Reimplemented from [Function](#).

5.30.2.20 `void NeuralNetwork::update () [virtual]`

Updates the weights according to the derivatives.

Implements [GradientFunction](#).

5.30.3 Member Data Documentation

5.30.3.1 float NeuralNetwork::_learningRateDiv

This value is used to keep track of the learning rate divider: it is equal to $(1 + t * \text{decreaseConstant})$. It is more efficient than the usual way of updating the learning rate, because it requires only one floating point addition per iteration, instead of an addition and a multiplication.

5.30.3.2 unsigned int NeuralNetwork::_nParams

Number of parameters.

5.30.3.3 float NeuralNetwork::_decreaseConstant

The learning rate decrease constant. Value should be ≥ 0 , usually in $[0, 1]$. The decrease constant is applied as a way to slowly decrease the learning rate during gradient descent to help convergence to a better minimum.

5.30.3.4 Layer NeuralNetwork::hiddenLayer

5.30.3.5 Layer NeuralNetwork::inputLayer

The three MLP layers (inputs -> hidden -> outputs).

5.30.3.6 float NeuralNetwork::_learningRate

The starting learning rate. Value should be ≥ 0 , usually in $[0, 1]$. The learning rate is used to adjust the speed of training. The higher the learning rate the faster the network is trained. However, the network has a better chance of being trained to a local minimum solution. A local minimum is a point at which the network stabilizes on a solution which is not the most optimal global solution. In the case of reinforcement learning, the learning rate determines to what extent the newly acquired information will override the old information. A factor of 0 will make the agent not learn anything, while a factor of 1 would make the agent consider only the most recent information. Source: <http://pages.cs.wisc.edu/~bolo/shipyard/neural/tort.html> http://en.wikipedia.org/wiki/Q-learning#Learning_rate

5.30.3.7 Layer NeuralNetwork::outputLayer

5.30.3.8 float NeuralNetwork::_weightDecay

The weight decay. Value should be ≥ 0 , usually in $[0, 1]$. Weight decay is a simple regularization method that limits the number of free parameters in the model so as to prevent over-fitting (in other words, to get a better generalization). In practice, it penalizes large weights and thus also limits the freedom in the model.

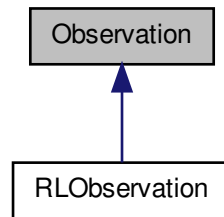
The documentation for this class was generated from the following files:

- [src/qualia/learning/NeuralNetwork.h](#)
- [src/qualia/learning/NeuralNetwork.cpp](#)

5.31 Observation Class Reference

```
#include <Observation.h>
```

Inheritance diagram for Observation:



Public Member Functions

- `Observation` (unsigned int `dim`)
Constructor (dimension needs to be known at construction).
- virtual `~Observation` ()
- `observation_t & operator[]` (int i) const
Access operator.
- virtual `Observation & copyFrom` (const `Observation` &src)
Copies data from src.
- unsigned int `dim` () const
Returns dimension of data.
- virtual void `saveData` (`XFile` *file) const
Save data to file (in binary format).
- virtual void `loadData` (`XFile` *file)
Loads data from file (in binary format). Dimension needs to be known in advance.

Public Attributes

- `observation_t * observations`
The array containing the observations (of size `dim()`).
- unsigned int `_dim`
Dimension of data.
- bool `terminal`
True iff this observation is terminal.

5.31.1 Detailed Description

Represent observations returned by the environment to the agents. Observations in `Qualia` are continuous. A specific observation is represented by a multi-dimensional array of real numbers. Notice that discrete observation components can be represented as well since the underlying data is an array of real numbers, so you can put integers in there as long as you know what you're doing.

5.31.2 Constructor & Destructor Documentation

5.31.2.1 `Observation::Observation (unsigned int dim)`

Constructor (dimension needs to be known at construction).

5.31.2.2 `Observation::~~Observation () [virtual]`

5.31.3 Member Function Documentation

5.31.3.1 `Observation & Observation::copyFrom (const Observation & src) [virtual]`

Copies data from *src*.

Reimplemented in [RLObservation](#).

5.31.3.2 `unsigned int Observation::dim () const [inline]`

Returns dimension of data.

5.31.3.3 `void Observation::loadData (XFile * file) [virtual]`

Loads data from *file* (in binary format). Dimension needs to be known in advance.

Reimplemented in [RLObservation](#).

5.31.3.4 `observation_t& Observation::operator[] (int i) const [inline]`

Access operator.

5.31.3.5 `void Observation::saveData (XFile * file) const [virtual]`

Save data to *file* (in binary format).

Reimplemented in [RLObservation](#).

5.31.4 Member Data Documentation

5.31.4.1 `unsigned int Observation::_dim`

Dimension of data.

5.31.4.2 `observation_t* Observation::observations`

The array containing the observations (of size `dim()`).

5.31.4.3 `bool Observation::terminal`

True iff this observation is terminal.

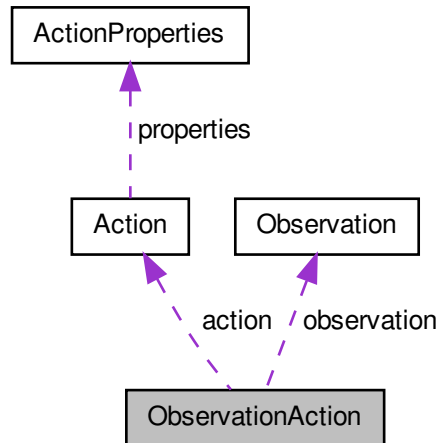
The documentation for this class was generated from the following files:

- [src/qualia/core/Observation.h](#)
- [src/qualia/core/Observation.cpp](#)

5.32 ObservationAction Struct Reference

```
#include <Qualia.h>
```

Collaboration diagram for ObservationAction:



Public Attributes

- [Observation](#) * `observation`
- [Action](#) * `action`

5.32.1 Member Data Documentation

5.32.1.1 **Action*** `ObservationAction::action`

5.32.1.2 **Observation*** `ObservationAction::observation`

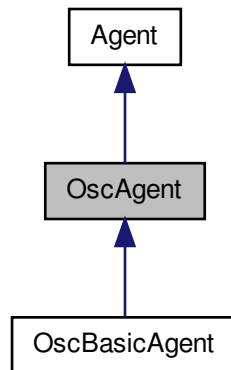
The documentation for this struct was generated from the following file:

- `src/qualia/core/Qualia.h`

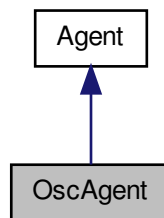
5.33 OscAgent Class Reference

```
#include <OscAgent.h>
```

Inheritance diagram for OscAgent:



Collaboration diagram for OscAgent:



Public Member Functions

- `OscAgent` (int id, int observationDim, int actionDim, int observationBufferDim)
- virtual `~OscAgent` ()
- char * `getPath` (const char *path)
- virtual void `init` ()
Initializes the agent.
- virtual `Action` * `start` (const `Observation` *observation)
Chooses the first action in the episode.
- virtual `Action` * `step` (const `Observation` *observation)
Performs an episode step.
- virtual `Action` * `readAction` (action_dim_t *buffer)=0

Static Public Member Functions

- static int `handlerInit` (const char *path, const char *types, lo_arg **argv, int argc, void *data, void *user_data)

- static int [handlerStartStep](#) (const char *path, const char *types, lo_arg **argv, int argc, void *data, void *user_data)

Public Attributes

- int [id](#)
- int [observationDim](#)
- int [actionDim](#)
- int [observationBufferDim](#)
- [action_dim_t](#) * [actionBuffer](#)
- volatile bool [locked](#)
- char [actionTypes](#) [100]

5.33.1 Constructor & Destructor Documentation

5.33.1.1 `OscAgent::OscAgent (int id, int observationDim, int actionDim, int observationBufferDim)`

5.33.1.2 `OscAgent::~OscAgent ()` [virtual]

5.33.2 Member Function Documentation

5.33.2.1 `char * OscAgent::getPath (const char * path)`

5.33.2.2 `int OscAgent::handlerInit (const char * path, const char * types, lo_arg ** argv, int argc, void * data, void * user_data)` [static]

5.33.2.3 `int OscAgent::handlerStartStep (const char * path, const char * types, lo_arg ** argv, int argc, void * data, void * user_data)` [static]

5.33.2.4 `void OscAgent::init ()` [virtual]

Initializes the agent.

Reimplemented from [Agent](#).

5.33.2.5 `virtual Action* OscAgent::readAction (action_dim_t * buffer)` [pure virtual]

Implemented in [OscBasicAgent](#).

5.33.2.6 `Action * OscAgent::start (const Observation * observation)` [virtual]

Chooses the first action in the episode.

Implements [Agent](#).

5.33.2.7 `Action * OscAgent::step (const Observation * observation)` [virtual]

Performs an episode step.

Implements [Agent](#).

5.33.3 Member Data Documentation

5.33.3.1 `action_dim_t*` `OscAgent::actionBuffer`

5.33.3.2 `int` `OscAgent::actionDim`

5.33.3.3 `char` `OscAgent::actionTypes[100]`

5.33.3.4 `int` `OscAgent::id`

5.33.3.5 `volatile bool` `OscAgent::locked`

5.33.3.6 `int` `OscAgent::observationBufferDim`

5.33.3.7 `int` `OscAgent::observationDim`

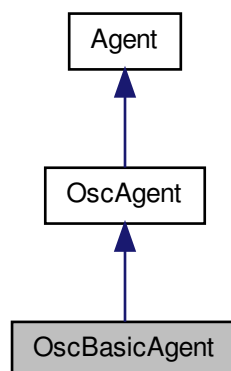
The documentation for this class was generated from the following files:

- `src/qualia/plugins/osc/OscAgent.h`
- `src/qualia/plugins/osc/OscAgent.cpp`

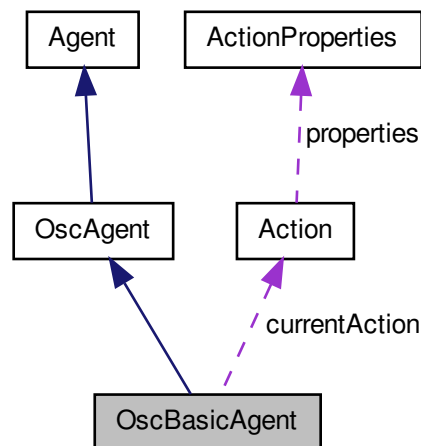
5.34 OscBasicAgent Class Reference

```
#include <OscBasicAgent.h>
```

Inheritance diagram for `OscBasicAgent`:



Collaboration diagram for OscBasicAgent:



Public Member Functions

- [OscBasicAgent](#) (int *id*, int *observationDim*, int *observationBufferDim*, [ActionProperties](#) **actionProperties*)
- virtual [~OscBasicAgent](#) ()
- virtual [Action](#) * [readAction](#) ([action_dim_t](#) **buffer*)

Public Attributes

- [Action](#) *currentAction*

Additional Inherited Members

5.34.1 Constructor & Destructor Documentation

5.34.1.1 `OscBasicAgent::OscBasicAgent (int id, int observationDim, int observationBufferDim, ActionProperties *actionProperties)`

5.34.1.2 `OscBasicAgent::~~OscBasicAgent ()` [virtual]

5.34.2 Member Function Documentation

5.34.2.1 `Action * OscBasicAgent::readAction (action_dim_t *buffer)` [virtual]

Implements [OscAgent](#).

5.34.3 Member Data Documentation

5.34.3.1 `Action OscBasicAgent::currentAction`

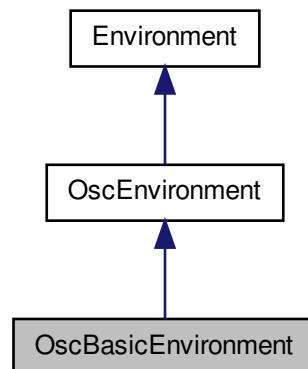
The documentation for this class was generated from the following files:

- [src/qualia/plugins/osc/OscBasicAgent.h](#)
- [src/qualia/plugins/osc/OscBasicAgent.cpp](#)

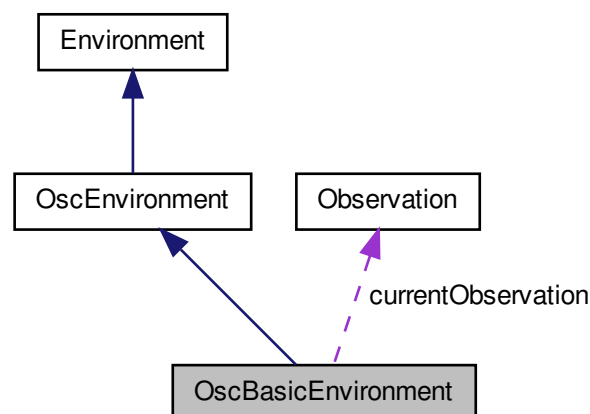
5.35 OscBasicEnvironment Class Reference

```
#include <OscBasicEnvironment.h>
```

Inheritance diagram for OscBasicEnvironment:



Collaboration diagram for OscBasicEnvironment:



Public Member Functions

- [OscBasicEnvironment](#) (int id, int observationDim, int actionDim)

- virtual [~OscBasicEnvironment](#) ()
- virtual [Observation](#) * [readObservation](#) ([observation_t](#) *buffer)

Public Attributes

- [Observation](#) [currentObservation](#)

Additional Inherited Members

5.35.1 Constructor & Destructor Documentation

5.35.1.1 [OscBasicEnvironment::OscBasicEnvironment](#) ([int](#) *id*, [int](#) *observationDim*, [int](#) *actionDim*)

5.35.1.2 [OscBasicEnvironment::~~OscBasicEnvironment](#) () [virtual]

5.35.2 Member Function Documentation

5.35.2.1 [Observation](#) * [OscBasicEnvironment::readObservation](#) ([observation_t](#) * *buffer*) [virtual]

Implements [OscEnvironment](#).

5.35.3 Member Data Documentation

5.35.3.1 [Observation](#) [OscBasicEnvironment::currentObservation](#)

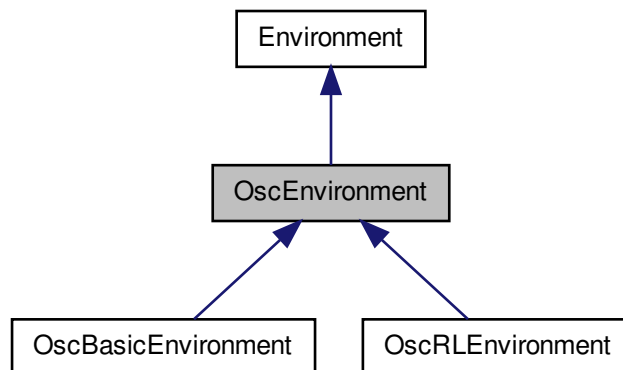
The documentation for this class was generated from the following files:

- [src/qualia/plugins/osc/OscBasicEnvironment.h](#)
- [src/qualia/plugins/osc/OscBasicEnvironment.cpp](#)

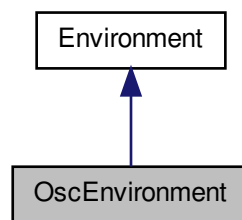
5.36 OscEnvironment Class Reference

```
#include <OscEnvironment.h>
```

Inheritance diagram for OscEnvironment:



Collaboration diagram for OscEnvironment:



Public Member Functions

- [OscEnvironment](#) (int [id](#), int [observationDim](#), int [actionDim](#), int [observationBufferDim](#))
- virtual [~OscEnvironment](#) ()
- char * [getPath](#) (const char *path)
- virtual void [init](#) ()
Initializes the environment.
- virtual [Observation](#) * [start](#) ()
Sends the first observation.
- virtual [Observation](#) * [step](#) (const [Action](#) *action)
Performs action action and returns observation.
- virtual [Observation](#) * [readObservation](#) ([observation_t](#) *buffer)=0

Static Public Member Functions

- static int [handlerInit](#) (const char *path, const char *types, lo_arg **argv, int argc, void *data, void *user_data)

- static int [handlerStartStep](#) (const char *path, const char *types, lo_arg **argv, int argc, void *data, void *user_data)

Public Attributes

- int [id](#)
- int [observationDim](#)
- int [actionDim](#)
- int [observationBufferDim](#)
- [observation_t](#) * [observationBuffer](#)
- volatile bool [locked](#)
- char [observationTypes](#) [100]

5.36.1 Constructor & Destructor Documentation

5.36.1.1 `OscEnvironment::OscEnvironment (int id, int observationDim, int actionDim, int observationBufferDim)`

5.36.1.2 `OscEnvironment::~~OscEnvironment ()` [virtual]

5.36.2 Member Function Documentation

5.36.2.1 `char * OscEnvironment::getPath (const char * path)`

5.36.2.2 `int OscEnvironment::handlerInit (const char * path, const char * types, lo_arg ** argv, int argc, void * data, void * user_data)` [static]

5.36.2.3 `int OscEnvironment::handlerStartStep (const char * path, const char * types, lo_arg ** argv, int argc, void * data, void * user_data)` [static]

5.36.2.4 `void OscEnvironment::init ()` [virtual]

Initializes the environment.

Reimplemented from [Environment](#).

5.36.2.5 `virtual Observation* OscEnvironment::readObservation (observation_t * buffer)` [pure virtual]

Implemented in [OscRLEnvironment](#), and [OscBasicEnvironment](#).

5.36.2.6 `Observation * OscEnvironment::start ()` [virtual]

Sends the first observation.

Implements [Environment](#).

5.36.2.7 `Observation * OscEnvironment::step (const Action * action)` [virtual]

Performs action *action* and returns observation.

Implements [Environment](#).

5.36.3 Member Data Documentation

5.36.3.1 `int OscEnvironment::actionDim`

5.36.3.2 `int OscEnvironment::id`

5.36.3.3 `volatile bool OscEnvironment::locked`

5.36.3.4 `observation_t* OscEnvironment::observationBuffer`

5.36.3.5 `int OscEnvironment::observationBufferDim`

5.36.3.6 `int OscEnvironment::observationDim`

5.36.3.7 `char OscEnvironment::observationTypes[100]`

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

- `src/qualia/plugins/osc/OscEnvironment.h`
- `src/qualia/plugins/osc/OscEnvironment.cpp`

5.37 OscManager Class Reference

```
#include <OscManager.h>
```

Static Public Member Functions

- static void `initOsc` (const char *host, const char *port, const char *remotePort)
- static void `repeatChar` (char *dst, char c, int times)
- static `lo_address client` ()
- static `lo_address server` ()
- static void `error` (int num, const char *msg, const char *path)

Static Public Attributes

- static `lo_address _client`
- static `lo_server_thread _server`

Private Member Functions

- `OscManager` ()
- virtual `~OscManager` ()

5.37.1 Constructor & Destructor Documentation

5.37.1.1 `OscManager::OscManager ()` [private]

5.37.1.2 `OscManager::~~OscManager ()` [private],[virtual]

5.37.2 Member Function Documentation

5.37.2.1 `static lo_address OscManager::client ()` `[inline],[static]`

5.37.2.2 `void OscManager::error (int num, const char * msg, const char * path)` `[static]`

5.37.2.3 `void OscManager::initOsc (const char * host, const char * port, const char * remotePort)` `[static]`

5.37.2.4 `void OscManager::repeatChar (char * dst, char c, int times)` `[static]`

5.37.2.5 `static lo_address OscManager::server ()` `[inline],[static]`

5.37.3 Member Data Documentation

5.37.3.1 `lo_address OscManager::_client` `[static]`

5.37.3.2 `lo_server_thread OscManager::_server` `[static]`

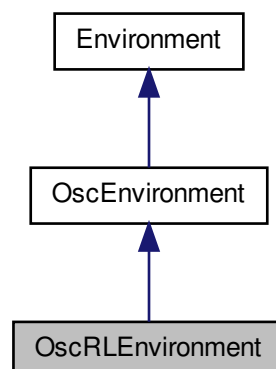
The documentation for this class was generated from the following files:

- [src/qualia/plugins/osc/OscManager.h](#)
- [src/qualia/plugins/osc/OscManager.cpp](#)

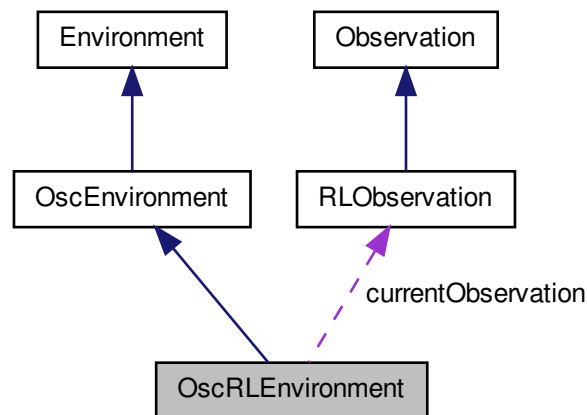
5.38 OscRLEnvironment Class Reference

```
#include <OscRLEnvironment.h>
```

Inheritance diagram for OscRLEnvironment:



Collaboration diagram for OscRLEnvironment:



Public Member Functions

- [OscRLEnvironment](#) (int *id*, int *observationDim*, int *actionDim*)
- virtual [~OscRLEnvironment](#) ()
- virtual [Observation * readObservation](#) ([observation_t](#) **buffer*)

Public Attributes

- [RLObservation currentObservation](#)

Additional Inherited Members

5.38.1 Constructor & Destructor Documentation

5.38.1.1 `OscRLEnvironment::OscRLEnvironment (int id, int observationDim, int actionDim)`

5.38.1.2 `OscRLEnvironment::~~OscRLEnvironment ()` [virtual]

5.38.2 Member Function Documentation

5.38.2.1 `Observation * OscRLEnvironment::readObservation (observation_t * buffer)` [virtual]

Implements [OscEnvironment](#).

5.38.3 Member Data Documentation

5.38.3.1 `RLObservation OscRLEnvironment::currentObservation`

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

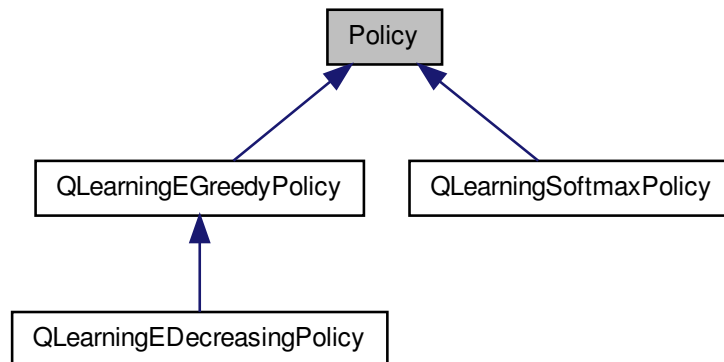
- `src/qualia/plugins/osc/OscRLEnvironment.h`

- [src/qualia/plugins/osc/OscRLEnvironment.cpp](#)

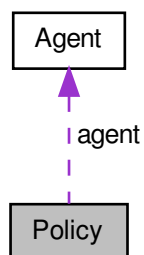
5.39 Policy Class Reference

```
#include <Policy.h>
```

Inheritance diagram for Policy:



Collaboration diagram for Policy:



Public Member Functions

- [Policy](#) ()
- virtual [~Policy](#) ()
- virtual void [init](#) ()
- virtual void [setAgent](#) ([Agent](#) *agent_)
- virtual void [chooseAction](#) ([Action](#) *action, const [Observation](#) *observation)=0

Public Attributes

- [Agent * agent](#)

5.39.1 Constructor & Destructor Documentation

5.39.1.1 `Policy::Policy ()` `[inline]`

5.39.1.2 `virtual Policy::~~Policy ()` `[inline]`, `[virtual]`

5.39.2 Member Function Documentation

5.39.2.1 `virtual void Policy::chooseAction (Action * action, const Observation * observation)` `[pure virtual]`

This method is implemented by subclasses. It chooses an action based on given observation #observation# and puts it in #action#.

Implemented in [QLearningSoftmaxPolicy](#), [QLearningEDecreasingPolicy](#), and [QLearningEGreedyPolicy](#).

5.39.2.2 `virtual void Policy::init ()` `[inline]`, `[virtual]`

Reimplemented in [QLearningEDecreasingPolicy](#).

5.39.2.3 `virtual void Policy::setAgent (Agent * agent_)` `[inline]`, `[virtual]`

5.39.3 Member Data Documentation

5.39.3.1 `Agent* Policy::agent`

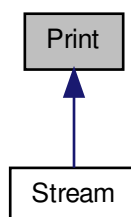
The documentation for this class was generated from the following file:

- [src/qualia/rl/Policy.h](#)

5.40 Print Class Reference

```
#include <Print.h>
```

Inheritance diagram for Print:



Public Member Functions

- [Print](#) ()
- int [getWriteError](#) ()
- void [clearWriteError](#) ()
- virtual size_t [write](#) (uint8_t)=0
- size_t [write](#) (const char *str)
- virtual size_t [write](#) (const uint8_t *buffer, size_t size)
- size_t [print](#) (const String &)
- size_t [print](#) (const char[])
- size_t [print](#) (char)
- size_t [print](#) (unsigned char, int=DEC)
- size_t [print](#) (int, int=DEC)
- size_t [print](#) (unsigned int, int=DEC)
- size_t [print](#) (long, int=DEC)
- size_t [print](#) (unsigned long, int=DEC)
- size_t [print](#) (double, int=2)
- size_t [print](#) (const [Printable](#) &)
- size_t [println](#) (const String &s)
- size_t [println](#) (const char[])
- size_t [println](#) (char)
- size_t [println](#) (unsigned char, int=DEC)
- size_t [println](#) (int, int=DEC)
- size_t [println](#) (unsigned int, int=DEC)
- size_t [println](#) (long, int=DEC)
- size_t [println](#) (unsigned long, int=DEC)
- size_t [println](#) (double, int=2)
- size_t [println](#) (const [Printable](#) &)
- size_t [println](#) (void)

Protected Member Functions

- void [setWriteError](#) (int err=1)

Private Member Functions

- size_t [printNumber](#) (unsigned long, uint8_t)
- size_t [printFloat](#) (double, uint8_t)

Private Attributes

- int [write_error](#)

5.40.1 Constructor & Destructor Documentation

5.40.1.1 `Print::Print () [inline]`

5.40.2 Member Function Documentation

5.40.2.1 `void Print::clearWriteError () [inline]`

5.40.2.2 `int Print::getWriteError () [inline]`

- 5.40.2.3 `size_t Print::print (const String & s)`
- 5.40.2.4 `size_t Print::print (const char str[])`
- 5.40.2.5 `size_t Print::print (char c)`
- 5.40.2.6 `size_t Print::print (unsigned char b, int base = DEC)`
- 5.40.2.7 `size_t Print::print (int n, int base = DEC)`
- 5.40.2.8 `size_t Print::print (unsigned int n, int base = DEC)`
- 5.40.2.9 `size_t Print::print (long n, int base = DEC)`
- 5.40.2.10 `size_t Print::print (unsigned long n, int base = DEC)`
- 5.40.2.11 `size_t Print::print (double n, int digits = 2)`
- 5.40.2.12 `size_t Print::print (const Printable & x)`
- 5.40.2.13 `size_t Print::printFloat (double number, uint8_t digits)` [private]
- 5.40.2.14 `size_t Print::println (const String & s)`
- 5.40.2.15 `size_t Print::println (const char c[])`
- 5.40.2.16 `size_t Print::println (char c)`
- 5.40.2.17 `size_t Print::println (unsigned char b, int base = DEC)`
- 5.40.2.18 `size_t Print::println (int num, int base = DEC)`
- 5.40.2.19 `size_t Print::println (unsigned int num, int base = DEC)`
- 5.40.2.20 `size_t Print::println (long num, int base = DEC)`
- 5.40.2.21 `size_t Print::println (unsigned long num, int base = DEC)`
- 5.40.2.22 `size_t Print::println (double num, int digits = 2)`
- 5.40.2.23 `size_t Print::println (const Printable & x)`
- 5.40.2.24 `size_t Print::println (void)`
- 5.40.2.25 `size_t Print::printNumber (unsigned long n, uint8_t base)` [private]
- 5.40.2.26 `void Print::setWriteError (int err = 1)` [inline], [protected]
- 5.40.2.27 `virtual size_t Print::write (uint8_t)` [pure virtual]
- 5.40.2.28 `size_t Print::write (const char * str)` [inline]
- 5.40.2.29 `size_t Print::write (const uint8_t * buffer, size_t size)` [virtual]

5.40.3 Member Data Documentation

5.40.3.1 `int Print::write_error` `[private]`

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

- `src/qualia/compat/Print.h`
- `src/qualia/compat/Print.cpp`

5.41 Printable Class Reference

```
#include <Printable.h>
```

Public Member Functions

- `virtual size_t printTo (Print &p) const =0`

5.41.1 Detailed Description

The `Printable` class provides a way for new classes to allow themselves to be printed. By deriving from `Printable` and implementing the `printTo` method, it will then be possible for users to print out instances of this class by passing them into the usual `Print::print` and `Print::println` methods.

5.41.2 Member Function Documentation

5.41.2.1 `virtual size_t Printable::printTo (Print & p) const` `[pure virtual]`

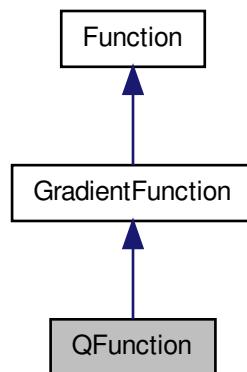
The documentation for this class was generated from the following file:

- `src/qualia/compat/Printable.h`

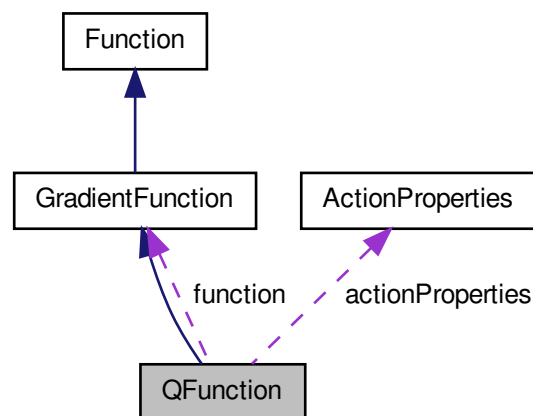
5.42 QFunction Class Reference

```
#include <QFunction.h>
```

Inheritance diagram for QFunction:



Collaboration diagram for QFunction:



Public Member Functions

- **QFunction** (**GradientFunction** ***function**, unsigned int **observationDim**, **ActionProperties** ***actionProperties**)
- virtual **~QFunction** ()
- virtual void **init** ()
Initializes the function.
- virtual unsigned int **nInputs** () const
Returns the number of inputs.
- virtual unsigned int **nOutputs** () const
Returns the number of outputs.

- virtual unsigned int `nParams` () const
Returns the number of parameters.
- virtual void `setInput` (int `i`, real `x`)
Sets input `i` to value `x`.
- virtual real `getOutput` (int `i`) const
Get output `i`.
- virtual void `propagate` ()
Propagates inputs to outputs.
- virtual void `backpropagate` (real *`outputError`)
Backpropagates the error, updating the derivatives.
- virtual void `update` ()
Updates the weights according to the derivatives.
- virtual void `save` (XFile *`file`)
Saves the model to a file.
- virtual void `load` (XFile *`file`)
Loads the model from a file.
- real `getValue` (const Observation *`observation`, const Action *`action`)
- void `getMaxAction` (Action *`dst`, const Observation *`observation`, real *`maxQ`=0)

Public Attributes

- GradientFunction * `function`
- real * `input`
- unsigned int `observationDim`
- ActionProperties * `actionProperties`

5.42.1 Constructor & Destructor Documentation

5.42.1.1 QFunction::QFunction (GradientFunction * `function`, unsigned int `observationDim`, ActionProperties * `actionProperties`)

5.42.1.2 QFunction::~~QFunction () [virtual]

5.42.2 Member Function Documentation

5.42.2.1 virtual void QFunction::backpropagate (real * `outputError`) [inline],[virtual]

Backpropagates the error, updating the derivatives.

Implements [GradientFunction](#).

5.42.2.2 void QFunction::getMaxAction (Action * `dst`, const Observation * `observation`, real * `maxQ` = 0)

5.42.2.3 virtual real QFunction::getOutput (int `i`) const [inline],[virtual]

Get output `i`.

Implements [Function](#).

5.42.2.4 `real QFunction::getValue (const Observation * observation, const Action * action)`

5.42.2.5 `virtual void QFunction::init () [inline],[virtual]`

Initializes the function.

Reimplemented from [Function](#).

5.42.2.6 `virtual void QFunction::load (XFile * file) [inline],[virtual]`

Loads the model from a file.

Reimplemented from [GradientFunction](#).

5.42.2.7 `virtual unsigned int QFunction::nInputs () const [inline],[virtual]`

Returns the number of inputs.

Implements [Function](#).

5.42.2.8 `virtual unsigned int QFunction::nOutputs () const [inline],[virtual]`

Returns the number of outputs.

Implements [Function](#).

5.42.2.9 `virtual unsigned int QFunction::nParams () const [inline],[virtual]`

Returns the number of parameters.

Implements [GradientFunction](#).

5.42.2.10 `virtual void QFunction::propagate () [inline],[virtual]`

Propagates inputs to outputs.

Implements [Function](#).

5.42.2.11 `virtual void QFunction::save (XFile * file) [inline],[virtual]`

Saves the model to a file.

Reimplemented from [GradientFunction](#).

5.42.2.12 `virtual void QFunction::setInput (int i, real x) [inline],[virtual]`

Sets input *i* to value *x*.

Implements [Function](#).

5.42.2.13 `virtual void QFunction::update () [inline],[virtual]`

Updates the weights according to the derivatives.

Implements [GradientFunction](#).

5.42.3 Member Data Documentation

5.42.3.1 **ActionProperties*** QFunction::actionProperties

5.42.3.2 **GradientFunction*** QFunction::function

5.42.3.3 **real*** QFunction::input

5.42.3.4 **unsigned int** QFunction::observationDim

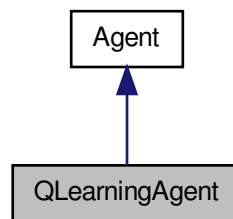
The documentation for this class was generated from the following files:

- [src/qualia/rl/QFunction.h](#)
- [src/qualia/rl/QFunction.cpp](#)

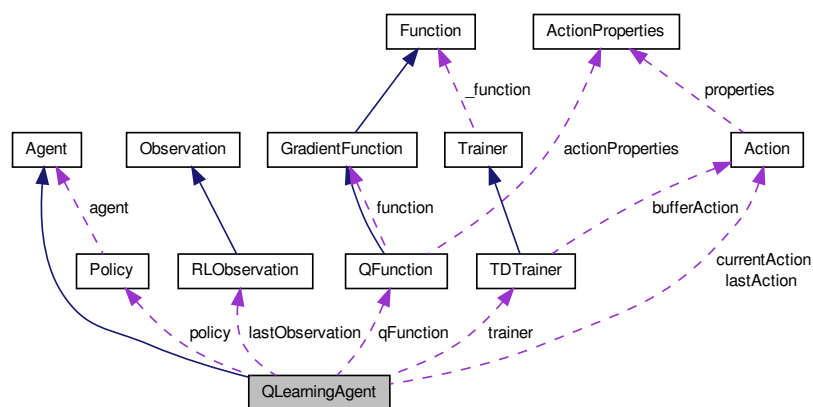
5.43 QLearningAgent Class Reference

```
#include <QLearningAgent.h>
```

Inheritance diagram for QLearningAgent:



Collaboration diagram for QLearningAgent:



Public Member Functions

- `QLearningAgent` (`QFunction` *`qFunction`, `Policy` *`policy`, unsigned int `observationDim`, `ActionProperties` *`actionProperties`, float `lambda`, float `gamma`, bool `offPolicy=false`)
- virtual `~QLearningAgent` ()
- virtual void `init` ()
Initializes the agent.
- virtual `Action` * `start` (const `Observation` *`observation`)
Chooses the first action in the episode.
- virtual `Action` * `step` (const `Observation` *`observation`)
Performs an episode step.
- virtual void `end` (const `Observation` *`observation`)
Performs the last episode step.
- virtual void `save` (`XFile` *`file`)
Saves the agent.
- virtual void `load` (`XFile` *`file`)
Loads an agent.

Public Attributes

- bool `isLearning`
Toggle learning.
- `Policy` * `policy`
The policy used by the agent.
- `QFunction` * `qFunction`
The state-action value approximator function.
- `TDTrainer` `trainer`
The trainer that we use to train the function.
- `Action` `lastAction`
- `Action` `currentAction`
- `RLObservation` `lastObservation`
- unsigned int `observationDim`

5.43.1 Detailed Description

An agent that behaves according to an action-value, TD-lambda reinforcement learning algorithm. The model allows for both on-policy (SARSA) and off-policy (Q-learning) learning.

5.43.2 Constructor & Destructor Documentation

5.43.2.1 `QLearningAgent::QLearningAgent (QFunction * qFunction, Policy * policy, unsigned int observationDim, ActionProperties * actionProperties, float lambda, float gamma, bool offPolicy = false)`

5.43.2.2 `QLearningAgent::~~QLearningAgent ()` [virtual]

5.43.3 Member Function Documentation

5.43.3.1 `void QLearningAgent::end (const Observation * observation)` [virtual]

Performs the last episode step.

Reimplemented from `Agent`.

5.43.3.2 void QLearningAgent::init () [virtual]

Initializes the agent.

Reimplemented from [Agent](#).

5.43.3.3 void QLearningAgent::load (XFile * file) [virtual]

Loads an agent.

Reimplemented from [Agent](#).

5.43.3.4 void QLearningAgent::save (XFile * file) [virtual]

Saves the agent.

Reimplemented from [Agent](#).

5.43.3.5 Action * QLearningAgent::start (const Observation * observation) [virtual]

Chooses the first action in the episode.

Implements [Agent](#).

5.43.3.6 Action * QLearningAgent::step (const Observation * observation) [virtual]

Performs an episode step.

Implements [Agent](#).

5.43.4 Member Data Documentation**5.43.4.1 Action QLearningAgent::currentAction****5.43.4.2 bool QLearningAgent::isLearning**

Toggle learning.

5.43.4.3 Action QLearningAgent::lastAction**5.43.4.4 RLObservation QLearningAgent::lastObservation****5.43.4.5 unsigned int QLearningAgent::observationDim****5.43.4.6 Policy* QLearningAgent::policy**

The policy used by the agent.

5.43.4.7 QFunction* QLearningAgent::qFunction

The state-action value approximator function.

5.43.4.8 TDTrainer QLearningAgent::trainer

The trainer that we use to train the function.

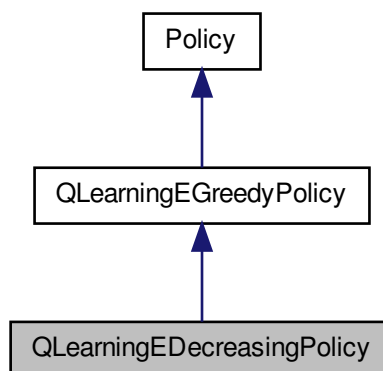
The documentation for this class was generated from the following files:

- [src/qualia/rl/QLearningAgent.h](#)
- [src/qualia/rl/QLearningAgent.cpp](#)

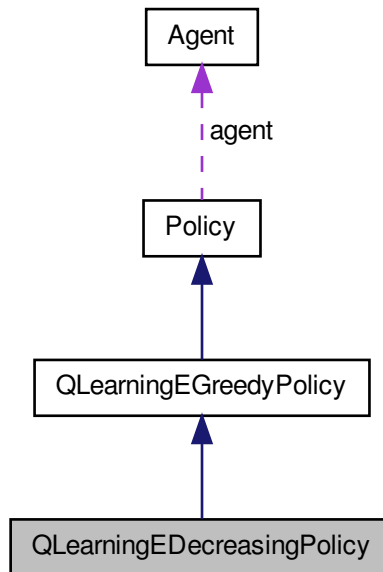
5.44 QLearningEDecreasingPolicy Class Reference

```
#include <QLearningEDecreasingPolicy.h>
```

Inheritance diagram for QLearningEDecreasingPolicy:



Collaboration diagram for QLearningEDecreasingPolicy:



Public Member Functions

- [QLearningEDecreasingPolicy](#) (float [epsilon](#), float [decreaseConstant](#))
- virtual void [init](#) ()
- virtual void [chooseAction](#) ([Action](#) *action, const [Observation](#) *observation)
- virtual float [getCurrentEpsilon](#) () const

*Returns the current epsilon value ie. $\epsilon = \text{epsilon} / (1 + t * \text{decreaseConstant})$.*

Public Attributes

- float [decreaseConstant](#)
- float [_epsilonDiv](#)

5.44.1 Constructor & Destructor Documentation

5.44.1.1 [QLearningEDecreasingPolicy::QLearningEDecreasingPolicy](#) (float *epsilon*, float *decreaseConstant*)

5.44.2 Member Function Documentation

5.44.2.1 void [QLearningEDecreasingPolicy::chooseAction](#) ([Action](#) * *action*, const [Observation](#) * *observation*)
[virtual]

This method is implemented by subclasses. It chooses an action based on given observation #observation# and puts it in #action#.

Reimplemented from [QLearningEGreedyPolicy](#).

5.44.2.2 `float QLearningEDecreasingPolicy::getCurrentEpsilon () const` [virtual]

Returns the current epsilon value ie. $\epsilon = \text{epsilon} / (1 + t * \text{decreaseConstant})$.

5.44.2.3 `void QLearningEDecreasingPolicy::init ()` [virtual]

Reimplemented from [Policy](#).

5.44.3 Member Data Documentation

5.44.3.1 `float QLearningEDecreasingPolicy::_epsilonDiv`

5.44.3.2 `float QLearningEDecreasingPolicy::_decreaseConstant`

The decrease constant. Value should be ≥ 0 , usually in $[0, 1]$. The decrease constant is applied in a similar fashion to the one for the stochastic gradient (see [NeuralNetwork.h](#)). Here, it is used to slowly decrease the epsilon value, thus allowing the agent to adapt its policy over time from being more exploratory to being more greedy.

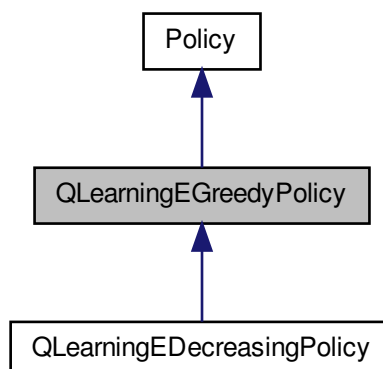
The documentation for this class was generated from the following files:

- [src/qualia/rl/QLearningEDecreasingPolicy.h](#)
- [src/qualia/rl/QLearningEDecreasingPolicy.cpp](#)

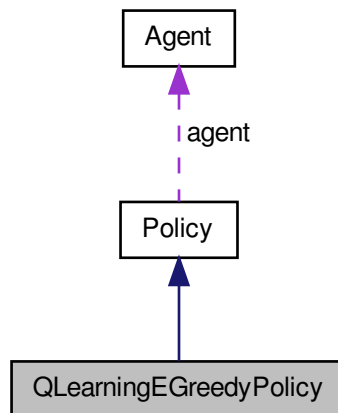
5.45 QLearningEGreedyPolicy Class Reference

```
#include <QLearningEGreedyPolicy.h>
```

Inheritance diagram for QLearningEGreedyPolicy:



Collaboration diagram for QLearningEGreedyPolicy:



Public Member Functions

- [QLearningEGreedyPolicy](#) (float [epsilon](#))
- virtual [~QLearningEGreedyPolicy](#) ()
- virtual void [chooseAction](#) ([Action](#) *action, const [Observation](#) *observation)

Public Attributes

- float [epsilon](#)
The value (should be in $[0,1]$).

5.45.1 Detailed Description

Implements the -greedy policy.

5.45.2 Constructor & Destructor Documentation

5.45.2.1 [QLearningEGreedyPolicy::QLearningEGreedyPolicy](#) (float *epsilon*)

5.45.2.2 [QLearningEGreedyPolicy::~~QLearningEGreedyPolicy](#) () [virtual]

5.45.3 Member Function Documentation

5.45.3.1 void [QLearningEGreedyPolicy::chooseAction](#) ([Action](#) * *action*, const [Observation](#) * *observation*)
[virtual]

This method is implemented by subclasses. It chooses an action based on given observation #observation# and puts it in #action#.

Implements [Policy](#).

Reimplemented in [QLearningEDecreasingPolicy](#).

5.45.4 Member Data Documentation

5.45.4.1 float QLearningEGreedyPolicy::epsilon

The value (should be in [0,1]).

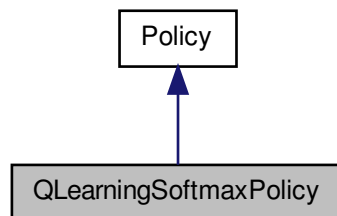
The documentation for this class was generated from the following files:

- [src/qualia/rl/QLearningEGreedyPolicy.h](#)
- [src/qualia/rl/QLearningEGreedyPolicy.cpp](#)

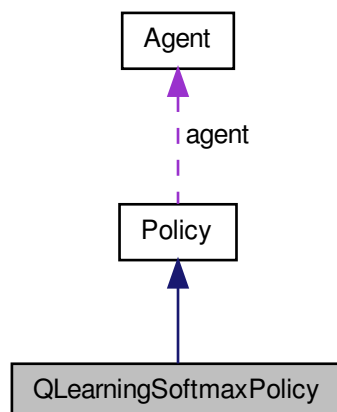
5.46 QLearningSoftmaxPolicy Class Reference

```
#include <QLearningSoftmaxPolicy.h>
```

Inheritance diagram for QLearningSoftmaxPolicy:



Collaboration diagram for QLearningSoftmaxPolicy:



Public Member Functions

- [QLearningSoftmaxPolicy](#) (float [temperature](#)=1.0, float [epsilon](#)=0.0)
- virtual [~QLearningSoftmaxPolicy](#) ()
- virtual void [chooseAction](#) ([Action](#) *action, const [Observation](#) *observation)

Public Attributes

- float [temperature](#)
- float [epsilon](#)

5.46.1 Detailed Description

Implements the softmax policy. The class contains an optional (*epsilon*) parameter that behaves in a similar fashion as the -greedy policy, meaning that there is a probability that the action is chosen randomly uniformly accross the action state and a probability of (1-) that it resorts to the softmax policy ie. picks randomly, but this time according to the softmax distribution.

5.46.2 Constructor & Destructor Documentation

5.46.2.1 [QLearningSoftmaxPolicy::QLearningSoftmaxPolicy](#) (float *temperature* = 1 . 0, float *epsilon* = 0 . 0)

5.46.2.2 [QLearningSoftmaxPolicy::~~QLearningSoftmaxPolicy](#) () [[virtual](#)]

5.46.3 Member Function Documentation

5.46.3.1 void [QLearningSoftmaxPolicy::chooseAction](#) ([Action](#) * *action*, const [Observation](#) * *observation*)
[[virtual](#)]

This method is implemented by subclasses. It chooses an action based on given observation #observation# and puts it in #action#.

Implements [Policy](#).

5.46.4 Member Data Documentation

5.46.4.1 float [QLearningSoftmaxPolicy::epsilon](#)

An optional parameter.

See Also

[QLearningEGreedyPolicy](#)

5.46.4.2 float [QLearningSoftmaxPolicy::temperature](#)

The temperature controls the "peakiness" (or "greediness") of the policy. Higher temperature means more peaky/-greedy distribution, whereas lower temperatures results in more flat / uniformly distributed choices.

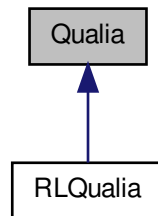
The documentation for this class was generated from the following files:

- [src/qualia/rl/QLearningSoftmaxPolicy.h](#)
- [src/qualia/rl/QLearningSoftmaxPolicy.cpp](#)

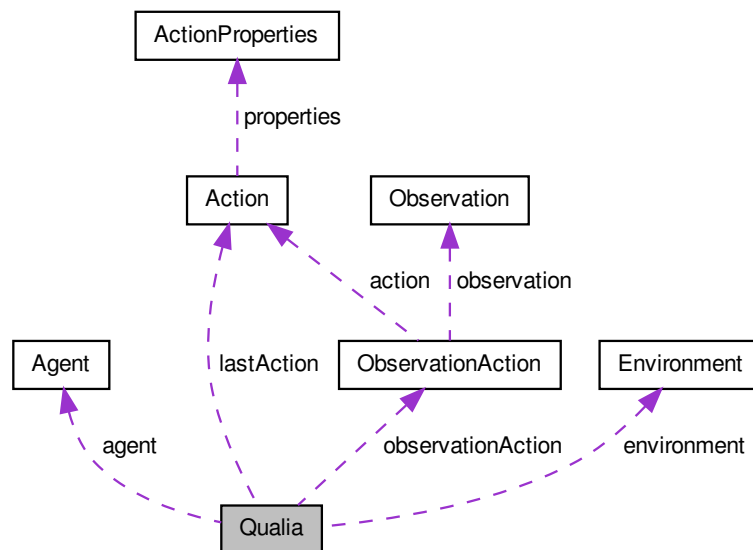
5.47 Qualia Class Reference

```
#include <Qualia.h>
```

Inheritance diagram for Qualia:



Collaboration diagram for Qualia:



Public Member Functions

- [Qualia](#) ([Agent](#) *agent, [Environment](#) *env)
- virtual [~Qualia](#) ()
- virtual void [init](#) ()
- virtual [ObservationAction](#) * [start](#) ()
- virtual [ObservationAction](#) * [step](#) ()
- virtual bool [episode](#) (unsigned int maxSteps)

Public Attributes

- [Agent](#) * [agent](#)
- [Environment](#) * [environment](#)
- [Action](#) * [lastAction](#)
- [ObservationAction](#) [observationAction](#)
- int [nSteps](#)
- int [nEpisodes](#)

5.47.1 Constructor & Destructor Documentation

5.47.1.1 [Qualia::Qualia](#) ([Agent](#) * *agent*, [Environment](#) * *env*)

5.47.1.2 [Qualia::~~Qualia](#) () [virtual]

5.47.2 Member Function Documentation

5.47.2.1 [bool](#) [Qualia::episode](#) (unsigned int *maxSteps*) [virtual]

5.47.2.2 [void](#) [Qualia::init](#) () [virtual]

Reimplemented in [RLQualia](#).

5.47.2.3 [ObservationAction](#) * [Qualia::start](#) () [virtual]

Reimplemented in [RLQualia](#).

5.47.2.4 [ObservationAction](#) * [Qualia::step](#) () [virtual]

Reimplemented in [RLQualia](#).

5.47.3 Member Data Documentation

5.47.3.1 [Agent](#)* [Qualia::agent](#)

5.47.3.2 [Environment](#)* [Qualia::environment](#)

5.47.3.3 [Action](#)* [Qualia::lastAction](#)

5.47.3.4 int [Qualia::nEpisodes](#)

5.47.3.5 int [Qualia::nSteps](#)

5.47.3.6 [ObservationAction](#) [Qualia::observationAction](#)

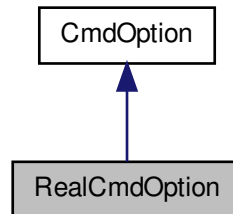
The documentation for this class was generated from the following files:

- [src/qualia/core/Qualia.h](#)
- [src/qualia/core/Qualia.cpp](#)

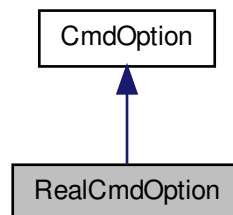
5.48 RealCmdOption Class Reference

```
#include <CmdOption.h>
```

Inheritance diagram for RealCmdOption:



Collaboration diagram for RealCmdOption:



Public Member Functions

- [RealCmdOption](#) (const char *name_, [real](#) *ptr_, [real](#) init_value_, const char *help_="", bool save_=[false](#))
- virtual void [initValue](#) ()
Initialize the value of the option.
- virtual void [printValue](#) ([DiskXFile](#) *file_)
If [is_setted#](#) is true, print the current value, else the init value.
- virtual void [read](#) (int *argc_, char ***argv_)
- virtual void [load](#) ([DiskXFile](#) *file)
- virtual void [save](#) ([DiskXFile](#) *file)
- [~RealCmdOption](#) ()

Public Attributes

- [real](#) * ptr
- [real](#) init_value

5.48.1 Detailed Description

This class defines a real command-line option.

Author

Ronan Collobert (collober@idiap.ch)

See Also

[CmdLine](#)

5.48.2 Constructor & Destructor Documentation

5.48.2.1 `RealCmdOption::RealCmdOption (const char * name_, real * ptr_, real init_value_, const char * help_ = " ", bool save_ = false)`

5.48.2.2 `RealCmdOption::~~RealCmdOption ()`

5.48.3 Member Function Documentation

5.48.3.1 `void RealCmdOption::initValue ()` [virtual]

Initialize the value of the option.

Reimplemented from [CmdOption](#).

5.48.3.2 `void RealCmdOption::load (DiskXFile * file)` [virtual]

Reimplemented from [CmdOption](#).

5.48.3.3 `void RealCmdOption::printValue (DiskXFile * file_)` [virtual]

If `is_setted#` is true, print the current value, else the init value.

Reimplemented from [CmdOption](#).

5.48.3.4 `void RealCmdOption::read (int * argc_, char *** argv_)` [virtual]

Read the option on the command line. `argv_` and `argc_` have to point of the next option after that.

Reimplemented from [CmdOption](#).

5.48.3.5 `void RealCmdOption::save (DiskXFile * file)` [virtual]

Reimplemented from [CmdOption](#).

5.48.4 Member Data Documentation

5.48.4.1 `real RealCmdOption::init_value`

5.48.4.2 `real* RealCmdOption::ptr`

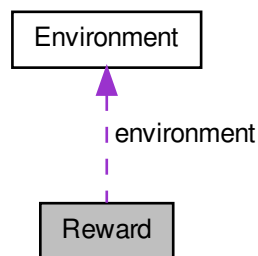
The documentation for this class was generated from the following files:

- [src/qualia/computer/CmdOption.h](#)
- [src/qualia/computer/CmdOption.cpp](#)

5.49 Reward Class Reference

```
#include <Reward.h>
```

Collaboration diagram for Reward:



Public Member Functions

- [Reward](#) ()
- virtual [~Reward](#) ()
- virtual void [setEnvironment](#) ([Environment](#) *environment_)
- virtual [real reward](#) (const [Observation](#) *before, const [Action](#) *action, const [Observation](#) *after)=0

Abstract method overridden by subclasses, implementing a reward as $r = f(s, a, s')$.

Public Attributes

- [Environment](#) * [environment](#)

5.49.1 Detailed Description

Abstract class that provides a framework for specifying reward functions. Needs to be used in pair with the [Reward-Environment](#) class.

5.49.2 Constructor & Destructor Documentation

5.49.2.1 [Reward::Reward](#) () `[inline]`

5.49.2.2 [virtual Reward::~~Reward](#) () `[inline], [virtual]`

5.49.3 Member Function Documentation

5.49.3.1 `virtual real Reward::reward (const Observation * before, const Action * action, const Observation * after)`
[pure virtual]

Abstract method overridden by subclasses, implementing a reward as $r = f(s, a, s')$.

5.49.3.2 `virtual void Reward::setEnvironment (Environment * environment)` [inline],[virtual]

5.49.4 Member Data Documentation

5.49.4.1 `Environment* Reward::environment`

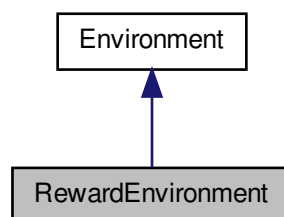
The documentation for this class was generated from the following file:

- `src/qualia/rl/Reward.h`

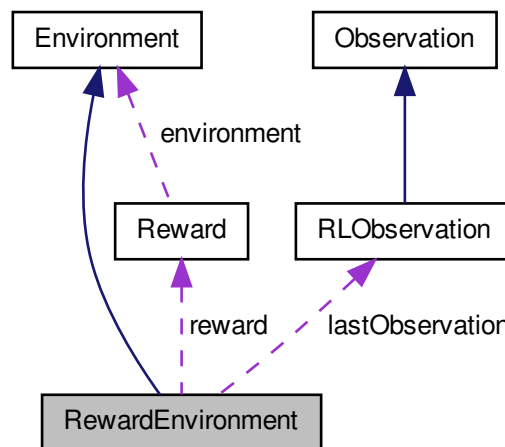
5.50 RewardEnvironment Class Reference

```
#include <RewardEnvironment.h>
```

Inheritance diagram for RewardEnvironment:



Collaboration diagram for RewardEnvironment:



Public Member Functions

- `RewardEnvironment` (unsigned int observationDim, `Reward` *reward)
- virtual `~RewardEnvironment` ()
- virtual `Observation` * `start` ()
Sends the first observation.
- virtual `Observation` * `step` (const `Action` *action)
Performs action action and returns observation.
- virtual `RLObservation` * `doStart` ()=0
Abstract method implemented by subclasses, called during `start()`.
- virtual `RLObservation` * `doAction` (const `Action` *action)=0

Public Attributes

- `Reward` * `reward`
- `RLObservation` `lastObservation`

5.50.1 Detailed Description

Abstract class that provides a framework for using reward functions. The environment performs the action but delegates the reward assignment to a `Reward` instance.

5.50.2 Constructor & Destructor Documentation

5.50.2.1 `RewardEnvironment::RewardEnvironment` (unsigned int *observationDim*, `Reward` * *reward*)

5.50.2.2 `RewardEnvironment::~~RewardEnvironment` () [virtual]

5.50.3 Member Function Documentation

5.50.3.1 `virtual RLObservation* RewardEnvironment::doAction (const Action * action)` [pure virtual]

Abstract method implemented by subclasses, called during `step()`. Performs the action *action* without setting a reward.

5.50.3.2 `virtual RLObservation* RewardEnvironment::doStart ()` [pure virtual]

Abstract method implemented by subclasses, called during `start()`.

5.50.3.3 `Observation * RewardEnvironment::start ()` [virtual]

Sends the first observation.

Implements [Environment](#).

5.50.3.4 `Observation * RewardEnvironment::step (const Action * action)` [virtual]

Performs action *action* and returns observation.

Implements [Environment](#).

5.50.4 Member Data Documentation

5.50.4.1 `RLObservation RewardEnvironment::lastObservation`

5.50.4.2 `Reward* RewardEnvironment::reward`

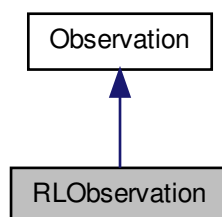
The documentation for this class was generated from the following files:

- [src/qualia/rl/RewardEnvironment.h](#)
- [src/qualia/rl/RewardEnvironment.cpp](#)

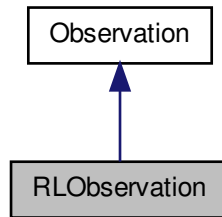
5.51 RLObservation Class Reference

```
#include <RLObservation.h>
```

Inheritance diagram for RLObservation:



Collaboration diagram for RLObservation:



Public Member Functions

- [RLObservation](#) (unsigned int [dim](#))
- [Observation](#) & [copyFrom](#) (const [Observation](#) &src)
Copies data from src.
- virtual void [saveData](#) ([XFile](#) *file) const
Save data to file (in binary format).
- virtual void [loadData](#) ([XFile](#) *file)
Loads data from file (in binary format). Dimension needs to be known in advance.

Public Attributes

- [real reward](#)
The reward component.

5.51.1 Detailed Description

An observation that contains a reward, for use in Reinforcement Learning (RL).

5.51.2 Constructor & Destructor Documentation

5.51.2.1 `RLObservation::RLObservation (unsigned int dim)`

5.51.3 Member Function Documentation

5.51.3.1 `Observation & RLObservation::copyFrom (const Observation & src)` `[virtual]`

Copies data from *src*.

Reimplemented from [Observation](#).

5.51.3.2 `void RLObservation::loadData (XFile * file)` `[virtual]`

Loads data from *file* (in binary format). Dimension needs to be known in advance.

Reimplemented from [Observation](#).

5.51.3.3 void RLObservation::saveData (XFile * *file*) const [virtual]

Save data to *file* (in binary format).

Reimplemented from [Observation](#).

5.51.4 Member Data Documentation

5.51.4.1 real RLObservation::reward

The reward component.

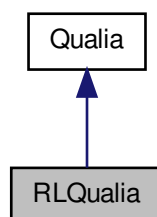
The documentation for this class was generated from the following files:

- [src/qualia/rl/RLObservation.h](#)
- [src/qualia/rl/RLObservation.cpp](#)

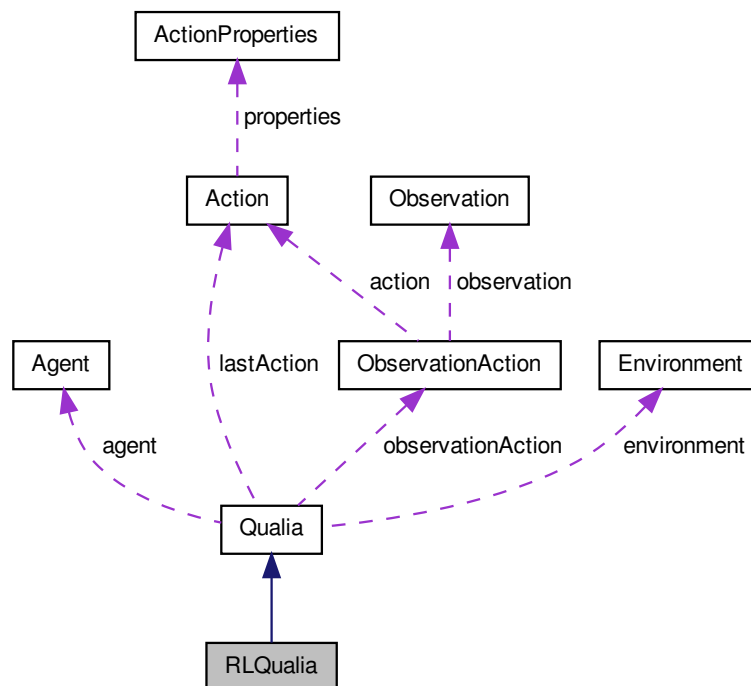
5.52 RLQualia Class Reference

```
#include <RLQualia.h>
```

Inheritance diagram for RLQualia:



Collaboration diagram for RLQualia:



Public Member Functions

- [RLQualia](#) ([Agent](#) *agent, [Environment](#) *env)
- virtual void [init](#) ()
- virtual [ObservationAction](#) * [start](#) ()
- virtual [ObservationAction](#) * [step](#) ()

Public Attributes

- [real](#) totalReward

5.52.1 Constructor & Destructor Documentation

5.52.1.1 [RLQualia::RLQualia](#) ([Agent](#) * agent, [Environment](#) * env)

5.52.2 Member Function Documentation

5.52.2.1 void [RLQualia::init](#) () [virtual]

Reimplemented from [Qualia](#).

5.52.2.2 [ObservationAction](#) * [RLQualia::start](#) () [virtual]

Reimplemented from [Qualia](#).

5.52.2.3 **ObservationAction** * RLQualia::step () [virtual]

Reimplemented from [Qualia](#).

5.52.3 **Member Data Documentation**5.52.3.1 **real** RLQualia::totalReward

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

- src/qualia/rl/[RLQualia.h](#)
- src/qualia/rl/[RLQualia.cpp](#)

5.53 **MapperConnector::SignalData Struct Reference**

```
#include <MapperConnector.h>
```

Public Member Functions

- [SignalData](#) (mapper_signal sig_, int n_, bool isBlocking_, float *initialData=0)
- [~SignalData](#) ()

Public Attributes

- mapper_signal [sig](#)
- float * [data](#)
- int [n](#)
- bool [isBlocking](#)
- bool [flag](#)

5.53.1 **Constructor & Destructor Documentation**

5.53.1.1 **MapperConnector::SignalData::SignalData** (mapper_signal *sig_*, int *n_*, bool *isBlocking_*, float * *initialData* = 0)

5.53.1.2 **MapperConnector::SignalData::~~SignalData** ()

5.53.2 **Member Data Documentation**

5.53.2.1 **float*** MapperConnector::SignalData::data

5.53.2.2 **bool** MapperConnector::SignalData::flag

5.53.2.3 **bool** MapperConnector::SignalData::isBlocking

5.53.2.4 **int** MapperConnector::SignalData::n

5.53.2.5 **mapper_signal** MapperConnector::SignalData::sig

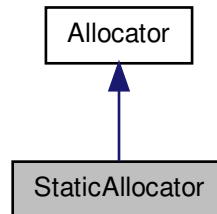
The documentation for this struct was generated from the following files:

- src/qualia/plugins/mapper/[MapperConnector.h](#)
- src/qualia/plugins/mapper/[MapperConnector.cpp](#)

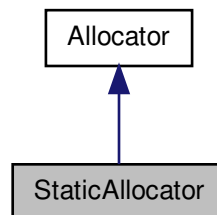
5.54 StaticAllocator Class Reference

```
#include <StaticAllocator.h>
```

Inheritance diagram for StaticAllocator:



Collaboration diagram for StaticAllocator:



Public Member Functions

- `StaticAllocator` (unsigned char *`buffer`, size_t size)

Public Attributes

- unsigned char * `buffer`
- size_t `bufferSize`
- unsigned int `bufferIdx`
- unsigned int `nLeaks`
- unsigned char * `lastLeak`

Protected Member Functions

- virtual void * `malloc` (size_t size)
- virtual void * `realloc` (void *ptr, size_t size)

- virtual void [free](#) (void *ptr)
- virtual void [freeAll](#) ()

5.54.1 Detailed Description

An allocator (see [Allocator](#)) that "allocates" memory based on a pre-allocated static memory pool/buffer. Useful to manage memory on architectures that don't support well dynamic allocation (such as AVR-based systems). On such systems, it is usually recommended NOT to use dynamic allocation to avoid problems.

WARNING: Calling [StaticAllocator::free\(\)](#) does NOT free the pointer at all. You should in fact NEVER have to call that function (because the memory then becomes completely useless). The object keeps track of any calls to [free\(\)](#) by incrementing the nLeaks counter. The variable lastLeak is also updated with the value of the pointer on which [free\(\)](#) was called last.

NOTE: An alternative is to tune the heap start and end in malloc. <http://www.nongnu.org/avr-libc/user-manual/malloc.html>

Usage:

```
unsigned char mybuffer[100];
StaticAllocator alloc(mybuffer, 100);
int* myarray = (int*)alloc.malloc(10*sizeof(int));
Object myobject = new(&alloc) Object(1,2);
```

5.54.2 Constructor & Destructor Documentation

5.54.2.1 [StaticAllocator::StaticAllocator](#) (unsigned char * *buffer*, size_t *size*)

5.54.3 Member Function Documentation

5.54.3.1 void [StaticAllocator::free](#) (void * *ptr*) [protected], [virtual]

Reimplemented from [Allocator](#).

5.54.3.2 void [StaticAllocator::freeAll](#) () [protected], [virtual]

5.54.3.3 void * [StaticAllocator::malloc](#) (size_t *size*) [protected], [virtual]

Reimplemented from [Allocator](#).

5.54.3.4 void * [StaticAllocator::realloc](#) (void * *ptr*, size_t *size*) [protected], [virtual]

Reimplemented from [Allocator](#).

5.54.4 Member Data Documentation

5.54.4.1 unsigned char* [StaticAllocator::buffer](#)

5.54.4.2 unsigned int [StaticAllocator::bufferIdx](#)

5.54.4.3 size_t [StaticAllocator::bufferSize](#)

5.54.4.4 unsigned char* [StaticAllocator::lastLeak](#)

5.54.4.5 unsigned int StaticAllocator::nLeaks

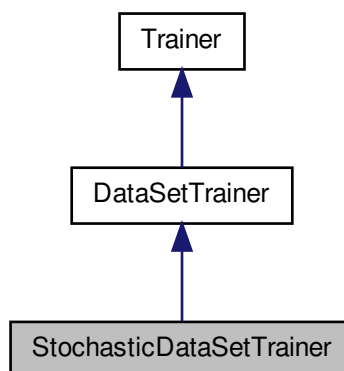
The documentation for this class was generated from the following files:

- src/qualia/core/[StaticAllocator.h](#)
- src/qualia/core/[StaticAllocator.cpp](#)

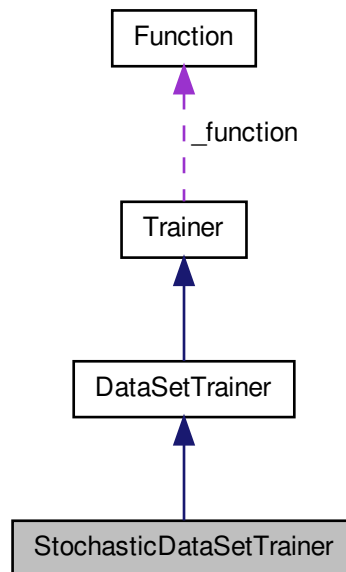
5.55 StochasticDataSetTrainer Class Reference

```
#include <StochasticDataSetTrainer.h>
```

Inheritance diagram for StochasticDataSetTrainer:



Collaboration diagram for StochasticDataSetTrainer:



Public Member Functions

- [StochasticDataSetTrainer](#) ([Function](#) *function)
- virtual [~StochasticDataSetTrainer](#) ()
- virtual void [trainExample](#) ([real](#) *example)=0
- virtual void [_doTrainEpisode](#) ([DataSet](#) *data)

This method should be overridden by subclasses to train a single episode over the dataset.

Additional Inherited Members

5.55.1 Detailed Description

Dataset trainer that trains on one example at a time.

5.55.2 Constructor & Destructor Documentation

5.55.2.1 [StochasticDataSetTrainer::StochasticDataSetTrainer](#) ([Function](#) * *function*)

5.55.2.2 [StochasticDataSetTrainer::~~StochasticDataSetTrainer](#) () [virtual]

5.55.3 Member Function Documentation

5.55.3.1 void [StochasticDataSetTrainer::_doTrainEpisode](#) ([DataSet](#) * *data*) [virtual]

This method should be overridden by subclasses to train a single episode over the dataset.

Implements [DataSetTrainer](#).

5.55.3.2 `virtual void StochasticDataSetTrainer::trainExample (real * example)` [pure virtual]

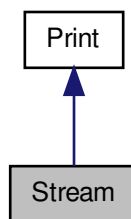
The documentation for this class was generated from the following files:

- [src/qualia/learning/StochasticDataSetTrainer.h](#)
- [src/qualia/learning/StochasticDataSetTrainer.cpp](#)

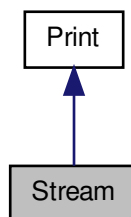
5.56 Stream Class Reference

```
#include <Stream.h>
```

Inheritance diagram for Stream:



Collaboration diagram for Stream:



Public Member Functions

- virtual int [available](#) ()=0
- virtual int [read](#) ()=0
- virtual int [peek](#) ()=0
- virtual void [flush](#) ()=0
- [Stream](#) ()
- void [setTimeout](#) (unsigned long timeout)
- bool [find](#) (char *target)

- bool [find](#) (char *target, size_t length)
- bool [findUntil](#) (char *target, char *terminator)
- bool [findUntil](#) (char *target, size_t targetLen, char *terminate, size_t termLen)
- long [parseInt](#) ()
- float [parseFloat](#) ()
- size_t [readBytes](#) (char *buffer, size_t length)
- size_t [readBytesUntil](#) (char terminator, char *buffer, size_t length)
- String [readString](#) ()
- String [readStringUntil](#) (char terminator)

Protected Member Functions

- long [parseInt](#) (char skipChar)
- float [parseFloat](#) (char skipChar)

Private Member Functions

- int [timedRead](#) ()
- int [timedPeek](#) ()
- int [peekNextDigit](#) ()

Private Attributes

- unsigned long [_timeout](#)
- unsigned long [_startMillis](#)

5.56.1 Constructor & Destructor Documentation

5.56.1.1 `Stream::Stream ()` `[inline]`

5.56.2 Member Function Documentation

5.56.2.1 `virtual int Stream::available ()` `[pure virtual]`

5.56.2.2 `bool Stream::find (char * target)`

5.56.2.3 `bool Stream::find (char * target, size_t length)`

5.56.2.4 `bool Stream::findUntil (char * target, char * terminator)`

5.56.2.5 `bool Stream::findUntil (char * target, size_t targetLen, char * terminate, size_t termLen)`

5.56.2.6 `virtual void Stream::flush ()` `[pure virtual]`

5.56.2.7 `float Stream::parseFloat ()`

5.56.2.8 `float Stream::parseFloat (char skipChar)` `[protected]`

5.56.2.9 `long Stream::parseInt ()`

5.56.2.10 `long Stream::parseInt (char skipChar)` `[protected]`

5.56.2.11 `virtual int Stream::peek ()` `[pure virtual]`

5.56.2.12 `int Stream::peekNextDigit ()` [private]

5.56.2.13 `virtual int Stream::read ()` [pure virtual]

5.56.2.14 `size_t Stream::readBytes (char * buffer, size_t length)`

5.56.2.15 `size_t Stream::readBytesUntil (char terminator, char * buffer, size_t length)`

5.56.2.16 `String Stream::readString ()`

5.56.2.17 `String Stream::readStringUntil (char terminator)`

5.56.2.18 `void Stream::setTimeout (unsigned long timeout)`

5.56.2.19 `int Stream::timedPeek ()` [private]

5.56.2.20 `int Stream::timedRead ()` [private]

5.56.3 Member Data Documentation

5.56.3.1 `unsigned long Stream::_startMillis` [private]

5.56.3.2 `unsigned long Stream::_timeout` [private]

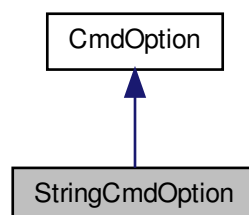
The documentation for this class was generated from the following files:

- `src/qualia/compat/Stream.h`
- `src/qualia/compat/Stream.cpp`

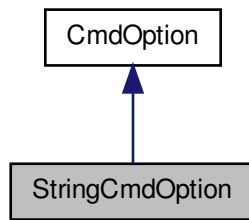
5.57 StringCmdOption Class Reference

```
#include <CmdOption.h>
```

Inheritance diagram for StringCmdOption:



Collaboration diagram for StringCmdOption:



Public Member Functions

- [StringCmdOption](#) (const char *name_, char **ptr_, const char *init_value_, const char *help_="", bool save_ = false)
- virtual void [initValue](#) ()
Initialize the value of the option.
- virtual void [printValue](#) ([DiskXFile](#) *file_)
If [is_setted#](#) is true, print the current value, else the init value.
- virtual void [read](#) (int *argc_, char ***argv_)
- virtual void [load](#) ([DiskXFile](#) *file)
- virtual void [save](#) ([DiskXFile](#) *file)
- [~StringCmdOption](#) ()

Public Attributes

- char ** [ptr](#)
- char * [init_value](#)

5.57.1 Detailed Description

This class defines a string command-line option.

Author

Ronan Collobert (collober@idiap.ch)

See Also

[CmdLine](#)

5.57.2 Constructor & Destructor Documentation

5.57.2.1 `StringCmdOption::StringCmdOption (const char * name_, char ** ptr_, const char * init_value_, const char * help_ = " ", bool save_ = false)`

5.57.2.2 `StringCmdOption::~~StringCmdOption ()`

5.57.3 Member Function Documentation

5.57.3.1 void StringCmdOption::initValue () [virtual]

Initialize the value of the option.

Reimplemented from [CmdOption](#).

5.57.3.2 void StringCmdOption::load (DiskXFile * file) [virtual]

Reimplemented from [CmdOption](#).

5.57.3.3 void StringCmdOption::printValue (DiskXFile * file_) [virtual]

If [is_setted#](#) is true, print the current value, else the init value.

Reimplemented from [CmdOption](#).

5.57.3.4 void StringCmdOption::read (int * argc_, char *** argv_) [virtual]

Read the option on the command line. argv_ and argc_ have to point of the next option after that.

Reimplemented from [CmdOption](#).

5.57.3.5 void StringCmdOption::save (DiskXFile * file) [virtual]

Reimplemented from [CmdOption](#).

5.57.4 Member Data Documentation

5.57.4.1 char* StringCmdOption::init_value

5.57.4.2 char** StringCmdOption::ptr

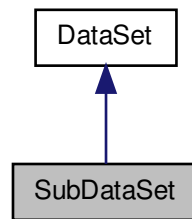
The documentation for this class was generated from the following files:

- src/qualia/computer/[CmdOption.h](#)
- src/qualia/computer/[CmdOption.cpp](#)

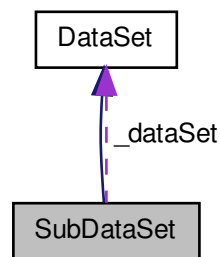
5.58 SubDataSet Class Reference

```
#include <SubDataSet.h>
```


Inheritance diagram for SubDataSet:



Collaboration diagram for SubDataSet:



Public Member Functions

- [SubDataSet](#) ([DataSet](#) *dataSet, int *indices, int nExamples)
- virtual [~SubDataSet](#) ()
- virtual void [init](#) ()
- virtual void [reset](#) ()

To be called before every episode.

- virtual void [setExample](#) (int t)
Sets this->example to given index t.

Static Public Member Functions

- static [SubDataSet](#) * [getShuffledDataSet](#) ([DataSet](#) *dataSet, int nExamples=-1)
- static void [getStandardDataSets](#) ([SubDataSet](#) **train, [SubDataSet](#) **valid, [SubDataSet](#) **test, [DataSet](#) *dataSet, float propTrain=0.5f, float propValid=0.2f, float propTest=0.3f)

Public Attributes

- [DataSet](#) * [_dataSet](#)
- int * [_indices](#)

5.58.1 Constructor & Destructor Documentation

5.58.1.1 `SubDataSet::SubDataSet (DataSet * dataSet, int * indices, int nExamples)`

5.58.1.2 `SubDataSet::~~SubDataSet ()` [virtual]

5.58.2 Member Function Documentation

5.58.2.1 `SubDataSet * SubDataSet::getShuffledDataSet (DataSet * dataSet, int nExamples = -1)` [static]

5.58.2.2 `void SubDataSet::getStandardDataSets (SubDataSet ** train, SubDataSet ** valid, SubDataSet ** test, DataSet * dataSet, float propTrain = 0.5f, float propValid = 0.2f, float propTest = 0.3f)` [static]

5.58.2.3 `void SubDataSet::init ()` [virtual]

To be called before training. Should allocate example, among other things. Default version allocates example to the size of *dim*.

Reimplemented from [DataSet](#).

5.58.2.4 `void SubDataSet::reset ()` [virtual]

To be called before every episode.

Reimplemented from [DataSet](#).

5.58.2.5 `void SubDataSet::setExample (int t)` [virtual]

Sets this->example to given index *t*.

Implements [DataSet](#).

5.58.3 Member Data Documentation

5.58.3.1 `DataSet* SubDataSet::_dataSet`

5.58.3.2 `int* SubDataSet::_indices`

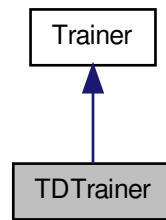
The documentation for this class was generated from the following files:

- `src/qualia/computer/SubDataSet.h`
- `src/qualia/computer/SubDataSet.cpp`

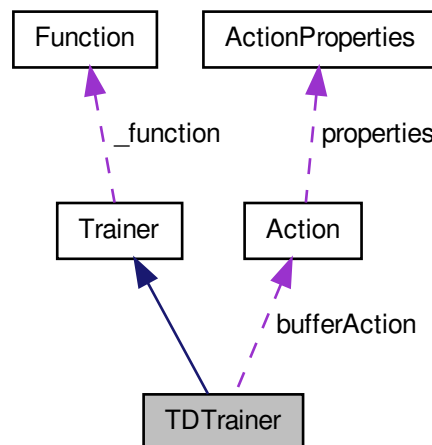
5.59 TDTrainer Class Reference

```
#include <TDTrainer.h>
```

Inheritance diagram for TDTrainer:



Collaboration diagram for TDTrainer:



Public Member Functions

- **TDTrainer** (*QFunction* *qFunction, unsigned int *observationDim*, *ActionProperties* *actionProperties, float *lambda*, float *gamma*, bool *offPolicy*=false)
- virtual **~TDTrainer** ()
- virtual void **init** ()
- virtual void **step** (const *RLObservation* *lastObservation, const *Action* *lastAction, const *RLObservation* *observation, const *Action* *action)

Performs a training step.

Public Attributes

- *real* * **eTraces**

The elligibility traces.

- float [gamma](#)
- float [lambda](#)
- bool [offPolicy](#)
- Action [bufferAction](#)
- unsigned int [observationDim](#)
- unsigned int [actionDim](#)

5.59.1 Detailed Description

This class trains a [QFunction](#) using the Temporal-Difference (TD-) algorithm.

5.59.2 Constructor & Destructor Documentation

5.59.2.1 `TDTrainer::TDTrainer (QFunction * qFunction, unsigned int observationDim, ActionProperties * actionProperties, float lambda, float gamma, bool offPolicy = false)`

5.59.2.2 `TDTrainer::~~TDTrainer () [virtual]`

5.59.3 Member Function Documentation

5.59.3.1 `void TDTrainer::init () [virtual]`

Reimplemented from [Trainer](#).

5.59.3.2 `void TDTrainer::step (const RLObservation * lastObservation, const Action * lastAction, const RLObservation * observation, const Action * action) [virtual]`

Performs a training step.

5.59.4 Member Data Documentation

5.59.4.1 unsigned int `TDTrainer::actionDim`

5.59.4.2 Action `TDTrainer::bufferAction`

5.59.4.3 `real*` `TDTrainer::eTraces`

The eligibility traces.

5.59.4.4 float `TDTrainer::gamma`

Discounting factor. Value should be in [0, 1], typical value in [0.9, 1). The discount factor determines the importance of future rewards. A factor of 0 will make the agent "opportunistic" by only considering current rewards, while a factor approaching 1 will make it strive for a long-term high reward. If the discount factor meets or exceeds 1, the Q values may diverge. Source: http://en.wikipedia.org/wiki/Q-learning#Discount_factor

5.59.4.5 float `TDTrainer::lambda`

Trace decay. Value should be in [0, 1], typical value in (0, 0.1]. Heuristic parameter controlling the temporal credit assignment of how an error detected at a given time step feeds back to correct previous estimates. When $\lambda = 0$, no feedback occurs beyond the current time step, while when $\lambda = 1$, the error feeds back without decay arbitrarily far in time. Intermediate values of λ provide a smooth way to interpolate between these two limiting cases. Source: <http://www.research.ibm.com/massive/tdl.html>

5.59.4.6 unsigned int TDTrainer::observationDim

5.59.4.7 bool TDTrainer::offPolicy

Controls whether to use the off-policy learning algorithm (Q-Learning) or the on-policy algorithm (Sarsa). Default value: false ie. on-policy (Sarsa) learning NOTE: Off-policy learning should be used at all time when training on a pre-generated dataset. When the agent is trained online (eg. in real time) the on-policy algorithm will result in the agent showing a better online performance at the expense of finding a sub-optimal solution. On the opposite, the off-policy strategy will converge to the optimal solution but will usually show a lower online performance as it will more often make mistakes.

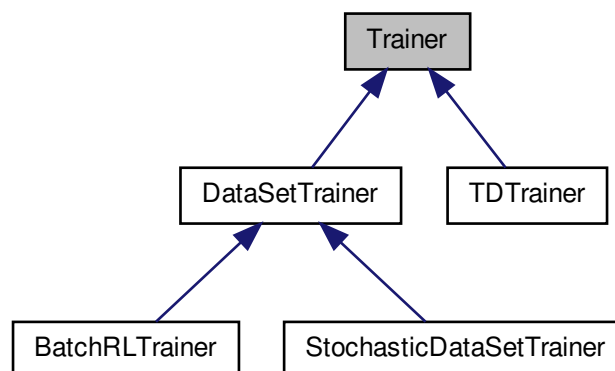
The documentation for this class was generated from the following files:

- [src/qualia/rl/TDTrainer.h](#)
- [src/qualia/rl/TDTrainer.cpp](#)

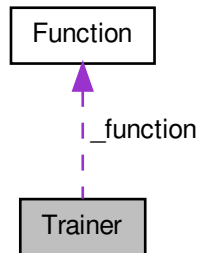
5.60 Trainer Class Reference

```
#include <Trainer.h>
```

Inheritance diagram for Trainer:



Collaboration diagram for Trainer:



Public Member Functions

- [Trainer](#) ([Function](#) *function)
Constructor.
- virtual [~Trainer](#) ()
- int [nEpisodes](#) () const
- virtual void [init](#) ()

Public Attributes

- [Function](#) * [_function](#)
The function this [Trainer](#) is optimizing.
- int [_nEpisodes](#)
The number of episodes this trainer went through (read-only).

5.60.1 Detailed Description

A trainer is able to train a function. This class does almost nothing, all the logic being deferred to subclasses.

Main subclasses:

- [TDTrainer](#)
- [DataSetTrainer](#)

5.60.2 Constructor & Destructor Documentation

5.60.2.1 Trainer::Trainer ([Function](#) * *function*)

Constructor.

5.60.2.2 Trainer::~~Trainer () [virtual]

5.60.3 Member Function Documentation

5.60.3.1 `void Trainer::init () [virtual]`

Reimplemented in [TDTrainer](#), and [BatchRLTrainer](#).

5.60.3.2 `int Trainer::nEpisodes () const [inline]`

5.60.4 Member Data Documentation

5.60.4.1 `Function* Trainer::_function`

The function this [Trainer](#) is optimizing.

5.60.4.2 `int Trainer::_nEpisodes`

The number of episodes this trainer went through (read-only).

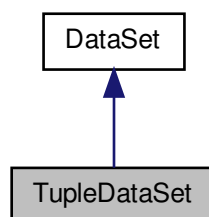
The documentation for this class was generated from the following files:

- [src/qualia/learning/Trainer.h](#)
- [src/qualia/learning/Trainer.cpp](#)

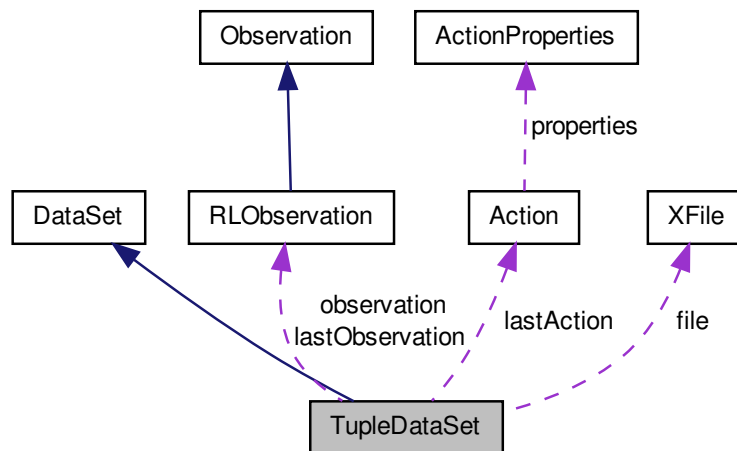
5.61 TupleDataSet Class Reference

```
#include <TupleDataSet.h>
```

Inheritance diagram for TupleDataSet:



Collaboration diagram for TupleDataSet:



Public Member Functions

- `TupleDataSet (XFile *file, unsigned int observationDim, ActionProperties *actionProperties)`
- `virtual ~TupleDataSet ()`
- `virtual void init ()`
- `virtual void reset ()`
To be called before every episode.
- `virtual void setExample (int t)`
Sets this->example to given index t.
- `unsigned int observationDim () const`
- `unsigned int actionDim () const`

Static Public Member Functions

- `static void tupleFromExample (RLObservation *lastObservation, Action *lastAction, real *reward, RLObservation *observation, const real *example)`
- `static void tupleToExample (real *example, const RLObservation &lastObservation, const Action &lastAction, real reward, const RLObservation &observation)`

Public Attributes

- `XFile * file`
- `RLObservation lastObservation`
- `Action lastAction`
- `RLObservation observation`
- `int currentExampleIndex`

5.61.1 Detailed Description

A data set containing reinforcement learning (s,a,r,s') tuples (for batch learning). It reads from an [XFile](#) in the [Qualia RAW](#) format (cf. [FileExportEnvironment](#)).

See Also

[BatchRLTrainer](#)

5.61.2 Constructor & Destructor Documentation

5.61.2.1 `TupleDataSet::TupleDataSet (XFile * file, unsigned int observationDim, ActionProperties * actionProperties)`

5.61.2.2 `TupleDataSet::~~TupleDataSet () [virtual]`

5.61.3 Member Function Documentation

5.61.3.1 `unsigned int TupleDataSet::actionDim () const [inline]`

5.61.3.2 `void TupleDataSet::init () [virtual]`

To be called before training. Should allocate example, among other things. Default version allocates example to the size of *dim*.

Reimplemented from [DataSet](#).

5.61.3.3 `unsigned int TupleDataSet::observationDim () const [inline]`

5.61.3.4 `void TupleDataSet::reset () [virtual]`

To be called before every episode.

Reimplemented from [DataSet](#).

5.61.3.5 `void TupleDataSet::setExample (int t) [virtual]`

Sets this->example to given index *t*.

Implements [DataSet](#).

5.61.3.6 `void TupleDataSet::tupleFromExample (RLObservation * lastObservation, Action * lastAction, real * reward, RLObservation * observation, const real * example) [static]`

5.61.3.7 `void TupleDataSet::tupleToExample (real * example, const RLObservation & lastObservation, const Action & lastAction, real reward, const RLObservation & observation) [static]`

5.61.4 Member Data Documentation

5.61.4.1 `int TupleDataSet::currentExampleIndex`

5.61.4.2 `XFile* TupleDataSet::file`

5.61.4.3 `Action TupleDataSet::lastAction`

5.61.4.4 `RLObservation TupleDataSet::lastObservation`

5.61.4.5 RLObservation TupleDataSet::observation

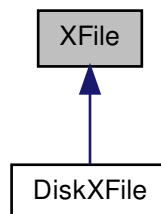
The documentation for this class was generated from the following files:

- [src/qualia/rl/TupleDataSet.h](#)
- [src/qualia/rl/TupleDataSet.cpp](#)

5.62 XFile Class Reference

```
#include <XFile.h>
```

Inheritance diagram for XFile:



Public Member Functions

- [XFile](#) ()
- virtual int [read](#) (void *ptr, int block_size, int n_blocks)=0
- virtual int [write](#) (const void *ptr, int block_size, int n_blocks)=0
- int [taggedRead](#) (void *ptr, int block_size, int n_blocks, const char *tag)
- int [taggedWrite](#) (const void *ptr, int block_size, int n_blocks, const char *tag)
Write and write the tag/the size.
- virtual int [eof](#) ()=0
Are we at the end ?
- virtual int [flush](#) ()=0
Flush the file.
- virtual int [seek](#) (long offset, int whence)=0
Seek.
- virtual long [tell](#) ()=0
Tell me where am I...
- virtual void [rewind](#) ()=0
Rewind.
- virtual char * [gets](#) (char *dest, int size_)=0
Print some text.
- virtual long [size](#) ()
- virtual [~XFile](#) ()

5.62.1 Detailed Description

[XFile](#). A File which could be anything. The syntax (and results) for method is very similar to C FILE. (Except for the FILE pointer which is not given in the parameters!).

Author

Ronan Collobert (collobert@idiap.ch)

5.62.2 Constructor & Destructor Documentation

5.62.2.1 `XFile::XFile ()`

5.62.2.2 `XFile::~XFile ()` [virtual]

5.62.3 Member Function Documentation

5.62.3.1 `virtual int XFile::eof ()` [pure virtual]

Are we at the end ?

Implemented in [DiskXFile](#).

5.62.3.2 `virtual int XFile::flush ()` [pure virtual]

Flush the file.

Implemented in [DiskXFile](#).

5.62.3.3 `virtual char* XFile::gets (char * dest, int size_)` [pure virtual]

[Print](#) some text.

Scan some text. Get one line (read at most #size_# characters).

Implemented in [DiskXFile](#).

5.62.3.4 `virtual int XFile::read (void * ptr, int block_size, int n_blocks)` [pure virtual]

Read something. Returns the number of blocks read or a value < 0 if there was an error.

Implemented in [DiskXFile](#).

5.62.3.5 `virtual void XFile::rewind ()` [pure virtual]

Rewind.

Implemented in [DiskXFile](#).

5.62.3.6 `virtual int XFile::seek (long offset, int whence)` [pure virtual]

Seek.

Implemented in [DiskXFile](#).

5.62.3.7 `long XFile::size () [virtual]`

5.62.3.8 `int XFile::taggedRead (void * ptr, int block_size, int n_blocks, const char * tag)`

Read and check the tag/the size. To be used with [taggedWrite\(\)](#)#. If the tag and the size readed doesn't correspond to the given tag and size, an error will occur and the returned value will be (-1).

5.62.3.9 `int XFile::taggedWrite (const void * ptr, int block_size, int n_blocks, const char * tag)`

Write and write the tag/the size.

5.62.3.10 `virtual long XFile::tell () [pure virtual]`

Tell me where am I...

Implemented in [DiskXFile](#).

5.62.3.11 `virtual int XFile::write (const void * ptr, int block_size, int n_blocks) [pure virtual]`

Write. Returns the number of blocks written or a value < 0 if there was an error.

Implemented in [DiskXFile](#).

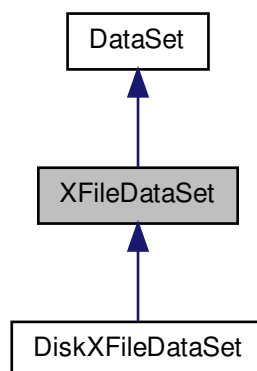
The documentation for this class was generated from the following files:

- [src/qualia/core/XFile.h](#)
- [src/qualia/core/XFile.cpp](#)

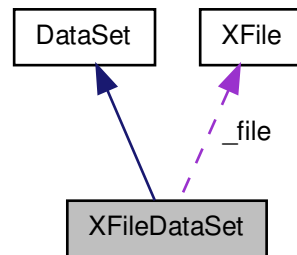
5.63 XFileDataSet Class Reference

```
#include <XFileDataSet.h>
```

Inheritance diagram for XFileDataSet:



Collaboration diagram for XFileDataSet:



Public Member Functions

- [XFileDataSet](#) ([XFile](#) *file)
- virtual [~XFileDataSet](#) ()
- virtual void [init](#) ()
- virtual void [reset](#) ()
To be called before every episode.
- virtual void [setExample](#) (int t)
Sets this->example to given index t.

Public Attributes

- [XFile](#) * [_file](#)
The file ([XFile](#)) that stores the dataset.
- int [_currentExampleIndex](#)
The index of the current example (internal use).

5.63.1 Detailed Description

A [DataSet](#) that reads data from an [XFile](#).

5.63.2 Constructor & Destructor Documentation

5.63.2.1 [XFileDataSet::XFileDataSet](#) ([XFile](#) * file)

5.63.2.2 [XFileDataSet::~~XFileDataSet](#) () [virtual]

5.63.3 Member Function Documentation

5.63.3.1 void [XFileDataSet::init](#) () [virtual]

To be called before training. Should allocate example, among other things. Default version allocates example to the size of *dim*.

Reimplemented from [DataSet](#).

Reimplemented in [DiskXFileDataSet](#).

5.63.3.2 void XFileDataSet::reset () [virtual]

To be called before every episode.

Reimplemented from [DataSet](#).

Reimplemented in [DiskXFileDataSet](#).

5.63.3.3 void XFileDataSet::setExample (int *t*) [virtual]

Sets this->example to given index *t*.

Implements [DataSet](#).

Reimplemented in [DiskXFileDataSet](#).

5.63.4 Member Data Documentation

5.63.4.1 int XFileDataSet::_currentExampleIndex

The index of the current example (internal use).

5.63.4.2 XFile* XFileDataSet::_file

The file ([XFile](#)) that stores the dataset.

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

- [src/qualia/learning/XFileDataSet.h](#)
- [src/qualia/learning/XFileDataSet.cpp](#)

Chapter 6

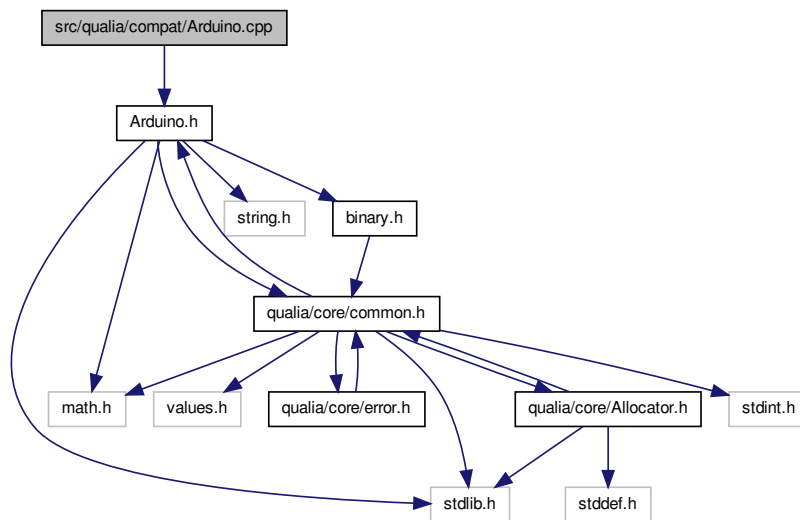
File Documentation

6.1 src/mainpage.dox File Reference

6.2 src/qualia/compat/Arduino.cpp File Reference

```
#include "Arduino.h"
```

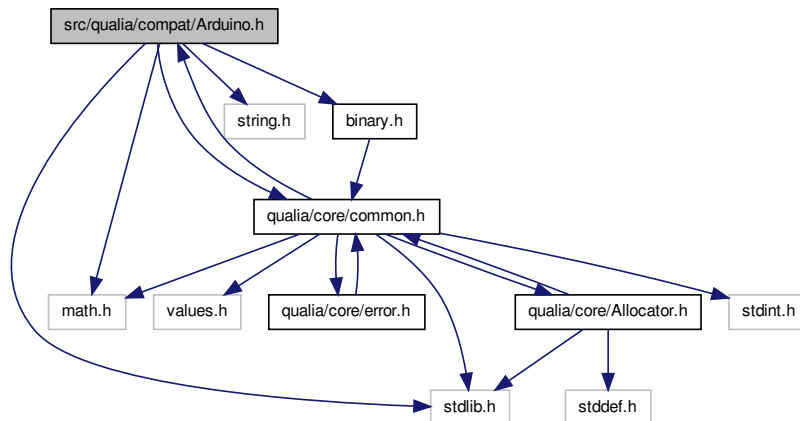
Include dependency graph for Arduino.cpp:



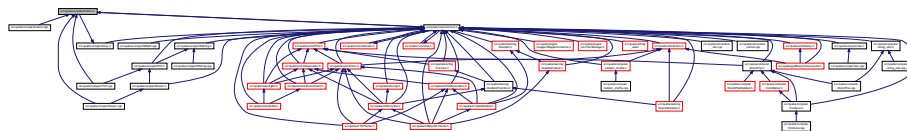
6.3 src/qualia/compat/Arduino.h File Reference

```
#include <qualia/core/common.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "binary.h"
```

Include dependency graph for Arduino.h:



This graph shows which files directly or indirectly include this file:



Macros

- `#define true 0x1`
- `#define false 0x0`
- `#define PI 3.1415926535897932384626433832795`
- `#define HALF_PI 1.5707963267948966192313216916398`
- `#define TWO_PI 6.283185307179586476925286766559`
- `#define DEG_TO_RAD 0.017453292519943295769236907684886`
- `#define RAD_TO_DEG 57.295779513082320876798154814105`
- `#define min(a, b) ((a)<(b)?(a):(b))`
- `#define max(a, b) ((a)>(b)?(a):(b))`
- `#define abs(x) ((x)>=0?(x):- (x))`
- `#define constrain(amt, low, high) ((amt)<(low)?(low):((amt)>(high)?(high):(amt)))`
- `#define round(x) ((x)>=0?(long)((x)+0.5):(long)((x)-0.5))`
- `#define radians(deg) ((deg)*DEG_TO_RAD)`
- `#define degrees(rad) ((rad)*RAD_TO_DEG)`
- `#define sq(x) ((x)*(x))`
- `#define lowByte(w) ((uint8_t) ((w) & 0xff))`
- `#define highByte(w) ((uint8_t) ((w) >> 8))`
- `#define bitRead(value, bit) (((value) >> (bit)) & 0x01)`
- `#define bitSet(value, bit) ((value) |= (1UL << (bit)))`
- `#define bitClear(value, bit) ((value) &= ~(1UL << (bit)))`
- `#define bitWrite(value, bit, bitvalue) (bitvalue ? bitSet(value, bit) : bitClear(value, bit))`
- `#define bit(b) (1UL << (b))`

Typedefs

- typedef unsigned int [word](#)
- typedef uint8_t [boolean](#)
- typedef uint8_t [byte](#)

Functions

- unsigned long [millis](#) (void)
- unsigned long [micros](#) (void)
- void [delay](#) (unsigned long)
- void [delayMicroseconds](#) (unsigned int us)

6.3.1 Macro Definition Documentation

6.3.1.1 `#define abs(x) ((x)>=0?(x):-x)`

6.3.1.2 `#define bit(b) (1UL << (b))`

6.3.1.3 `#define bitClear(value, bit) ((value) &= ~(1UL << (bit)))`

6.3.1.4 `#define bitRead(value, bit) (((value) >> (bit)) & 0x01)`

6.3.1.5 `#define bitSet(value, bit) ((value) |= (1UL << (bit)))`

6.3.1.6 `#define bitWrite(value, bit, bitvalue) (bitvalue ? bitSet(value, bit) : bitClear(value, bit))`

6.3.1.7 `#define constrain(amt, low, high) ((amt)<(low)?(low):((amt)>(high)?(high):(amt)))`

6.3.1.8 `#define DEG_TO_RAD 0.017453292519943295769236907684886`

6.3.1.9 `#define degrees(rad) ((rad)*RAD_TO_DEG)`

6.3.1.10 `#define false 0x0`

6.3.1.11 `#define HALF_PI 1.5707963267948966192313216916398`

6.3.1.12 `#define highByte(w) ((uint8_t) ((w) >> 8))`

6.3.1.13 `#define lowByte(w) ((uint8_t) ((w) & 0xff))`

6.3.1.14 `#define max(a, b) ((a)>(b)?(a):(b))`

6.3.1.15 `#define min(a, b) ((a)<(b)?(a):(b))`

6.3.1.16 `#define PI 3.1415926535897932384626433832795`

6.3.1.17 `#define RAD_TO_DEG 57.295779513082320876798154814105`

6.3.1.18 `#define radians(deg) ((deg)*DEG_TO_RAD)`

6.3.1.19 `#define round(x) ((x)>=0?(long)((x)+0.5):(long)((x)-0.5))`

6.3.1.20 `#define sq(x) ((x)*(x))`

6.3.1.21 `#define true 0x1`

This is a port of the [Arduino.h](#) file from the Arduino distribution. It ports some of the basic functionalities of the Arduino, making them available on non-Arduino AVR as well as computer platforms.

6.3.1.22 `#define TWO_PI 6.283185307179586476925286766559`

6.3.2 Typedef Documentation

6.3.2.1 `typedef uint8_t boolean`

6.3.2.2 `typedef uint8_t byte`

6.3.2.3 `typedef unsigned int word`

6.3.3 Function Documentation

6.3.3.1 `void delay (unsigned long)`

6.3.3.2 `void delayMicroseconds (unsigned int us)`

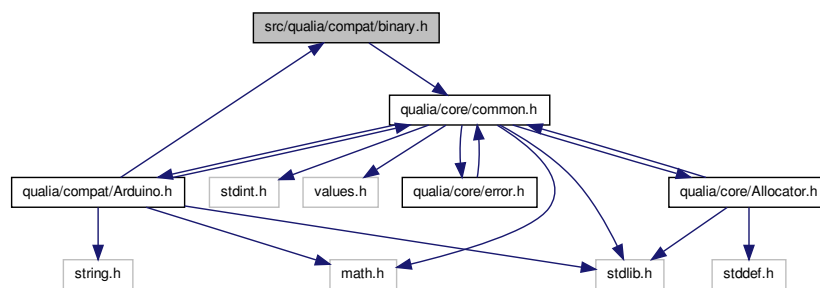
6.3.3.3 `unsigned long micros (void)`

6.3.3.4 `unsigned long millis (void)`

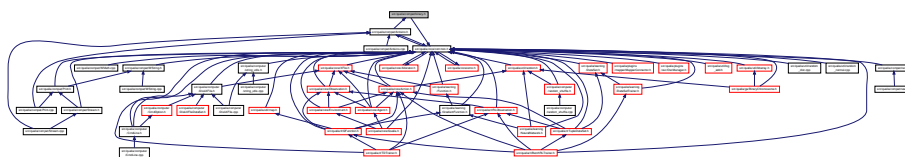
6.4 `src/qualia/compat/binary.h` File Reference

```
#include <qualia/core/common.h>
```

Include dependency graph for `binary.h`:



This graph shows which files directly or indirectly include this file:



Macros

- #define [B0](#) 0
- #define [B00](#) 0
- #define [B000](#) 0
- #define [B0000](#) 0
- #define [B00000](#) 0
- #define [B000000](#) 0
- #define [B0000000](#) 0
- #define [B00000000](#) 0
- #define [B1](#) 1
- #define [B01](#) 1
- #define [B001](#) 1
- #define [B0001](#) 1
- #define [B00001](#) 1
- #define [B000001](#) 1
- #define [B0000001](#) 1
- #define [B00000001](#) 1
- #define [B10](#) 2
- #define [B010](#) 2
- #define [B0010](#) 2
- #define [B00010](#) 2
- #define [B000010](#) 2
- #define [B0000010](#) 2
- #define [B00000010](#) 2
- #define [B11](#) 3
- #define [B011](#) 3
- #define [B0011](#) 3
- #define [B00011](#) 3
- #define [B000011](#) 3
- #define [B0000011](#) 3
- #define [B100](#) 4
- #define [B0100](#) 4
- #define [B00100](#) 4
- #define [B000100](#) 4
- #define [B0000100](#) 4
- #define [B00000100](#) 4
- #define [B101](#) 5
- #define [B0101](#) 5
- #define [B00101](#) 5
- #define [B000101](#) 5
- #define [B0000101](#) 5
- #define [B110](#) 6
- #define [B0110](#) 6
- #define [B00110](#) 6
- #define [B000110](#) 6
- #define [B0000110](#) 6
- #define [B111](#) 7
- #define [B0111](#) 7
- #define [B00111](#) 7
- #define [B000111](#) 7
- #define [B0000111](#) 7

- #define B00000111 7
- #define B1000 8
- #define B01000 8
- #define B001000 8
- #define B0001000 8
- #define B00001000 8
- #define B1001 9
- #define B01001 9
- #define B001001 9
- #define B0001001 9
- #define B00001001 9
- #define B1010 10
- #define B01010 10
- #define B001010 10
- #define B0001010 10
- #define B00001010 10
- #define B1011 11
- #define B01011 11
- #define B001011 11
- #define B0001011 11
- #define B00001011 11
- #define B1100 12
- #define B01100 12
- #define B001100 12
- #define B0001100 12
- #define B00001100 12
- #define B1101 13
- #define B01101 13
- #define B001101 13
- #define B0001101 13
- #define B00001101 13
- #define B1110 14
- #define B01110 14
- #define B001110 14
- #define B0001110 14
- #define B00001110 14
- #define B1111 15
- #define B01111 15
- #define B001111 15
- #define B0001111 15
- #define B00001111 15
- #define B10000 16
- #define B010000 16
- #define B0010000 16
- #define B00010000 16
- #define B10001 17
- #define B010001 17
- #define B0010001 17
- #define B00010001 17
- #define B10010 18
- #define B010010 18
- #define B0010010 18
- #define B00010010 18
- #define B10011 19
- #define B010011 19

- #define [B0010011](#) 19
- #define [B00010011](#) 19
- #define [B10100](#) 20
- #define [B010100](#) 20
- #define [B0010100](#) 20
- #define [B00010100](#) 20
- #define [B10101](#) 21
- #define [B010101](#) 21
- #define [B0010101](#) 21
- #define [B00010101](#) 21
- #define [B10110](#) 22
- #define [B010110](#) 22
- #define [B0010110](#) 22
- #define [B00010110](#) 22
- #define [B10111](#) 23
- #define [B010111](#) 23
- #define [B0010111](#) 23
- #define [B00010111](#) 23
- #define [B11000](#) 24
- #define [B011000](#) 24
- #define [B0011000](#) 24
- #define [B00011000](#) 24
- #define [B11001](#) 25
- #define [B011001](#) 25
- #define [B0011001](#) 25
- #define [B00011001](#) 25
- #define [B11010](#) 26
- #define [B011010](#) 26
- #define [B0011010](#) 26
- #define [B00011010](#) 26
- #define [B11011](#) 27
- #define [B011011](#) 27
- #define [B0011011](#) 27
- #define [B00011011](#) 27
- #define [B11100](#) 28
- #define [B011100](#) 28
- #define [B0011100](#) 28
- #define [B00011100](#) 28
- #define [B11101](#) 29
- #define [B011101](#) 29
- #define [B0011101](#) 29
- #define [B00011101](#) 29
- #define [B11110](#) 30
- #define [B011110](#) 30
- #define [B0011110](#) 30
- #define [B00011110](#) 30
- #define [B11111](#) 31
- #define [B011111](#) 31
- #define [B0011111](#) 31
- #define [B00011111](#) 31
- #define [B100000](#) 32
- #define [B0100000](#) 32
- #define [B00100000](#) 32
- #define [B100001](#) 33
- #define [B0100001](#) 33

- #define B00100001 33
- #define B100010 34
- #define B0100010 34
- #define B00100010 34
- #define B100011 35
- #define B0100011 35
- #define B00100011 35
- #define B100100 36
- #define B0100100 36
- #define B00100100 36
- #define B100101 37
- #define B0100101 37
- #define B00100101 37
- #define B100110 38
- #define B0100110 38
- #define B00100110 38
- #define B100111 39
- #define B0100111 39
- #define B00100111 39
- #define B101000 40
- #define B0101000 40
- #define B00101000 40
- #define B101001 41
- #define B0101001 41
- #define B00101001 41
- #define B101010 42
- #define B0101010 42
- #define B00101010 42
- #define B101011 43
- #define B0101011 43
- #define B00101011 43
- #define B101100 44
- #define B0101100 44
- #define B00101100 44
- #define B101101 45
- #define B0101101 45
- #define B00101101 45
- #define B101110 46
- #define B0101110 46
- #define B00101110 46
- #define B101111 47
- #define B0101111 47
- #define B00101111 47
- #define B110000 48
- #define B0110000 48
- #define B00110000 48
- #define B110001 49
- #define B0110001 49
- #define B00110001 49
- #define B110010 50
- #define B0110010 50
- #define B00110010 50
- #define B110011 51
- #define B0110011 51
- #define B00110011 51

- #define [B110100](#) 52
- #define [B0110100](#) 52
- #define [B00110100](#) 52
- #define [B110101](#) 53
- #define [B0110101](#) 53
- #define [B00110101](#) 53
- #define [B110110](#) 54
- #define [B0110110](#) 54
- #define [B00110110](#) 54
- #define [B110111](#) 55
- #define [B0110111](#) 55
- #define [B00110111](#) 55
- #define [B111000](#) 56
- #define [B0111000](#) 56
- #define [B00111000](#) 56
- #define [B111001](#) 57
- #define [B0111001](#) 57
- #define [B00111001](#) 57
- #define [B111010](#) 58
- #define [B0111010](#) 58
- #define [B00111010](#) 58
- #define [B111011](#) 59
- #define [B0111011](#) 59
- #define [B00111011](#) 59
- #define [B111100](#) 60
- #define [B0111100](#) 60
- #define [B00111100](#) 60
- #define [B111101](#) 61
- #define [B0111101](#) 61
- #define [B00111101](#) 61
- #define [B111110](#) 62
- #define [B0111110](#) 62
- #define [B00111110](#) 62
- #define [B111111](#) 63
- #define [B0111111](#) 63
- #define [B00111111](#) 63
- #define [B1000000](#) 64
- #define [B01000000](#) 64
- #define [B1000001](#) 65
- #define [B01000001](#) 65
- #define [B1000010](#) 66
- #define [B01000010](#) 66
- #define [B1000011](#) 67
- #define [B01000011](#) 67
- #define [B1000100](#) 68
- #define [B01000100](#) 68
- #define [B1000101](#) 69
- #define [B01000101](#) 69
- #define [B1000110](#) 70
- #define [B01000110](#) 70
- #define [B1000111](#) 71
- #define [B01000111](#) 71
- #define [B1001000](#) 72
- #define [B01001000](#) 72
- #define [B1001001](#) 73

- #define B01001001 73
- #define B1001010 74
- #define B01001010 74
- #define B1001011 75
- #define B01001011 75
- #define B1001100 76
- #define B01001100 76
- #define B1001101 77
- #define B01001101 77
- #define B1001110 78
- #define B01001110 78
- #define B1001111 79
- #define B01001111 79
- #define B1010000 80
- #define B01010000 80
- #define B1010001 81
- #define B01010001 81
- #define B1010010 82
- #define B01010010 82
- #define B1010011 83
- #define B01010011 83
- #define B1010100 84
- #define B01010100 84
- #define B1010101 85
- #define B01010101 85
- #define B1010110 86
- #define B01010110 86
- #define B1010111 87
- #define B01010111 87
- #define B1011000 88
- #define B01011000 88
- #define B1011001 89
- #define B01011001 89
- #define B1011010 90
- #define B01011010 90
- #define B1011011 91
- #define B01011011 91
- #define B1011100 92
- #define B01011100 92
- #define B1011101 93
- #define B01011101 93
- #define B1011110 94
- #define B01011110 94
- #define B1011111 95
- #define B01011111 95
- #define B1100000 96
- #define B01100000 96
- #define B1100001 97
- #define B01100001 97
- #define B1100010 98
- #define B01100010 98
- #define B1100011 99
- #define B01100011 99
- #define B1100100 100
- #define B01100100 100

- #define [B1100101](#) 101
- #define [B01100101](#) 101
- #define [B1100110](#) 102
- #define [B01100110](#) 102
- #define [B1100111](#) 103
- #define [B01100111](#) 103
- #define [B1101000](#) 104
- #define [B01101000](#) 104
- #define [B1101001](#) 105
- #define [B01101001](#) 105
- #define [B1101010](#) 106
- #define [B01101010](#) 106
- #define [B1101011](#) 107
- #define [B01101011](#) 107
- #define [B1101100](#) 108
- #define [B01101100](#) 108
- #define [B1101101](#) 109
- #define [B01101101](#) 109
- #define [B1101110](#) 110
- #define [B01101110](#) 110
- #define [B1101111](#) 111
- #define [B01101111](#) 111
- #define [B1110000](#) 112
- #define [B01110000](#) 112
- #define [B1110001](#) 113
- #define [B01110001](#) 113
- #define [B1110010](#) 114
- #define [B01110010](#) 114
- #define [B1110011](#) 115
- #define [B01110011](#) 115
- #define [B1110100](#) 116
- #define [B01110100](#) 116
- #define [B1110101](#) 117
- #define [B01110101](#) 117
- #define [B1110110](#) 118
- #define [B01110110](#) 118
- #define [B1110111](#) 119
- #define [B01110111](#) 119
- #define [B1111000](#) 120
- #define [B01111000](#) 120
- #define [B1111001](#) 121
- #define [B01111001](#) 121
- #define [B1111010](#) 122
- #define [B01111010](#) 122
- #define [B1111011](#) 123
- #define [B01111011](#) 123
- #define [B1111100](#) 124
- #define [B01111100](#) 124
- #define [B1111101](#) 125
- #define [B01111101](#) 125
- #define [B1111110](#) 126
- #define [B01111110](#) 126
- #define [B1111111](#) 127
- #define [B01111111](#) 127
- #define [B10000000](#) 128

- #define B10000001 129
- #define B10000010 130
- #define B10000011 131
- #define B10000100 132
- #define B10000101 133
- #define B10000110 134
- #define B10000111 135
- #define B10001000 136
- #define B10001001 137
- #define B10001010 138
- #define B10001011 139
- #define B10001100 140
- #define B10001101 141
- #define B10001110 142
- #define B10001111 143
- #define B10010000 144
- #define B10010001 145
- #define B10010010 146
- #define B10010011 147
- #define B10010100 148
- #define B10010101 149
- #define B10010110 150
- #define B10010111 151
- #define B10011000 152
- #define B10011001 153
- #define B10011010 154
- #define B10011011 155
- #define B10011100 156
- #define B10011101 157
- #define B10011110 158
- #define B10011111 159
- #define B10100000 160
- #define B10100001 161
- #define B10100010 162
- #define B10100011 163
- #define B10100100 164
- #define B10100101 165
- #define B10100110 166
- #define B10100111 167
- #define B10101000 168
- #define B10101001 169
- #define B10101010 170
- #define B10101011 171
- #define B10101100 172
- #define B10101101 173
- #define B10101110 174
- #define B10101111 175
- #define B10110000 176
- #define B10110001 177
- #define B10110010 178
- #define B10110011 179
- #define B10110100 180
- #define B10110101 181
- #define B10110110 182
- #define B10110111 183

- #define [B10111000](#) 184
- #define [B10111001](#) 185
- #define [B10111010](#) 186
- #define [B10111011](#) 187
- #define [B10111100](#) 188
- #define [B10111101](#) 189
- #define [B10111110](#) 190
- #define [B10111111](#) 191
- #define [B11000000](#) 192
- #define [B11000001](#) 193
- #define [B11000010](#) 194
- #define [B11000011](#) 195
- #define [B11000100](#) 196
- #define [B11000101](#) 197
- #define [B11000110](#) 198
- #define [B11000111](#) 199
- #define [B11001000](#) 200
- #define [B11001001](#) 201
- #define [B11001010](#) 202
- #define [B11001011](#) 203
- #define [B11001100](#) 204
- #define [B11001101](#) 205
- #define [B11001110](#) 206
- #define [B11001111](#) 207
- #define [B11010000](#) 208
- #define [B11010001](#) 209
- #define [B11010010](#) 210
- #define [B11010011](#) 211
- #define [B11010100](#) 212
- #define [B11010101](#) 213
- #define [B11010110](#) 214
- #define [B11010111](#) 215
- #define [B11011000](#) 216
- #define [B11011001](#) 217
- #define [B11011010](#) 218
- #define [B11011011](#) 219
- #define [B11011100](#) 220
- #define [B11011101](#) 221
- #define [B11011110](#) 222
- #define [B11011111](#) 223
- #define [B11100000](#) 224
- #define [B11100001](#) 225
- #define [B11100010](#) 226
- #define [B11100011](#) 227
- #define [B11100100](#) 228
- #define [B11100101](#) 229
- #define [B11100110](#) 230
- #define [B11100111](#) 231
- #define [B11101000](#) 232
- #define [B11101001](#) 233
- #define [B11101010](#) 234
- #define [B11101011](#) 235
- #define [B11101100](#) 236
- #define [B11101101](#) 237
- #define [B11101110](#) 238

- `#define B11101111` 239
- `#define B11110000` 240
- `#define B11110001` 241
- `#define B11110010` 242
- `#define B11110011` 243
- `#define B11110100` 244
- `#define B11110101` 245
- `#define B11110110` 246
- `#define B11110111` 247
- `#define B11111000` 248
- `#define B11111001` 249
- `#define B11111010` 250
- `#define B11111011` 251
- `#define B11111100` 252
- `#define B11111101` 253
- `#define B11111110` 254
- `#define B11111111` 255

6.4.1 Macro Definition Documentation

6.4.1.1 `#define B0 0`

6.4.1.2 `#define B00 0`

6.4.1.3 `#define B000 0`

6.4.1.4 `#define B0000 0`

6.4.1.5 `#define B00000 0`

6.4.1.6 `#define B000000 0`

6.4.1.7 `#define B0000000 0`

6.4.1.8 `#define B00000000 0`

6.4.1.9 `#define B00000001 1`

6.4.1.10 `#define B0000001 1`

6.4.1.11 `#define B00000010 2`

6.4.1.12 `#define B00000011 3`

6.4.1.13 `#define B000001 1`

6.4.1.14 `#define B0000010 2`

6.4.1.15 `#define B00000100 4`

6.4.1.16 `#define B00000101 5`

6.4.1.17 `#define B0000011 3`

6.4.1.18 `#define B00000110 6`

6.4.1.19 `#define B00000111 7`

6.4.1.20 `#define B00001 1`

6.4.1.21 `#define B000010 2`

6.4.1.22 `#define B0000100 4`

6.4.1.23 `#define B00001000 8`

6.4.1.24 `#define B00001001 9`

6.4.1.25 `#define B0000101 5`

6.4.1.26 `#define B00001010 10`

6.4.1.27 `#define B00001011 11`

6.4.1.28 `#define B000011 3`

6.4.1.29 `#define B0000110 6`

6.4.1.30 `#define B00001100 12`

6.4.1.31 `#define B00001101 13`

6.4.1.32 `#define B0000111 7`

6.4.1.33 `#define B00001110 14`

6.4.1.34 `#define B00001111 15`

6.4.1.35 `#define B0001 1`

6.4.1.36 `#define B00010 2`

6.4.1.37 `#define B000100 4`

6.4.1.38 `#define B0001000 8`

6.4.1.39 `#define B00010000 16`

6.4.1.40 `#define B00010001 17`

6.4.1.41 `#define B0001001 9`

6.4.1.42 `#define B00010010 18`

6.4.1.43 `#define B00010011 19`

6.4.1.44 `#define B000101 5`

6.4.1.45 `#define B0001010 10`

6.4.1.46 `#define B00010100 20`

6.4.1.47 `#define B00010101 21`

6.4.1.48 `#define B0001011 11`

6.4.1.49 `#define B00010110 22`

6.4.1.50 `#define B00010111 23`

6.4.1.51 `#define B00011 3`

6.4.1.52 `#define B000110 6`

6.4.1.53 `#define B0001100 12`

6.4.1.54 `#define B00011000 24`

6.4.1.55 `#define B00011001 25`

6.4.1.56 `#define B0001101 13`

6.4.1.57 `#define B00011010 26`

6.4.1.58 `#define B00011011 27`

6.4.1.59 `#define B000111 7`

6.4.1.60 `#define B0001110 14`

6.4.1.61 `#define B00011100 28`

6.4.1.62 `#define B00011101 29`

6.4.1.63 `#define B0001111 15`

6.4.1.64 `#define B00011110 30`

6.4.1.65 `#define B00011111 31`

6.4.1.66 `#define B001 1`

6.4.1.67 `#define B0010 2`

6.4.1.68 `#define B00100 4`

6.4.1.69 `#define B001000 8`

6.4.1.70 `#define B0010000 16`

6.4.1.71 `#define B00100000 32`

6.4.1.72 `#define B00100001 33`

6.4.1.73 `#define B0010001 17`

6.4.1.74 `#define B00100010 34`

6.4.1.75 `#define B00100011 35`

6.4.1.76 `#define B001001 9`

6.4.1.77 `#define B0010010 18`

6.4.1.78 `#define B00100100 36`

6.4.1.79 `#define B00100101 37`

6.4.1.80 `#define B0010011 19`

6.4.1.81 `#define B00100110 38`

6.4.1.82 `#define B00100111 39`

6.4.1.83 `#define B00101 5`

6.4.1.84 `#define B001010 10`

6.4.1.85 `#define B0010100 20`

6.4.1.86 `#define B00101000 40`

6.4.1.87 `#define B00101001 41`

6.4.1.88 `#define B0010101 21`

6.4.1.89 `#define B00101010 42`

6.4.1.90 `#define B00101011 43`

6.4.1.91 `#define B001011 11`

6.4.1.92 `#define B0010110 22`

6.4.1.93 `#define B00101100 44`

6.4.1.94 `#define B00101101 45`

6.4.1.95 `#define B0010111 23`

6.4.1.96 `#define B00101110 46`

6.4.1.97 `#define B00101111 47`

6.4.1.98 `#define B0011 3`

6.4.1.99 `#define B00110 6`

6.4.1.100 `#define B001100 12`

6.4.1.101 `#define B0011000 24`

6.4.1.102 `#define B00110000 48`

6.4.1.103 `#define B00110001 49`

6.4.1.104 `#define B0011001 25`

6.4.1.105 `#define B00110010 50`

6.4.1.106 `#define B00110011 51`

6.4.1.107 `#define B001101 13`

6.4.1.108 `#define B0011010 26`

6.4.1.109 `#define B00110100 52`

6.4.1.110 `#define B00110101 53`

6.4.1.111 `#define B0011011 27`

6.4.1.112 `#define B00110110 54`

6.4.1.113 `#define B00110111 55`

6.4.1.114 `#define B00111 7`

6.4.1.115 `#define B001110 14`

6.4.1.116 `#define B0011100 28`

6.4.1.117 `#define B00111000 56`

6.4.1.118 `#define B00111001 57`

6.4.1.119 `#define B0011101 29`

6.4.1.120 `#define B00111010 58`

6.4.1.121 `#define B00111011 59`

6.4.1.122 `#define B001111 15`

6.4.1.123 `#define B0011110 30`

6.4.1.124 `#define B00111100 60`

6.4.1.125 `#define B00111101 61`

6.4.1.126 `#define B0011111 31`

6.4.1.127 `#define B00111110 62`

6.4.1.128 `#define B00111111 63`

6.4.1.129 `#define B01 1`

6.4.1.130 `#define B010 2`

6.4.1.131 `#define B0100 4`

6.4.1.132 `#define B01000 8`

6.4.1.133 `#define B010000 16`

6.4.1.134 `#define B0100000 32`

6.4.1.135 `#define B01000000 64`

6.4.1.136 `#define B01000001 65`

6.4.1.137 `#define B0100001 33`

6.4.1.138 `#define B01000010 66`

6.4.1.139 `#define B01000011 67`

6.4.1.140 `#define B010001 17`

6.4.1.141 `#define B0100010 34`

6.4.1.142 `#define B01000100 68`

6.4.1.143 `#define B01000101 69`

6.4.1.144 `#define B0100011 35`

6.4.1.145 `#define B01000110 70`

6.4.1.146 `#define B01000111 71`

6.4.1.147 `#define B01001 9`

6.4.1.148 `#define B010010 18`

6.4.1.149 `#define B0100100 36`

6.4.1.150 `#define B01001000 72`

6.4.1.151 `#define B01001001 73`

6.4.1.152 `#define B0100101 37`

6.4.1.153 `#define B01001010 74`

6.4.1.154 `#define B01001011 75`

6.4.1.155 `#define B010011 19`

6.4.1.156 `#define B0100110 38`

6.4.1.157 `#define B01001100 76`

6.4.1.158 `#define B01001101 77`

6.4.1.159 `#define B0100111 39`

6.4.1.160 `#define B01001110 78`

6.4.1.161 `#define B01001111 79`

6.4.1.162 `#define B0101 5`

6.4.1.163 `#define B01010 10`

6.4.1.164 `#define B010100 20`

6.4.1.165 `#define B0101000 40`

6.4.1.166 `#define B01010000 80`

6.4.1.167 `#define B01010001 81`

6.4.1.168 `#define B0101001 41`

6.4.1.169 `#define B01010010 82`

6.4.1.170 `#define B01010011 83`

6.4.1.171 `#define B010101 21`

6.4.1.172 `#define B0101010 42`

6.4.1.173 `#define B01010100 84`

6.4.1.174 `#define B01010101 85`

6.4.1.175 `#define B0101011 43`

6.4.1.176 `#define B01010110 86`

6.4.1.177 `#define B01010111 87`

6.4.1.178 `#define B01011 11`

6.4.1.179 `#define B010110 22`

6.4.1.180 `#define B0101100 44`

6.4.1.181 `#define B01011000 88`

6.4.1.182 `#define B01011001 89`

6.4.1.183 `#define B0101101 45`

6.4.1.184 `#define B01011010 90`

6.4.1.185 `#define B01011011 91`

6.4.1.186 `#define B010111 23`

6.4.1.187 `#define B0101110 46`

6.4.1.188 `#define B01011100 92`

6.4.1.189 `#define B01011101 93`

6.4.1.190 `#define B0101111 47`

6.4.1.191 `#define B01011110 94`

6.4.1.192 `#define B01011111 95`

6.4.1.193 `#define B011 3`

6.4.1.194 `#define B0110 6`

6.4.1.195 `#define B01100 12`

6.4.1.196 `#define B011000 24`

6.4.1.197 `#define B0110000 48`

6.4.1.198 `#define B01100000 96`

6.4.1.199 `#define B01100001 97`

6.4.1.200 `#define B0110001 49`

6.4.1.201 `#define B01100010 98`

6.4.1.202 `#define B01100011 99`

6.4.1.203 `#define B011001 25`

6.4.1.204 `#define B0110010 50`

6.4.1.205 `#define B01100100 100`

6.4.1.206 `#define B01100101 101`

6.4.1.207 `#define B0110011 51`

6.4.1.208 `#define B01100110 102`

6.4.1.209 `#define B01100111 103`

6.4.1.210 `#define B01101 13`

6.4.1.211 `#define B011010 26`

6.4.1.212 `#define B0110100 52`

6.4.1.213 `#define B01101000 104`

6.4.1.214 `#define B01101001 105`

6.4.1.215 `#define B0110101 53`

6.4.1.216 `#define B01101010 106`

6.4.1.217 `#define B01101011 107`

6.4.1.218 `#define B011011 27`

6.4.1.219 `#define B0110110 54`

6.4.1.220 `#define B01101100 108`

6.4.1.221 `#define B01101101 109`

6.4.1.222 `#define B0110111 55`

6.4.1.223 `#define B01101110 110`

6.4.1.224 `#define B01101111 111`

6.4.1.225 `#define B0111 7`

6.4.1.226 `#define B01110 14`

6.4.1.227 `#define B011100 28`

6.4.1.228 `#define B0111000 56`

6.4.1.229 `#define B01110000 112`

6.4.1.230 `#define B01110001 113`

6.4.1.231 `#define B0111001 57`

6.4.1.232 `#define B01110010 114`

6.4.1.233 `#define B01110011 115`

6.4.1.234 `#define B011101 29`

6.4.1.235 `#define B0111010 58`

6.4.1.236 `#define B01110100 116`

6.4.1.237 `#define B01110101 117`

6.4.1.238 `#define B0111011 59`

6.4.1.239 `#define B01110110 118`

6.4.1.240 `#define B01110111 119`

6.4.1.241 `#define B01111 15`

6.4.1.242 `#define B011110 30`

6.4.1.243 `#define B0111100 60`

6.4.1.244 `#define B01111000 120`

6.4.1.245 `#define B01111001 121`

6.4.1.246 `#define B0111101 61`

6.4.1.247 `#define B01111010 122`

6.4.1.248 `#define B01111011 123`

6.4.1.249 `#define B011111 31`

6.4.1.250 `#define B0111110 62`

6.4.1.251 `#define B01111100 124`

6.4.1.252 `#define B01111101 125`

6.4.1.253 `#define B0111111 63`

6.4.1.254 `#define B01111110 126`

6.4.1.255 `#define B01111111 127`

6.4.1.256 `#define B1 1`

6.4.1.257 `#define B10 2`

6.4.1.258 `#define B100 4`

6.4.1.259 `#define B1000 8`

6.4.1.260 `#define B10000 16`

6.4.1.261 `#define B100000 32`

6.4.1.262 `#define B1000000 64`

6.4.1.263 `#define B10000000 128`

6.4.1.264 `#define B10000001 129`

6.4.1.265 `#define B1000001 65`

6.4.1.266 `#define B10000010 130`

6.4.1.267 `#define B10000011 131`

6.4.1.268 `#define B100001 33`

6.4.1.269 `#define B1000010 66`

6.4.1.270 `#define B10000100 132`

6.4.1.271 `#define B10000101 133`

6.4.1.272 `#define B1000011 67`

6.4.1.273 `#define B10000110 134`

6.4.1.274 `#define B10000111 135`

6.4.1.275 `#define B10001 17`

6.4.1.276 `#define B100010 34`

6.4.1.277 `#define B1000100 68`

6.4.1.278 `#define B10001000 136`

6.4.1.279 `#define B10001001 137`

6.4.1.280 `#define B1000101 69`

6.4.1.281 `#define B10001010 138`

6.4.1.282 `#define B10001011 139`

6.4.1.283 `#define B100011 35`

6.4.1.284 `#define B1000110 70`

6.4.1.285 `#define B10001100 140`

6.4.1.286 `#define B10001101 141`

6.4.1.287 `#define B1000111 71`

6.4.1.288 `#define B10001110 142`

6.4.1.289 `#define B10001111 143`

6.4.1.290 `#define B1001 9`

6.4.1.291 `#define B10010 18`

6.4.1.292 `#define B100100 36`

6.4.1.293 `#define B1001000 72`

6.4.1.294 `#define B10010000 144`

6.4.1.295 `#define B10010001 145`

6.4.1.296 `#define B1001001 73`

6.4.1.297 `#define B10010010 146`

6.4.1.298 `#define B10010011 147`

6.4.1.299 `#define B100101 37`

6.4.1.300 `#define B1001010 74`

6.4.1.301 `#define B10010100 148`

6.4.1.302 `#define B10010101 149`

6.4.1.303 `#define B1001011 75`

6.4.1.304 `#define B10010110 150`

6.4.1.305 `#define B10010111 151`

6.4.1.306 `#define B10011 19`

6.4.1.307 `#define B100110 38`

6.4.1.308 `#define B1001100 76`

6.4.1.309 `#define B10011000 152`

6.4.1.310 `#define B10011001 153`

6.4.1.311 `#define B1001101 77`

6.4.1.312 `#define B10011010 154`

6.4.1.313 `#define B10011011 155`

6.4.1.314 `#define B100111 39`

6.4.1.315 `#define B1001110 78`

6.4.1.316 `#define B10011100 156`

6.4.1.317 `#define B10011101 157`

6.4.1.318 `#define B1001111 79`

6.4.1.319 `#define B10011110 158`

6.4.1.320 `#define B10011111 159`

6.4.1.321 `#define B101 5`

6.4.1.322 `#define B1010 10`

6.4.1.323 `#define B10100 20`

6.4.1.324 `#define B101000 40`

6.4.1.325 `#define B1010000 80`

6.4.1.326 `#define B10100000 160`

6.4.1.327 `#define B10100001 161`

6.4.1.328 `#define B1010001 81`

6.4.1.329 `#define B10100010 162`

6.4.1.330 `#define B10100011 163`

6.4.1.331 `#define B101001 41`

6.4.1.332 `#define B1010010 82`

6.4.1.333 `#define B10100100 164`

6.4.1.334 `#define B10100101 165`

6.4.1.335 `#define B1010011 83`

6.4.1.336 `#define B10100110 166`

6.4.1.337 `#define B10100111 167`

6.4.1.338 `#define B10101 21`

6.4.1.339 `#define B101010 42`

6.4.1.340 `#define B1010100 84`

6.4.1.341 `#define B10101000 168`

6.4.1.342 `#define B10101001 169`

6.4.1.343 `#define B1010101 85`

6.4.1.344 `#define B10101010 170`

6.4.1.345 `#define B10101011 171`

6.4.1.346 `#define B101011 43`

6.4.1.347 `#define B1010110 86`

6.4.1.348 `#define B10101100 172`

6.4.1.349 `#define B10101101 173`

6.4.1.350 `#define B1010111 87`

6.4.1.351 `#define B10101110 174`

6.4.1.352 `#define B10101111 175`

6.4.1.353 `#define B1011 11`

6.4.1.354 `#define B10110 22`

6.4.1.355 `#define B101100 44`

6.4.1.356 `#define B1011000 88`

6.4.1.357 `#define B10110000 176`

6.4.1.358 `#define B10110001 177`

6.4.1.359 `#define B1011001 89`

6.4.1.360 `#define B10110010 178`

6.4.1.361 `#define B10110011 179`

6.4.1.362 `#define B101101 45`

6.4.1.363 `#define B1011010 90`

6.4.1.364 `#define B10110100 180`

6.4.1.365 `#define B10110101 181`

6.4.1.366 `#define B1011011 91`

6.4.1.367 `#define B10110110 182`

6.4.1.368 `#define B10110111 183`

6.4.1.369 `#define B10111 23`

6.4.1.370 `#define B101110 46`

6.4.1.371 `#define B1011100 92`

6.4.1.372 `#define B10111000 184`

6.4.1.373 `#define B10111001 185`

6.4.1.374 `#define B1011101 93`

6.4.1.375 `#define B10111010 186`

6.4.1.376 `#define B10111011 187`

6.4.1.377 `#define B101111 47`

6.4.1.378 `#define B1011110 94`

6.4.1.379 `#define B10111100 188`

6.4.1.380 `#define B10111101 189`

6.4.1.381 `#define B1011111 95`

6.4.1.382 `#define B10111110 190`

6.4.1.383 `#define B10111111 191`

6.4.1.384 `#define B11 3`

6.4.1.385 `#define B110 6`

6.4.1.386 `#define B1100 12`

6.4.1.387 `#define B11000 24`

6.4.1.388 `#define B110000 48`

6.4.1.389 `#define B1100000 96`

6.4.1.390 `#define B11000000 192`

6.4.1.391 `#define B11000001 193`

6.4.1.392 `#define B1100001 97`

6.4.1.393 `#define B11000010 194`

6.4.1.394 `#define B11000011 195`

6.4.1.395 `#define B110001 49`

6.4.1.396 `#define B1100010 98`

6.4.1.397 `#define B11000100 196`

6.4.1.398 `#define B11000101 197`

6.4.1.399 `#define B1100011 99`

6.4.1.400 `#define B11000110 198`

6.4.1.401 `#define B11000111 199`

6.4.1.402 `#define B11001 25`

6.4.1.403 `#define B110010 50`

6.4.1.404 `#define B1100100 100`

6.4.1.405 `#define B11001000 200`

6.4.1.406 `#define B11001001 201`

6.4.1.407 `#define B1100101 101`

6.4.1.408 `#define B11001010 202`

6.4.1.409 `#define B11001011 203`

6.4.1.410 `#define B110011 51`

6.4.1.411 `#define B1100110 102`

6.4.1.412 `#define B11001100 204`

6.4.1.413 `#define B11001101 205`

6.4.1.414 `#define B1100111 103`

6.4.1.415 `#define B11001110 206`

6.4.1.416 `#define B11001111 207`

6.4.1.417 `#define B1101 13`

6.4.1.418 `#define B11010 26`

6.4.1.419 `#define B110100 52`

6.4.1.420 `#define B1101000 104`

6.4.1.421 `#define B11010000 208`

6.4.1.422 `#define B11010001 209`

6.4.1.423 `#define B1101001 105`

6.4.1.424 `#define B11010010 210`

6.4.1.425 `#define B11010011 211`

6.4.1.426 `#define B110101 53`

6.4.1.427 `#define B1101010 106`

6.4.1.428 `#define B11010100 212`

6.4.1.429 `#define B11010101 213`

6.4.1.430 `#define B1101011 107`

6.4.1.431 `#define B11010110 214`

6.4.1.432 `#define B11010111 215`

6.4.1.433 `#define B11011 27`

6.4.1.434 `#define B110110 54`

6.4.1.435 `#define B1101100 108`

6.4.1.436 `#define B11011000 216`

6.4.1.437 `#define B11011001 217`

6.4.1.438 `#define B1101101 109`

6.4.1.439 `#define B11011010 218`

6.4.1.440 `#define B11011011 219`

6.4.1.441 `#define B110111 55`

6.4.1.442 `#define B1101110 110`

6.4.1.443 `#define B11011100 220`

6.4.1.444 `#define B11011101 221`

6.4.1.445 `#define B1101111 111`

6.4.1.446 `#define B11011110 222`

6.4.1.447 `#define B11011111 223`

6.4.1.448 `#define B111 7`

6.4.1.449 `#define B1110 14`

6.4.1.450 `#define B11100 28`

6.4.1.451 `#define B111000 56`

6.4.1.452 `#define B1110000 112`

6.4.1.453 `#define B11100000 224`

6.4.1.454 `#define B11100001 225`

6.4.1.455 `#define B1110001 113`

6.4.1.456 `#define B11100010 226`

6.4.1.457 `#define B11100011 227`

6.4.1.458 `#define B111001 57`

6.4.1.459 `#define B1110010 114`

6.4.1.460 `#define B11100100 228`

6.4.1.461 `#define B11100101 229`

6.4.1.462 `#define B1110011 115`

6.4.1.463 `#define B11100110 230`

6.4.1.464 `#define B11100111 231`

6.4.1.465 `#define B11101 29`

6.4.1.466 `#define B111010 58`

6.4.1.467 `#define B1110100 116`

6.4.1.468 `#define B11101000 232`

6.4.1.469 `#define B11101001 233`

6.4.1.470 `#define B1110101 117`

6.4.1.471 `#define B11101010 234`

6.4.1.472 `#define B11101011 235`

6.4.1.473 `#define B111011 59`

6.4.1.474 `#define B1110110 118`

6.4.1.475 `#define B11101100 236`

6.4.1.476 `#define B11101101 237`

6.4.1.477 `#define B1110111 119`

6.4.1.478 `#define B11101110 238`

6.4.1.479 `#define B11101111 239`

6.4.1.480 `#define B1111 15`

6.4.1.481 `#define B11110 30`

6.4.1.482 `#define B111100 60`

6.4.1.483 `#define B1111000 120`

6.4.1.484 `#define B11110000 240`

6.4.1.485 `#define B11110001 241`

6.4.1.486 `#define B1111001 121`

6.4.1.487 `#define B11110010 242`

6.4.1.488 `#define B11110011 243`

6.4.1.489 `#define B111101 61`

6.4.1.490 `#define B1111010 122`

6.4.1.491 `#define B11110100 244`

6.4.1.492 `#define B11110101 245`

6.4.1.493 `#define B1111011 123`

6.4.1.494 `#define B11110110 246`

6.4.1.495 `#define B11110111 247`

6.4.1.496 `#define B11111 31`

6.4.1.497 `#define B111110 62`

6.4.1.498 `#define B1111100 124`

6.4.1.499 `#define B11111000 248`

6.4.1.500 `#define B11111001 249`

6.4.1.501 `#define B1111101 125`

6.4.1.502 `#define B11111010 250`

6.4.1.503 `#define B11111011 251`

6.4.1.504 `#define B111111 63`

6.4.1.505 `#define B1111110 126`

6.4.1.506 `#define B11111100 252`

6.4.1.507 `#define B11111101 253`

6.4.1.508 `#define B1111111 127`

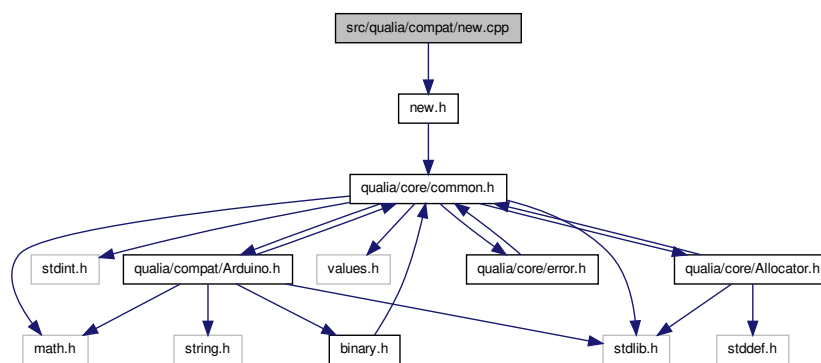
6.4.1.509 `#define B11111110 254`

6.4.1.510 `#define B11111111 255`

6.5 src/qualia/compat/new.cpp File Reference

```
#include "new.h"
```

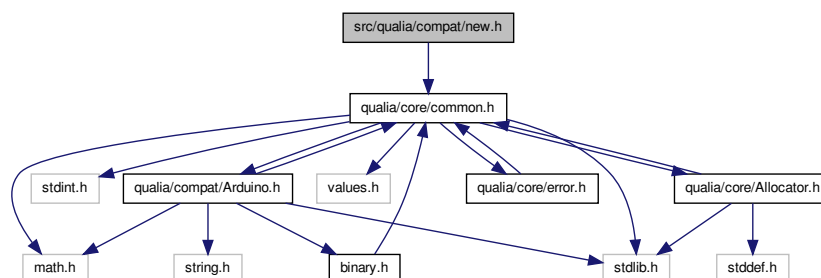
Include dependency graph for new.cpp:



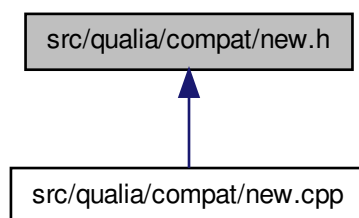
6.6 src/qualia/compat/new.h File Reference

```
#include <qualia/core/common.h>
```

Include dependency graph for new.h:



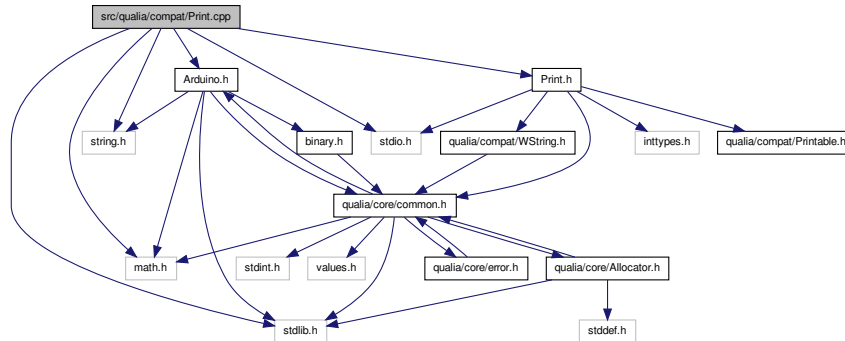
This graph shows which files directly or indirectly include this file:



6.7 src/qualia/compat/Print.cpp File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "Arduino.h"
#include "Print.h"
```

Include dependency graph for Print.cpp:



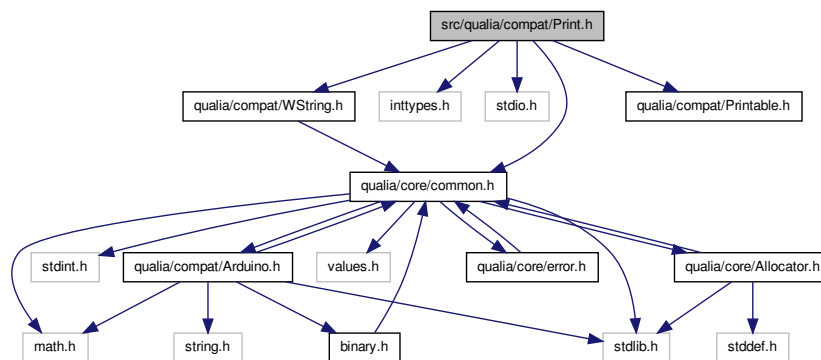
6.8 src/qualia/compat/Print.h File Reference

```

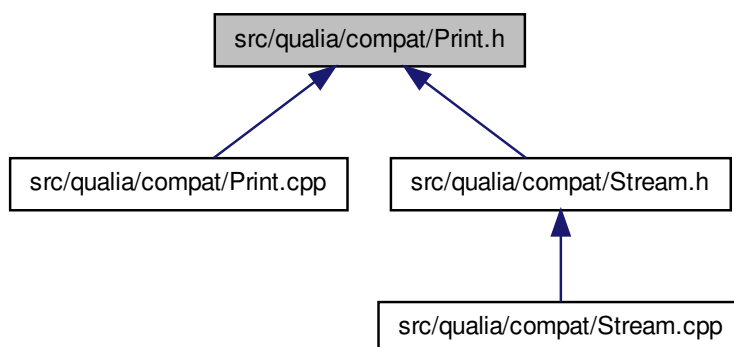
#include <qualia/core/common.h>
#include <inttypes.h>
#include <stdio.h>
#include <qualia/compat/WString.h>
#include <qualia/compat/Printable.h>

```

Include dependency graph for Print.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Print](#)

Macros

- `#define` [DEC](#) 10
- `#define` [HEX](#) 16
- `#define` [OCT](#) 8
- `#define` [BIN](#) 2

6.8.1 Macro Definition Documentation

6.8.1.1 `#define` BIN 2

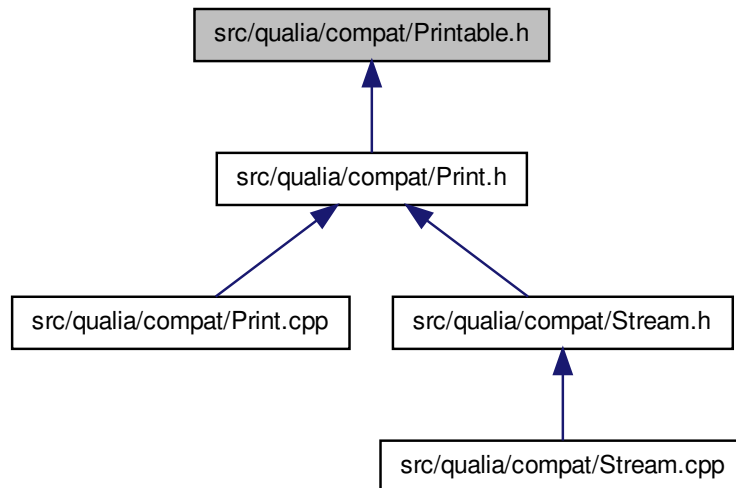
6.8.1.2 `#define` DEC 10

6.8.1.3 `#define` HEX 16

6.8.1.4 `#define` OCT 8

6.9 src/qualia/compat/Printable.h File Reference

This graph shows which files directly or indirectly include this file:



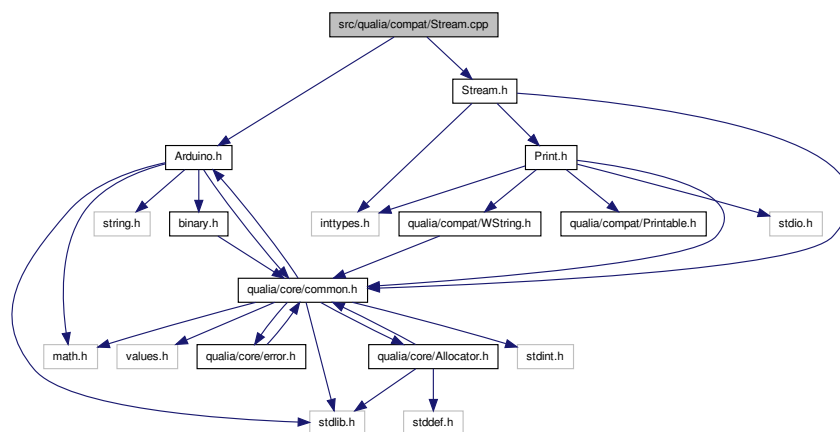
Classes

- class [Printable](#)

6.10 src/qualia/compat/Stream.cpp File Reference

```
#include "Arduino.h"
#include "Stream.h"
```

Include dependency graph for `Stream.cpp`:



Macros

- `#define` [PARSE_TIMEOUT](#) 1000
- `#define` [NO_SKIP_CHAR](#) 1

6.10.1 Macro Definition Documentation

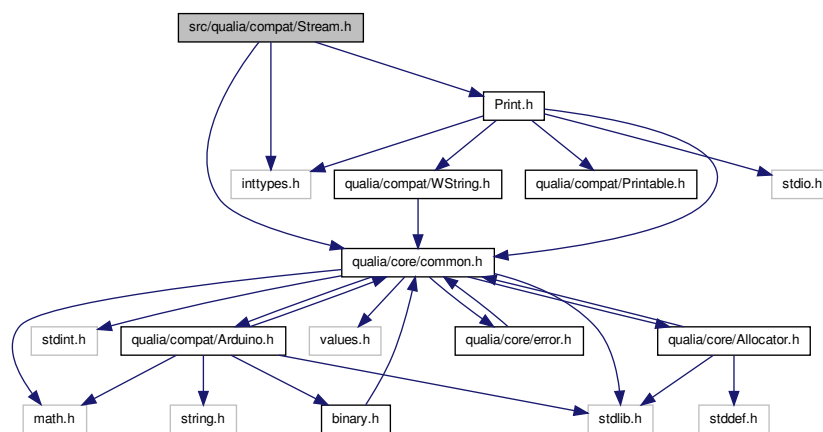
6.10.1.1 `#define` [NO_SKIP_CHAR](#) 1

6.10.1.2 `#define` [PARSE_TIMEOUT](#) 1000

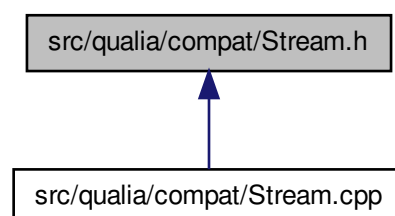
6.11 src/qualia/compat/Stream.h File Reference

```
#include <qualia/core/common.h>
#include <inttypes.h>
#include "Print.h"
```

Include dependency graph for Stream.h:



This graph shows which files directly or indirectly include this file:



Classes

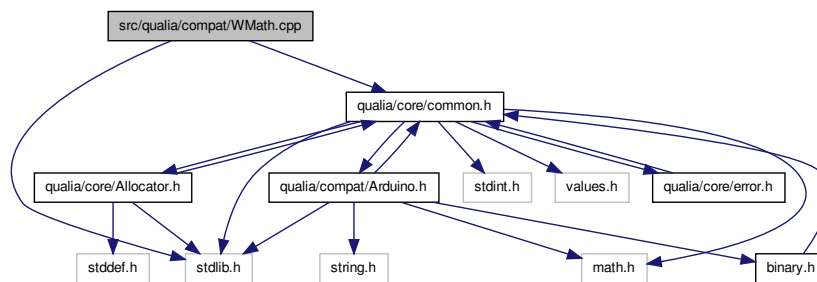
- class [Stream](#)

6.12 src/qualia/compat/WMath.cpp File Reference

```
#include <qualia/core/common.h>
```

```
#include "stdlib.h"
```

Include dependency graph for WMath.cpp:



Functions

- void [randomSeed](#) (unsigned int seed)
- long [random](#) (long howbig)
- long [random](#) (long howsmall, long howbig)
- long [map](#) (long x, long in_min, long in_max, long out_min, long out_max)
- uint16_t [makeWord](#) (unsigned int w)
- uint16_t [makeWord](#) (unsigned char h, unsigned char l)

6.12.1 Function Documentation

6.12.1.1 uint16_t makeWord (unsigned int *w*)

6.12.1.2 uint16_t makeWord (unsigned char *h*, unsigned char *l*)

6.12.1.3 long map (long *x*, long *in_min*, long *in_max*, long *out_min*, long *out_max*)

6.12.1.4 long random (long *howbig*)

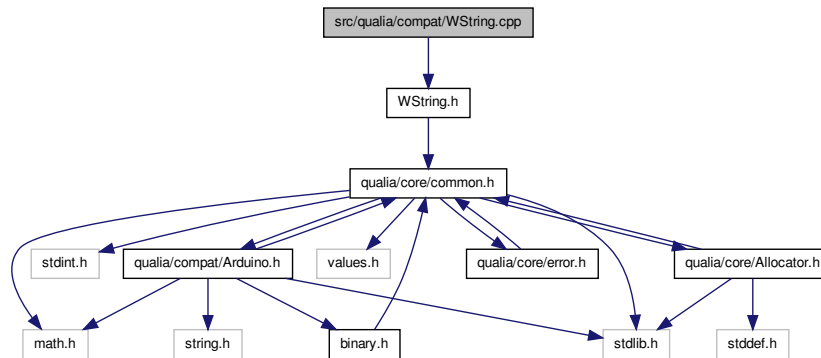
6.12.1.5 long random (long *howsmall*, long *howbig*)

6.12.1.6 void randomSeed (unsigned int *seed*)

6.13 src/qualia/compat/WString.cpp File Reference

```
#include "WString.h"
```

Include dependency graph for WString.cpp:



Functions

- StringSumHelper & [operator+](#) (const StringSumHelper &lhs, const String &rhs)
- StringSumHelper & [operator+](#) (const StringSumHelper &lhs, const char *cstr)
- StringSumHelper & [operator+](#) (const StringSumHelper &lhs, char c)
- StringSumHelper & [operator+](#) (const StringSumHelper &lhs, unsigned char num)
- StringSumHelper & [operator+](#) (const StringSumHelper &lhs, int num)
- StringSumHelper & [operator+](#) (const StringSumHelper &lhs, unsigned int num)
- StringSumHelper & [operator+](#) (const StringSumHelper &lhs, long num)
- StringSumHelper & [operator+](#) (const StringSumHelper &lhs, unsigned long num)

6.13.1 Function Documentation

6.13.1.1 StringSumHelper& operator+ (const StringSumHelper &lhs, const String &rhs)

6.13.1.2 StringSumHelper& operator+ (const StringSumHelper &lhs, const char * cstr)

6.13.1.3 StringSumHelper& operator+ (const StringSumHelper &lhs, char c)

6.13.1.4 StringSumHelper& operator+ (const StringSumHelper &lhs, unsigned char num)

6.13.1.5 StringSumHelper& operator+ (const StringSumHelper &lhs, int num)

6.13.1.6 StringSumHelper& operator+ (const StringSumHelper &lhs, unsigned int num)

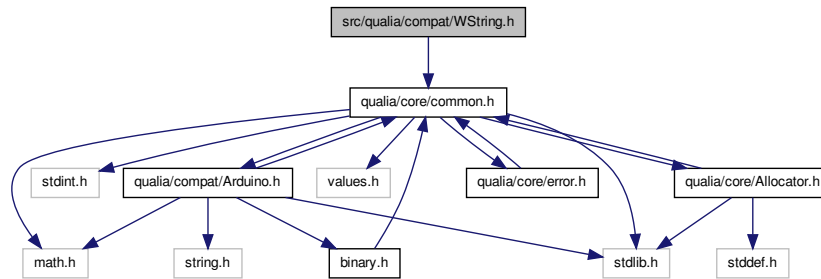
6.13.1.7 StringSumHelper& operator+ (const StringSumHelper &lhs, long num)

6.13.1.8 StringSumHelper& operator+ (const StringSumHelper &lhs, unsigned long num)

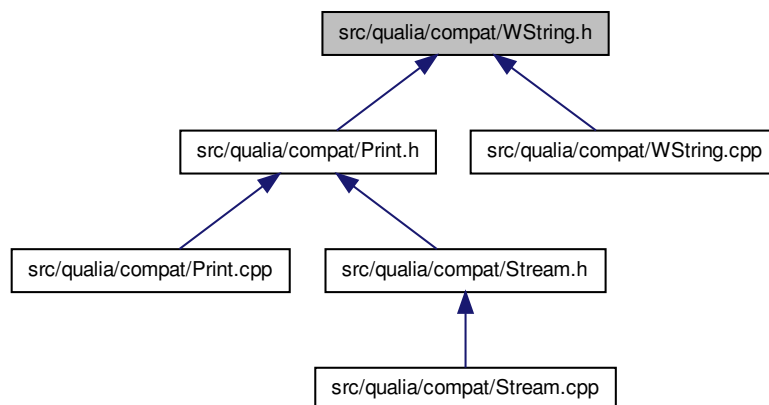
6.14 src/qualia/compat/WString.h File Reference

```
#include <qualia/core/common.h>
```

Include dependency graph for WString.h:



This graph shows which files directly or indirectly include this file:



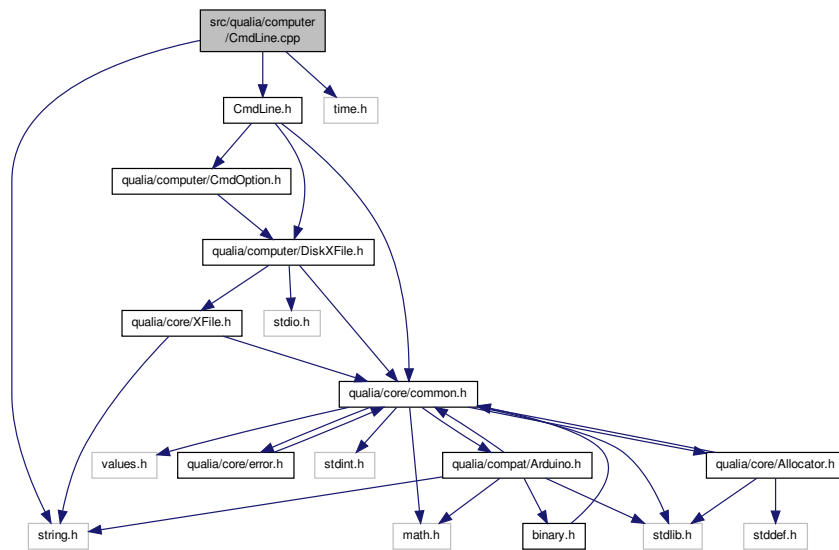
6.15 src/qualia/computer/CmdLine.cpp File Reference

```

#include "CmdLine.h"
#include <time.h>
#include <string.h>

```

Include dependency graph for CmdLine.cpp:



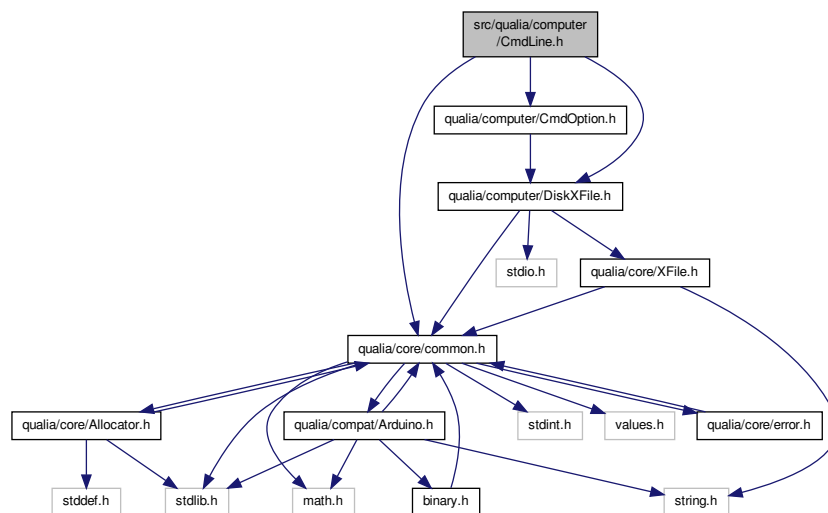
6.16 src/qualia/computer/CmdLine.h File Reference

```

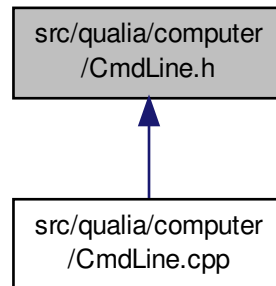
#include <qualia/core/common.h>
#include <qualia/computer/CmdOption.h>
#include <qualia/computer/DiskXFile.h>

```

Include dependency graph for CmdLine.h:



This graph shows which files directly or indirectly include this file:



Classes

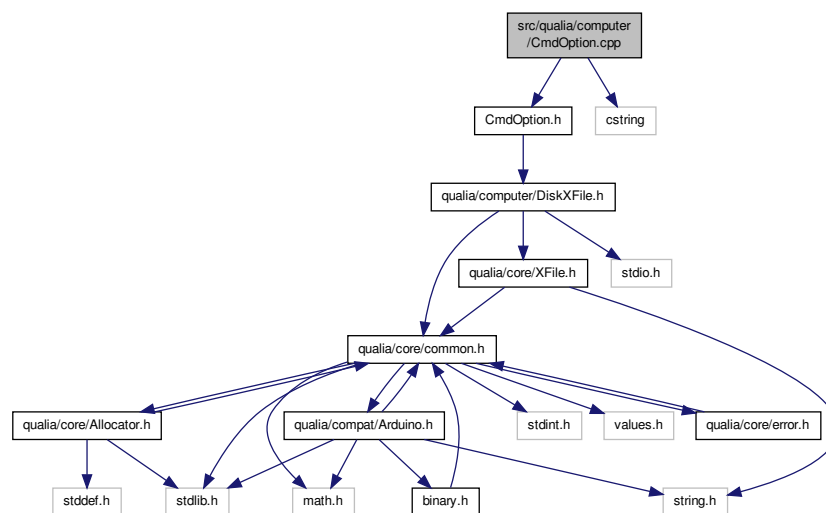
- class [CmdLine](#)

6.17 src/qualia/computer/CmdOption.cpp File Reference

```
#include "CmdOption.h"
```

```
#include <cstring>
```

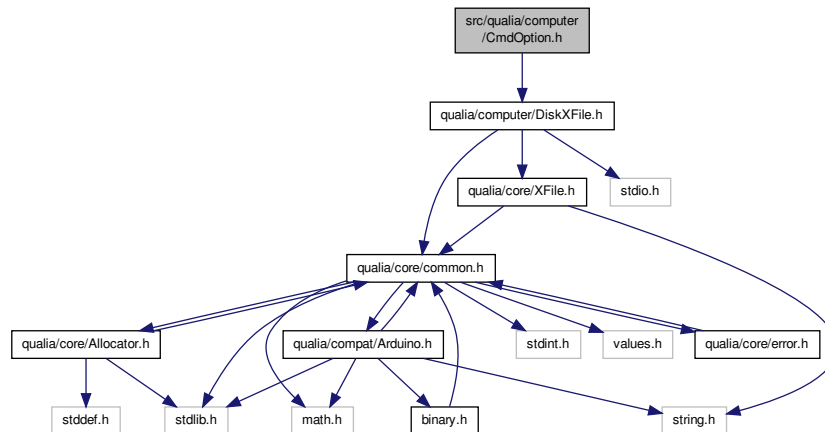
Include dependency graph for CmdOption.cpp:



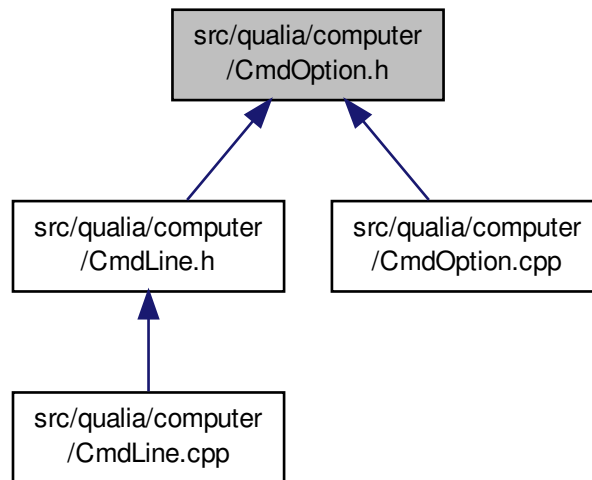
6.18 src/qualia/computer/CmdOption.h File Reference

```
#include <qualia/computer/DiskXFile.h>
```


Include dependency graph for CmdOption.h:



This graph shows which files directly or indirectly include this file:



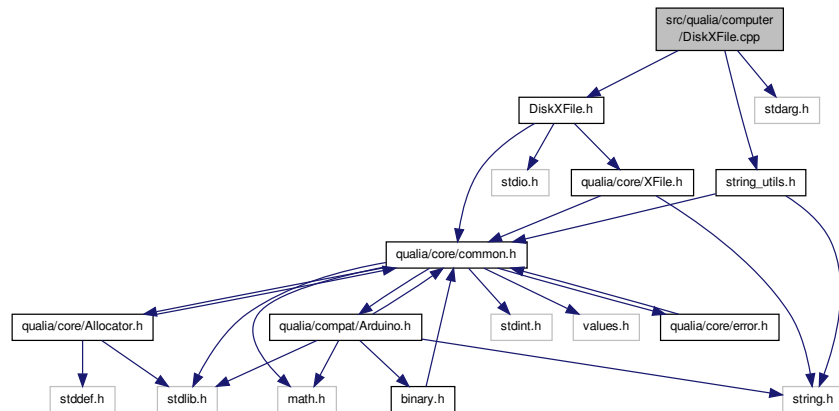
Classes

- class [CmdOption](#)
- class [IntCmdOption](#)
- class [RealCmdOption](#)
- class [BoolCmdOption](#)
- class [StringCmdOption](#)

6.19 src/qualia/computer/DiskXFile.cpp File Reference

```
#include "DiskXFile.h"
#include "string_utils.h"
#include <stdarg.h>
```

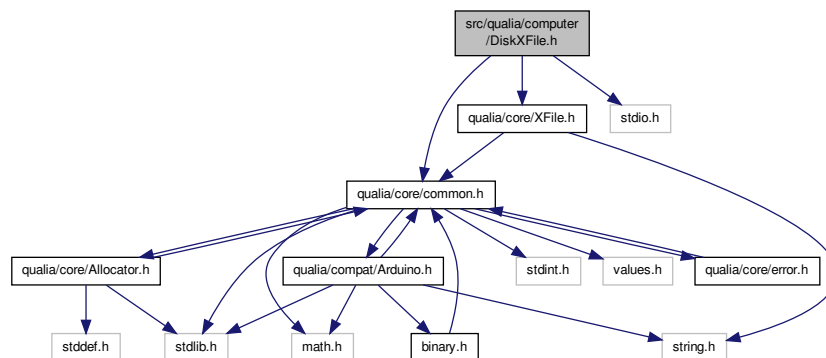
Include dependency graph for DiskXFile.cpp:



6.20 src/qualia/computer/DiskXFile.h File Reference

```
#include <qualia/core/common.h>
#include <qualia/core/XFile.h>
#include <stdio.h>
```

Include dependency graph for DiskXFile.h:



Macros

- `#define REAL_FORMAT "%f"`

6.21.1 Macro Definition Documentation

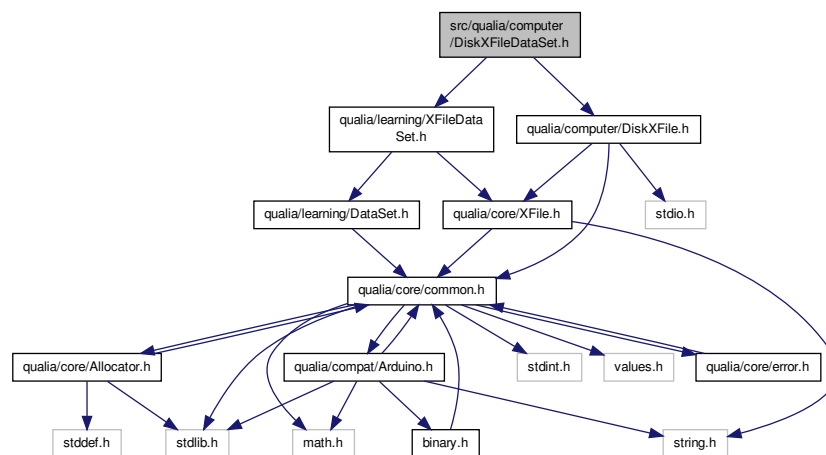
6.21.1.1 `#define REAL_FORMAT "%f"`

6.22 `src/qualia/computer/DiskXFileDataSet.h` File Reference

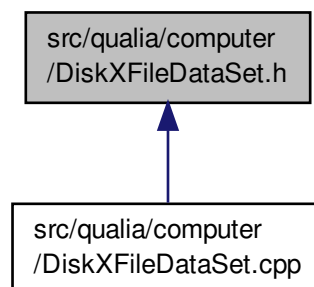
```
#include <qualia/learning/XFileDataSet.h>
```

```
#include <qualia/computer/DiskXFile.h>
```

Include dependency graph for `DiskXFileDataSet.h`:



This graph shows which files directly or indirectly include this file:



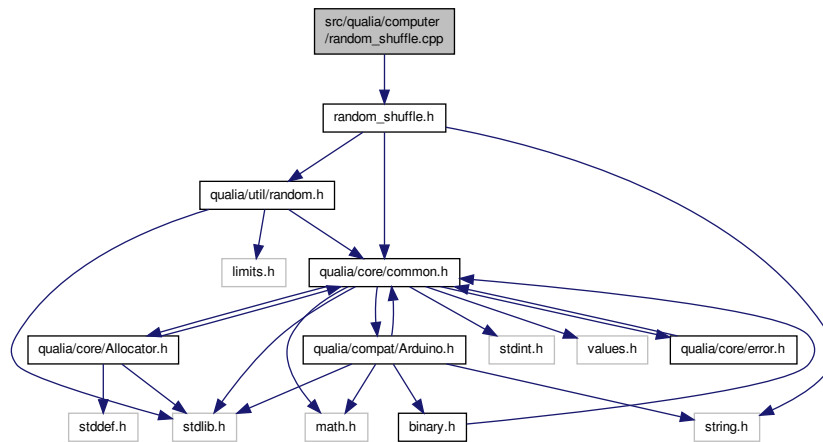
Classes

- class `DiskXFileDataSet`

6.23 src/qualia/computer/random_shuffle.cpp File Reference

```
#include "random_shuffle.h"
```

Include dependency graph for random_shuffle.cpp:



Functions

- void [randomShuffledIndices](#) (int *indices, int nIndices)
- void [randomShuffle](#) (int *tabular, int nElems)
- void [randomShuffle](#) (void *tabular, int sizeElem, int nElems)

6.23.1 Function Documentation

6.23.1.1 void [randomShuffle](#) (int * *tabular*, int *nElems*)

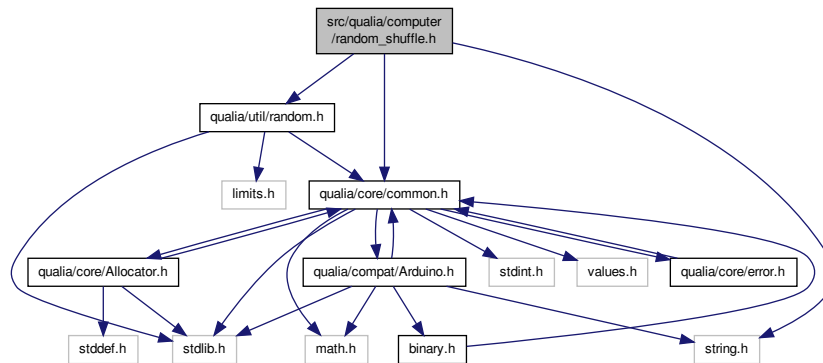
6.23.1.2 void [randomShuffle](#) (void * *tabular*, int *sizeElem*, int *nElems*)

6.23.1.3 void [randomShuffledIndices](#) (int * *indices*, int *nIndices*)

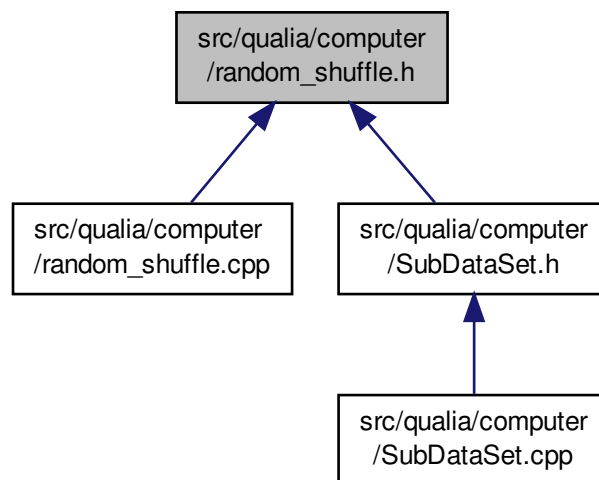
6.24 src/qualia/computer/random_shuffle.h File Reference

```
#include <qualia/core/common.h>
#include <qualia/util/random.h>
#include <string.h>
```

Include dependency graph for `random_shuffle.h`:



This graph shows which files directly or indirectly include this file:



Functions

- void [randomShuffledIndices](#) (int *indices, int nIndices)
- void [randomShuffle](#) (int *tabular, int nElems)
- void [randomShuffle](#) (void *tabular, int sizeElem, int nElem)

6.24.1 Function Documentation

6.24.1.1 void `randomShuffle` (int * *tabular*, int *nElems*)

6.24.1.2 void `randomShuffle` (void * *tabular*, int *sizeElem*, int *nElem*)

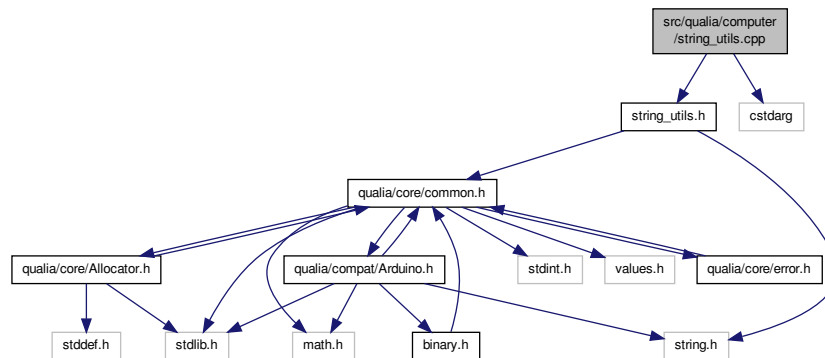
6.24.1.3 void randomShuffledIndices (int * indices, int nIndices)

6.25 src/qualia/computer/string_utils.cpp File Reference

```
#include "string_utils.h"
```

```
#include <stdarg>
```

Include dependency graph for string_utils.cpp:



Functions

- char * [strBaseName](#) (char *filename)
- char * [strRemoveSuffix](#) (char *filename, char c)
- char * [strConcat](#) (int n,...)

6.25.1 Function Documentation

6.25.1.1 char* [strBaseName](#) (char * filename)

Some simple functions for string operations.

Author

Samy Bengio (bengio@idiap.ch)

Johnny Mariethoz (Johnny.Mariethoz@idiap.ch)

Ronan Collobert (collober@idiap.ch) Returns the name of a file without leading pathname. (It's not a new string, but a pointer in the given string)

6.25.1.2 char* [strConcat](#) (int n, ...)

Returns the concatenation #n# strings. The strings are the parameters given after #n#; You have to free the memory!

6.25.1.3 char* [strRemoveSuffix](#) (char * filename, char c = ' . ')

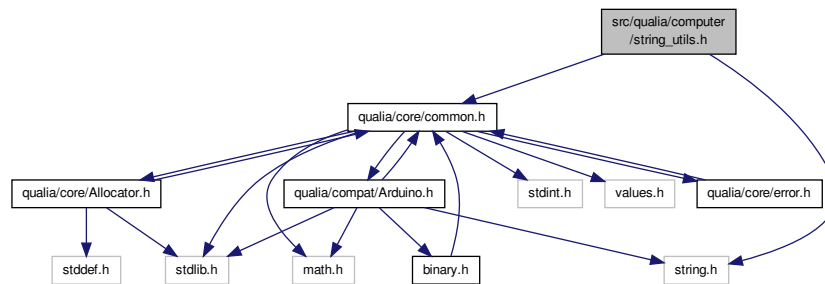
Returns a fresh copy of the name of a file without suffix. (Trailing chars after c) You have to free the memory!

6.26 src/qualia/computer/string_utils.h File Reference

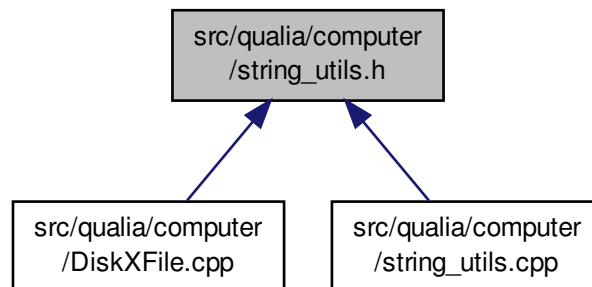
```
#include <qualia/core/common.h>
```

```
#include <string.h>
```

Include dependency graph for string_utils.h:



This graph shows which files directly or indirectly include this file:



Functions

- char * [strBaseName](#) (char *filename)
- char * [strRemoveSuffix](#) (char *filename, char c='.')
- char * [strConcat](#) (int n,...)

6.26.1 Function Documentation

6.26.1.1 char* strBaseName (char * filename)

Some simple functions for string operations.

Author

Samy Bengio (bengio@idiap.ch)

Johnny Mariethoz (Johnny.Mariethoz@idiap.ch)

Ronan Collobert (collober@idiap.ch) Returns the name of a file without leading pathname. (It's not a new string, but a pointer in the given string)

6.26.1.2 char* strConcat (int n, ...)

Returns the concatenation #n# strings. The strings are the parameters given after #n#; You have to free the memory!

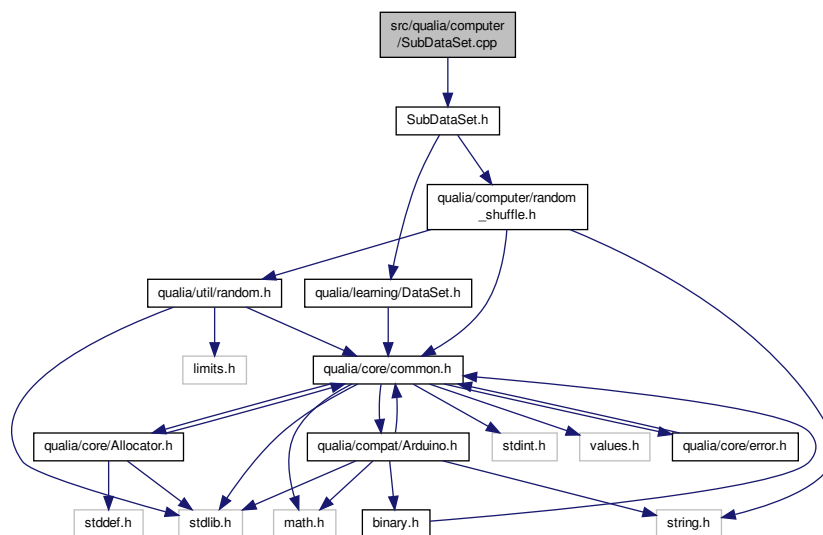
6.26.1.3 char* strRemoveSuffix (char * filename, char c = ' .')

Returns a fresh copy of the name of a file without suffix. (Trailing chars after c) You have to free the memory!

6.27 src/qualia/computer/SubDataSet.cpp File Reference

```
#include "SubDataSet.h"
```

Include dependency graph for SubDataSet.cpp:

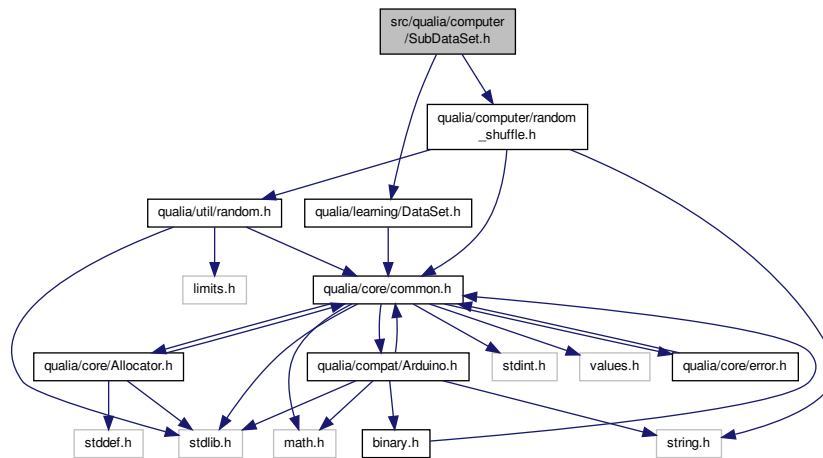


6.28 src/qualia/computer/SubDataSet.h File Reference

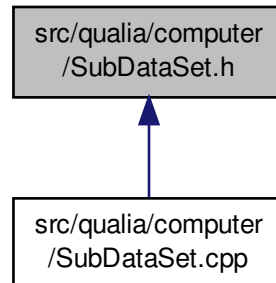
```
#include <qualia/learning/DataSet.h>
```

```
#include <qualia/computer/random_shuffle.h>
```

Include dependency graph for SubDataSet.h:



This graph shows which files directly or indirectly include this file:



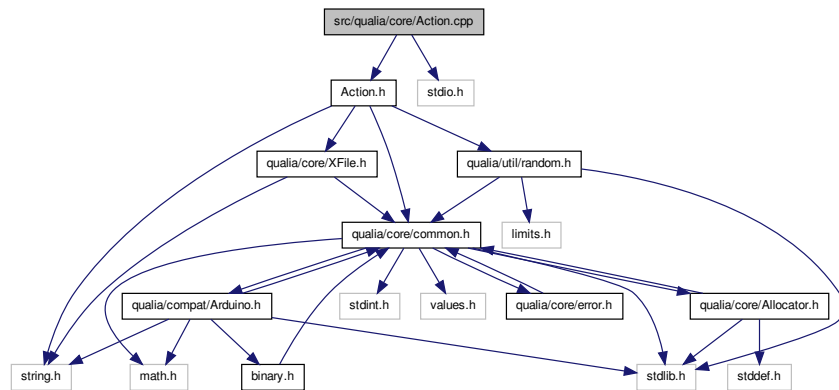
Classes

- class [SubDataSet](#)

6.29 src/qualia/core/Action.cpp File Reference

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

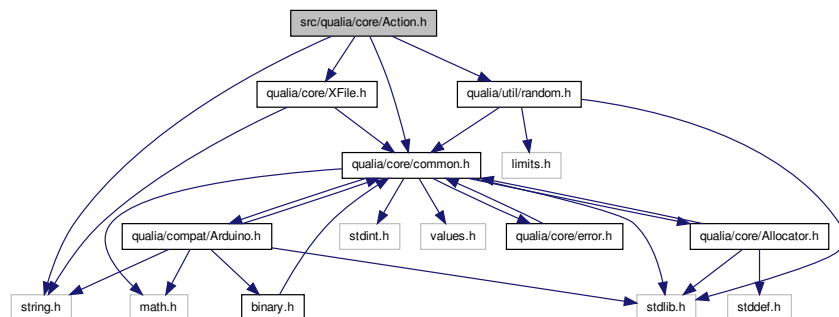
Include dependency graph for Action.cpp:



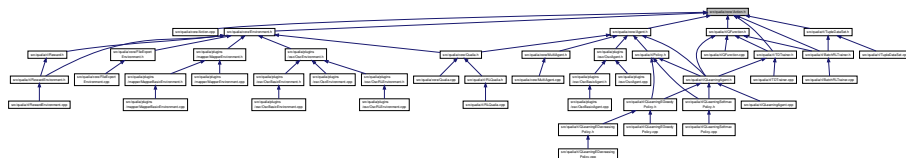
6.30 src/qualia/core/Action.h File Reference

```
#include <qualia/core/common.h>
#include <qualia/core/XFile.h>
#include <qualia/util/random.h>
#include <string.h>
```

Include dependency graph for Action.h:



This graph shows which files directly or indirectly include this file:



Classes

- class `ActionProperties`
- class `Action`

Typedefs

- typedef unsigned long [action_t](#)
- typedef unsigned int [action_dim_t](#)

6.30.1 Typedef Documentation

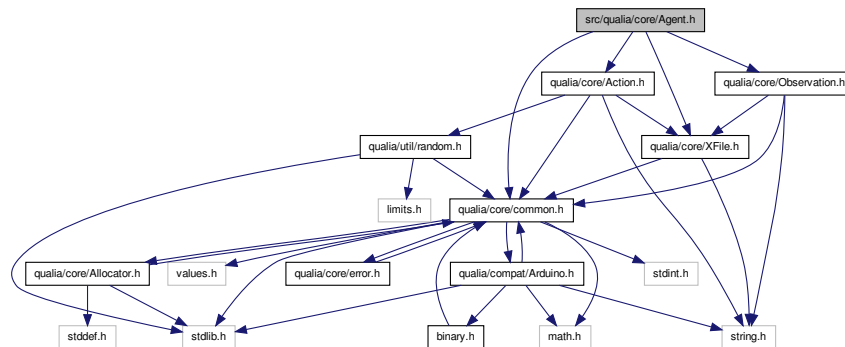
6.30.1.1 typedef unsigned int [action_dim_t](#)

6.30.1.2 typedef unsigned long [action_t](#)

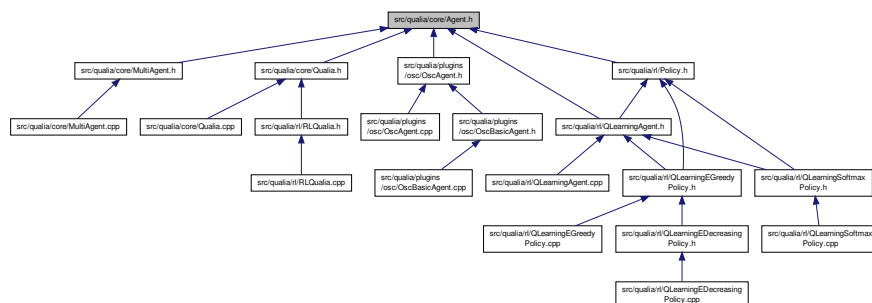
6.31 src/qualia/core/Agent.h File Reference

```
#include <qualia/core/common.h>
#include <qualia/core/Action.h>
#include <qualia/core/Observation.h>
#include <qualia/core/XFile.h>
```

Include dependency graph for Agent.h:



This graph shows which files directly or indirectly include this file:



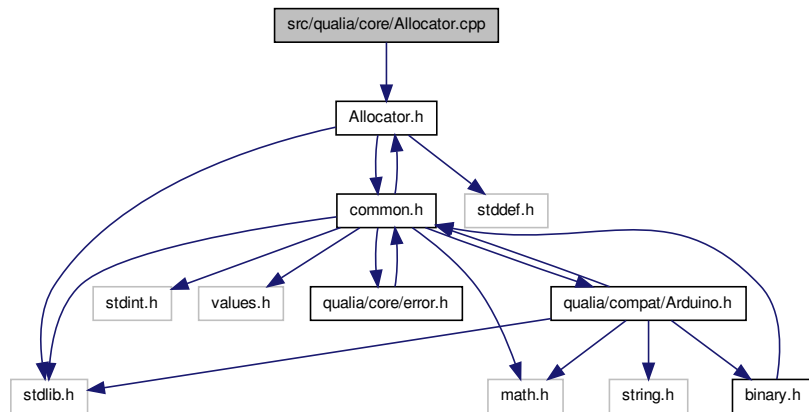
Classes

- class [Agent](#)

6.32 src/qualia/core/Allocator.cpp File Reference

```
#include "Allocator.h"
```

Include dependency graph for Allocator.cpp:



Functions

- void * [operator new](#) (size_t size, [Allocator](#) *alloc)

6.32.1 Function Documentation

6.32.1.1 void* operator new (size_t size, Allocator * alloc)

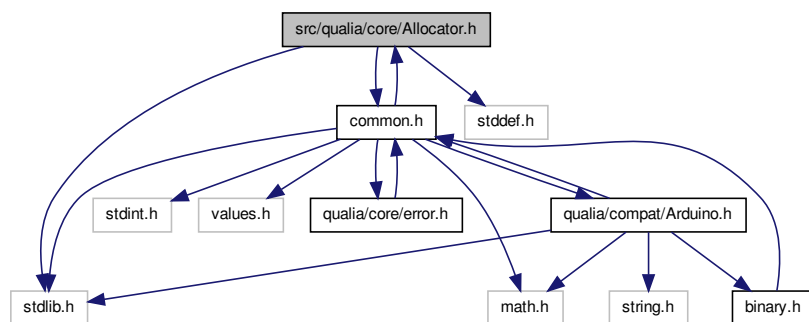
6.33 src/qualia/core/Allocator.h File Reference

```
#include "common.h"
```

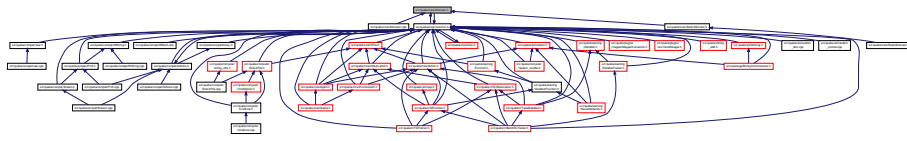
```
#include <stddef.h>
```

```
#include <stdlib.h>
```

Include dependency graph for Allocator.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Allocator](#)
- class [Alloc](#)

Functions

- void * [operator new](#) (size_t size, [Allocator](#) *alloc)

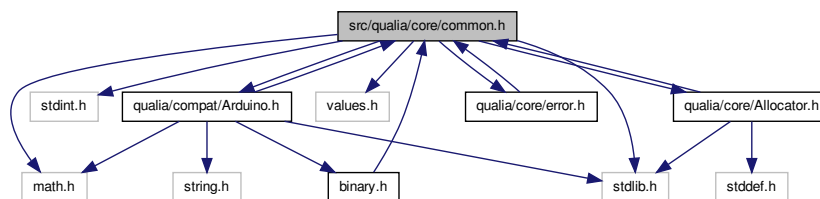
6.33.1 Function Documentation

6.33.1.1 void* operator new (size_t size, Allocator * alloc)

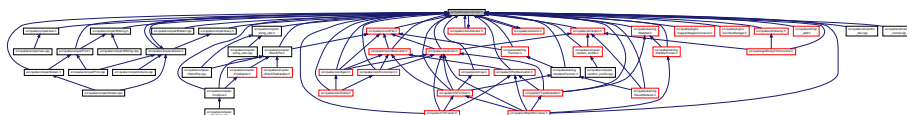
6.34 src/qualia/core/common.h File Reference

```
#include <stdlib.h>
#include <stdint.h>
#include <math.h>
#include <values.h>
#include <qualia/compat/Arduino.h>
#include <qualia/core/error.h>
#include <qualia/core/Allocator.h>
```

Include dependency graph for common.h:



This graph shows which files directly or indirectly include this file:



Macros

- `#define is_avr() defined(__AVR__)`
True iff we are compiling for AVR.
- `#define is_arduino() defined(ARDUINO)`
True iff we are compiling for Arduino (notice that `is_arduino()` implies `is_avr()`)
- `#define is_computer() !defined(__AVR__)`
True iff we are compiling for a non-AVR platform.
- `#define bitFlip(value, bit) ((value) ^= (1UL << (bit)))`
Flips bit `bit#` in element `#value#`.
- `#define INF FLT_MAX`
- `#define RANDOM_MAX RAND_MAX`

Typedefs

- typedef float `real`

6.34.1 Macro Definition Documentation

6.34.1.1 `#define bitFlip(value, bit) ((value) ^= (1UL << (bit)))`

Flips bit `bit#` in element `#value#`.

6.34.1.2 `#define INF FLT_MAX`

6.34.1.3 `#define is_arduino() defined(ARDUINO)`

True iff we are compiling for Arduino (notice that `is_arduino()` implies `is_avr()`)

6.34.1.4 `#define is_avr() defined(__AVR__)`

True iff we are compiling for AVR.

This file provides some common includes and definition. Among other things, it gives access to some common Arduino functions when compiling on a computer. This file should be included at the top of every header file that uses [Qualia](#).

6.34.1.5 `#define is_computer() !defined(__AVR__)`

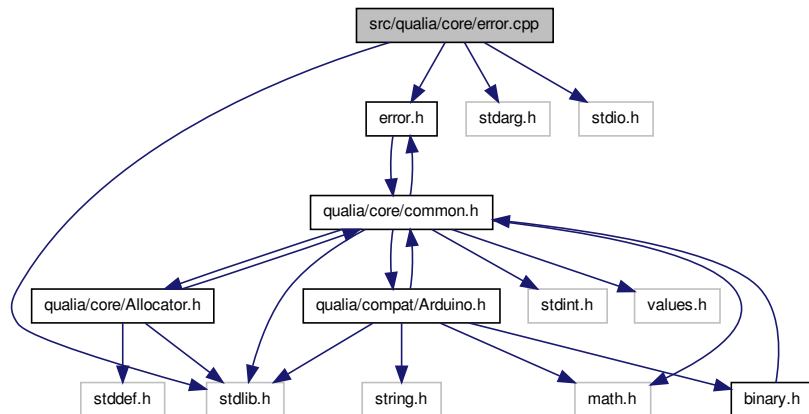
True iff we are compiling for a non-AVR platform.

6.34.1.6 `#define RANDOM_MAX RAND_MAX`

6.34.2 Typedef Documentation

6.34.2.1 typedef float `real`

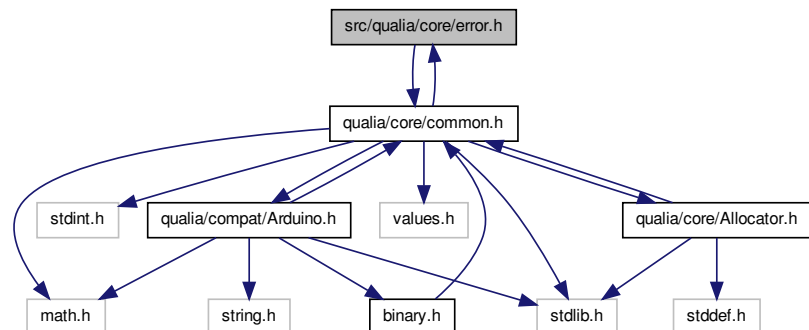
Include dependency graph for error.cpp:



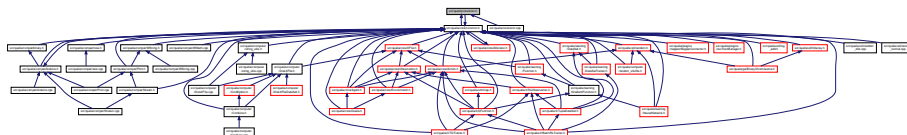
6.37 src/qualia/core/error.h File Reference

```
#include <qualia/core/common.h>
```

Include dependency graph for error.h:



This graph shows which files directly or indirectly include this file:



Macros

- `#define DEBUG_LEVEL DEBUG_LEVEL_NODEBUG`
- `#define DEBUG_LEVEL_NODEBUG -1`

- No debug.*
- `#define DEBUG_LEVEL_ERROR 0`
- Only catch errors.*
- `#define DEBUG_LEVEL_WARNING 1`
- Catch both errors and warnings.*
- `#define DEBUG_LEVEL_NOTICE 2`
- Catch errors, warnings and notices.*
- `#define DEBUG_ERROR (DEBUG_LEVEL >= DEBUG_LEVEL_ERROR)`
- `#define DEBUG_WARNING (DEBUG_LEVEL >= DEBUG_LEVEL_WARNING)`
- `#define DEBUG_NOTICE (DEBUG_LEVEL >= DEBUG_LEVEL_NOTICE)`
- `#define __STRING(x) #x`
- `#define __DEBUG_TRIGGER_ASSERT(expr, func) ((expr) ? static_cast<void>(0) : func("Fail: " __STRING(expr)))`;
- `#define __DEBUG_DUMMY_INSTRUCTION (static_cast<void>(0))`
- `#define Q_ERROR(...) __DEBUG_DUMMY_INSTRUCTION`
- `#define Q_ASSERT_ERROR_MESSAGE(expr,...) __DEBUG_DUMMY_INSTRUCTION`
- `#define Q_ASSERT_ERROR(expr) __DEBUG_DUMMY_INSTRUCTION`
- `#define Q_WARNING(...) __DEBUG_DUMMY_INSTRUCTION`
- `#define Q_ASSERT_WARNING_MESSAGE(expr,...) __DEBUG_DUMMY_INSTRUCTION`
- `#define Q_ASSERT_WARNING(expr) __DEBUG_DUMMY_INSTRUCTION`
- `#define Q_NOTICE(...) __DEBUG_DUMMY_INSTRUCTION`
- `#define Q_ASSERT_NOTICE_MESSAGE(expr,...) __DEBUG_DUMMY_INSTRUCTION`
- `#define Q_ASSERT_NOTICE(expr) __DEBUG_DUMMY_INSTRUCTION`
- `#define Q_PRINT(...) __DEBUG_DUMMY_INSTRUCTION`
- `#define Q_MESSAGE(...) __DEBUG_DUMMY_INSTRUCTION`

6.37.1 Macro Definition Documentation

6.37.1.1 `#define __DEBUG_DUMMY_INSTRUCTION (static_cast<void>(0))`

6.37.1.2 `#define __DEBUG_TRIGGER_ASSERT(expr, func) ((expr) ? static_cast<void>(0) : func("Fail: " __STRING(expr)))`;

6.37.1.3 `#define __STRING(x) #x`

6.37.1.4 `#define DEBUG_ERROR (DEBUG_LEVEL >= DEBUG_LEVEL_ERROR)`

6.37.1.5 `#define DEBUG_LEVEL DEBUG_LEVEL_NODEBUG`

Error/warning/notification routines/macros. Level of debugging can be set by defining `DEBUG_LEVEL` at compile time. Assertions and messages that are below the level will simply be erased by the preprocessor (macros are actually replaced by a dummy/void operation), thus saving on memory usage and execution time.

WARNING: Notice that `DEBUG_LEVEL` is not supported on AVR/Arduino ie. all error macros are disabled on such platforms.

6.37.1.6 `#define DEBUG_LEVEL_ERROR 0`

Only catch errors.

6.37.1.7 `#define DEBUG_LEVEL_NODEBUG -1`

No debug.

6.37.1.8 `#define DEBUG_LEVEL_NOTICE 2`

Catch errors, warnings and notices.

6.37.1.9 `#define DEBUG_LEVEL_WARNING 1`

Catch both errors and warnings.

6.37.1.10 `#define DEBUG_NOTICE (DEBUG_LEVEL >= DEBUG_LEVEL_NOTICE)`

6.37.1.11 `#define DEBUG_WARNING (DEBUG_LEVEL >= DEBUG_LEVEL_WARNING)`

6.37.1.12 `#define Q_ASSERT_ERROR(expr) __DEBUG_DUMMY_INSTRUCTION`

6.37.1.13 `#define Q_ASSERT_ERROR_MESSAGE(expr, ...) __DEBUG_DUMMY_INSTRUCTION`

6.37.1.14 `#define Q_ASSERT_NOTICE(expr) __DEBUG_DUMMY_INSTRUCTION`

6.37.1.15 `#define Q_ASSERT_NOTICE_MESSAGE(expr, ...) __DEBUG_DUMMY_INSTRUCTION`

6.37.1.16 `#define Q_ASSERT_WARNING(expr) __DEBUG_DUMMY_INSTRUCTION`

6.37.1.17 `#define Q_ASSERT_WARNING_MESSAGE(expr, ...) __DEBUG_DUMMY_INSTRUCTION`

6.37.1.18 `#define Q_ERROR(...) __DEBUG_DUMMY_INSTRUCTION`

6.37.1.19 `#define Q_MESSAGE(...) __DEBUG_DUMMY_INSTRUCTION`

6.37.1.20 `#define Q_NOTICE(...) __DEBUG_DUMMY_INSTRUCTION`

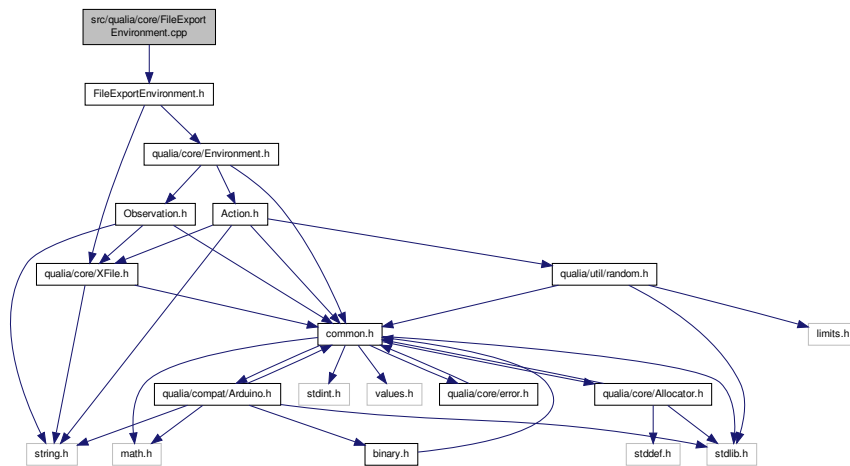
6.37.1.21 `#define Q_PRINT(...) __DEBUG_DUMMY_INSTRUCTION`

6.37.1.22 `#define Q_WARNING(...) __DEBUG_DUMMY_INSTRUCTION`

6.38 src/qualia/core/FileExportEnvironment.cpp File Reference

```
#include "FileExportEnvironment.h"
```

Include dependency graph for FileExportEnvironment.cpp:

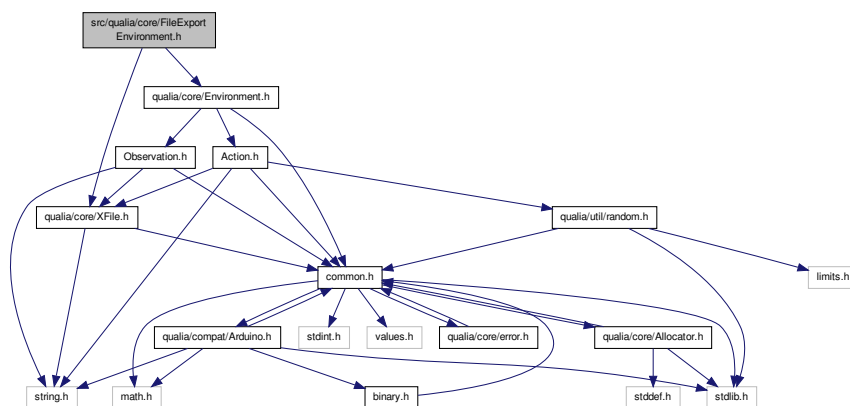


6.39 src/qualia/core/FileExportEnvironment.h File Reference

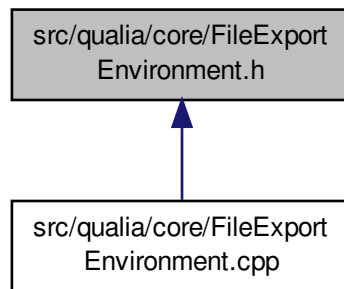
```
#include <qualia/core/Environment.h>
```

```
#include <qualia/core/XFile.h>
```

Include dependency graph for FileExportEnvironment.h:



This graph shows which files directly or indirectly include this file:



Classes

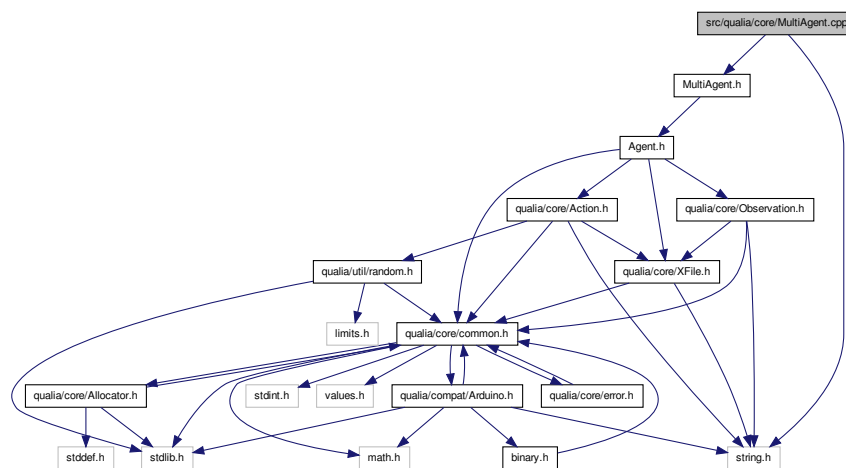
- class [FileExportEnvironment](#)

6.40 src/qualia/core/MultiAgent.cpp File Reference

```
#include "MultiAgent.h"
```

```
#include <string.h>
```

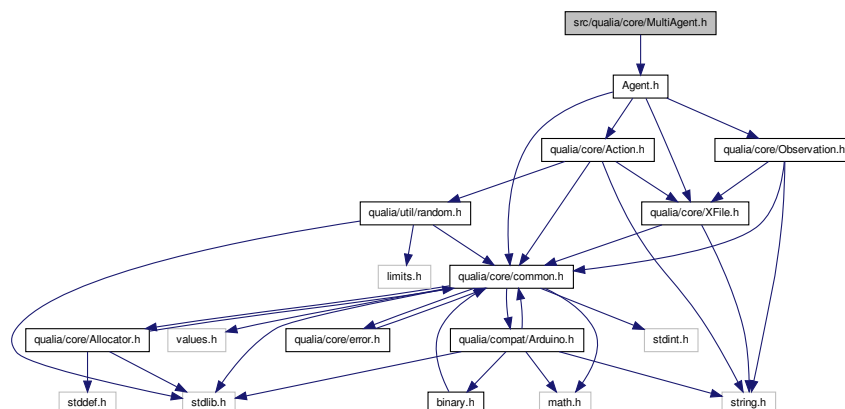
Include dependency graph for MultiAgent.cpp:



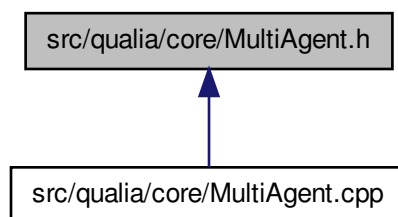
6.41 src/qualia/core/MultiAgent.h File Reference

```
#include "Agent.h"
```

Include dependency graph for MultiAgent.h:



This graph shows which files directly or indirectly include this file:



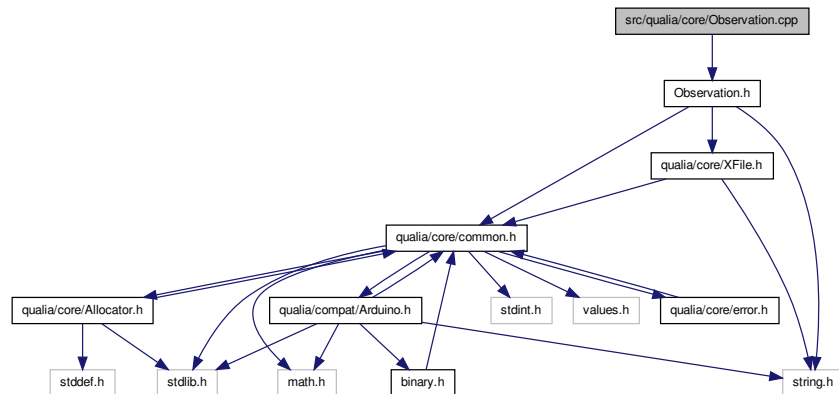
Classes

- class [MultiAgent](#)

6.42 src/qualia/core/Observation.cpp File Reference

```
#include "Observation.h"
```

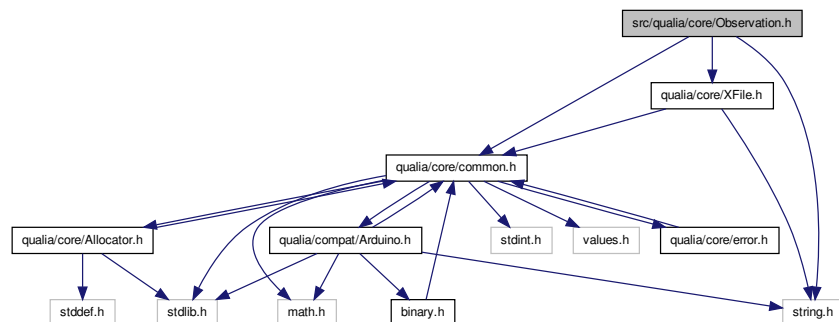
Include dependency graph for Observation.cpp:



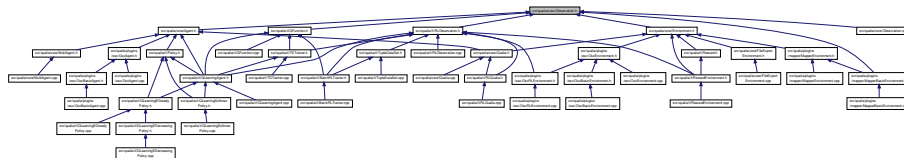
6.43 src/qualia/core/Observation.h File Reference

```
#include <qualia/core/common.h>
#include <qualia/core/XFile.h>
#include <string.h>
```

Include dependency graph for Observation.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Observation](#)

Typedefs

- typedef [real observation_t](#)

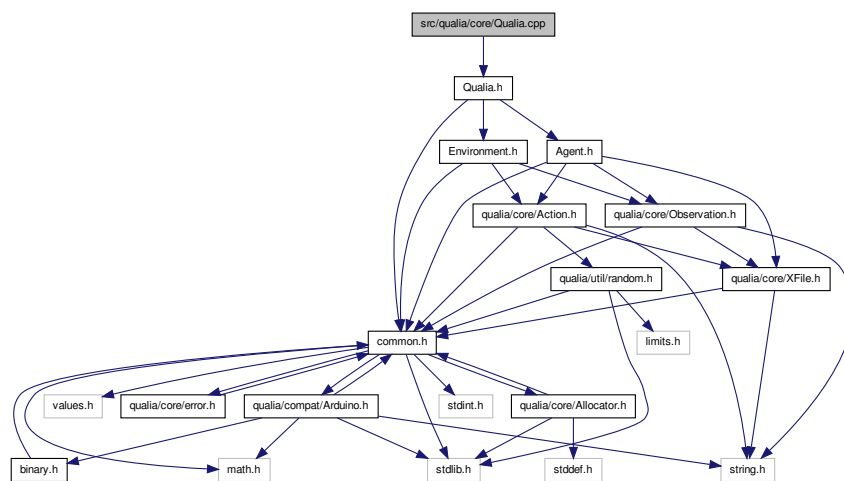
6.43.1 Typedef Documentation

6.43.1.1 typedef real observation_t

6.44 src/qualia/core/Qualia.cpp File Reference

```
#include "Qualia.h"
```

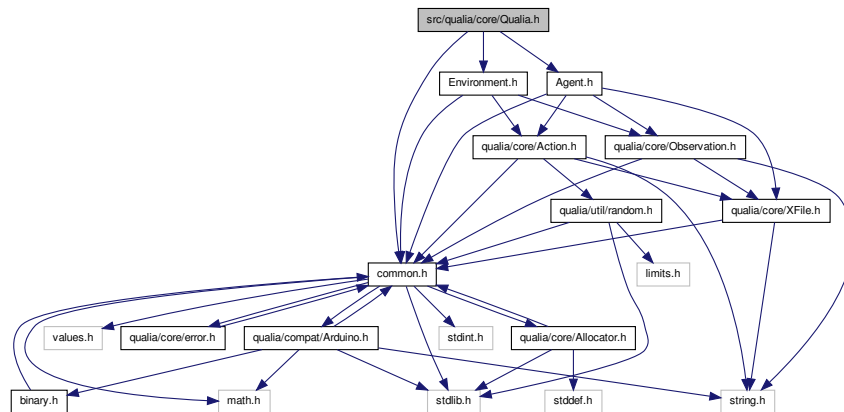
Include dependency graph for Qualia.cpp:



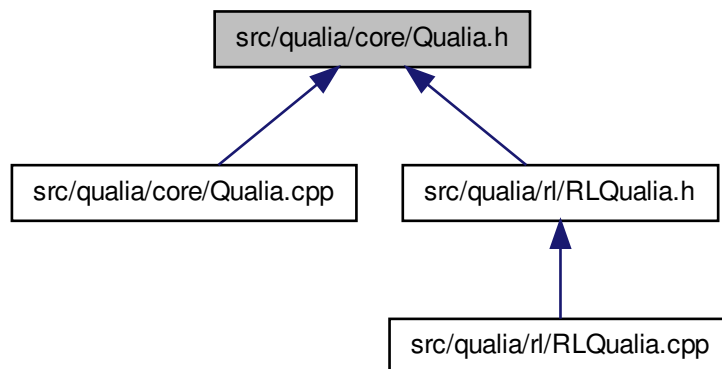
6.45 src/qualia/core/Qualia.h File Reference

```
#include "common.h"
#include "Agent.h"
#include "Environment.h"
```


Include dependency graph for Qualia.h:



This graph shows which files directly or indirectly include this file:



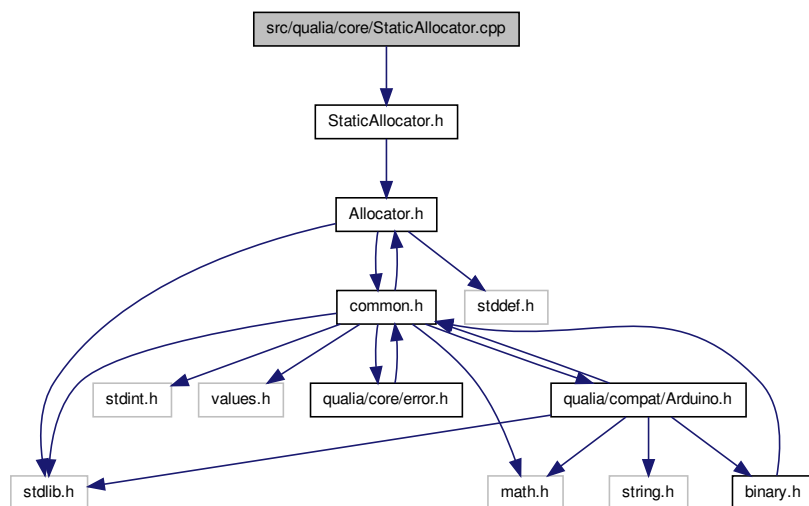
Classes

- struct [ObservationAction](#)
- class [Qualia](#)

6.46 src/qualia/core/StaticAllocator.cpp File Reference

```
#include "StaticAllocator.h"
```

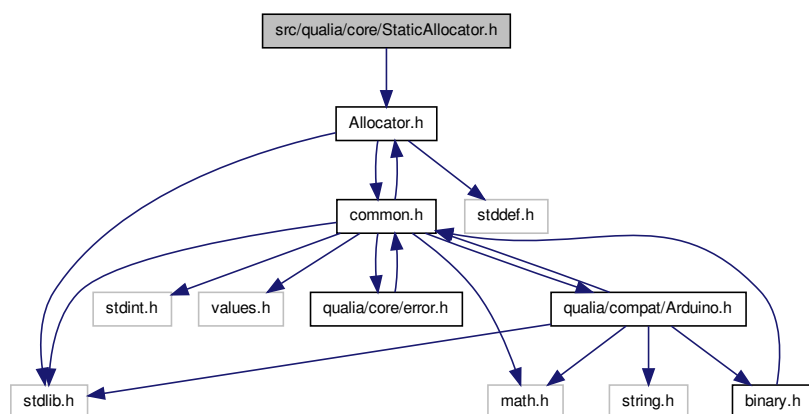
Include dependency graph for StaticAllocator.cpp:



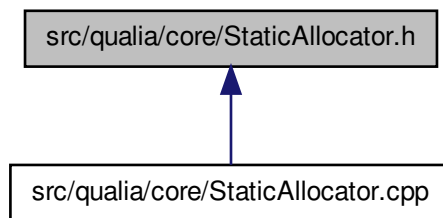
6.47 src/qualia/core/StaticAllocator.h File Reference

```
#include "Allocator.h"
```

Include dependency graph for StaticAllocator.h:



This graph shows which files directly or indirectly include this file:



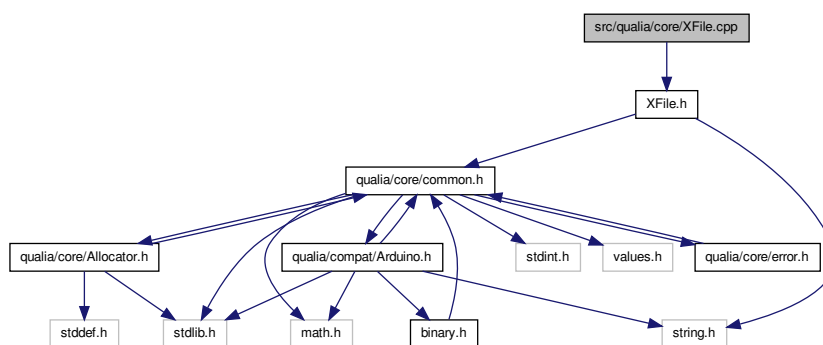
Classes

- class [StaticAllocator](#)

6.48 src/qualia/core/XFile.cpp File Reference

```
#include "XFile.h"
```

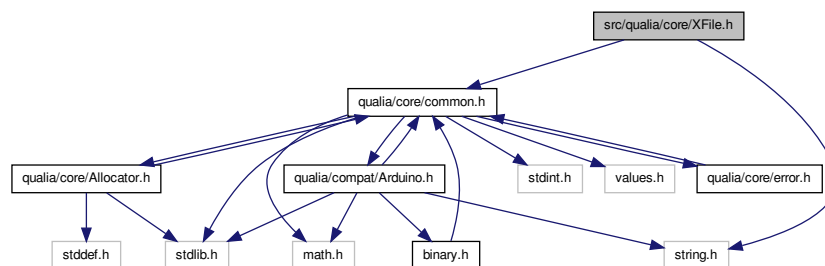
Include dependency graph for XFile.cpp:



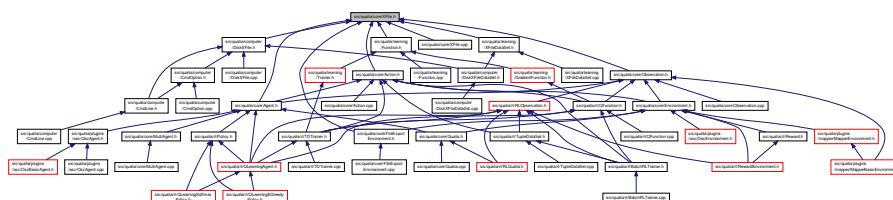
6.49 src/qualia/core/XFile.h File Reference

```
#include <qualia/core/common.h>
#include <string.h>
```

Include dependency graph for XFile.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [XFile](#)

Macros

- `#define` [SEEK_SET](#) 0x0
- `#define` [SEEK_CUR](#) 0x1
- `#define` [SEEK_END](#) 0x2

6.49.1 Macro Definition Documentation

6.49.1.1 `#define` [SEEK_CUR](#) 0x1

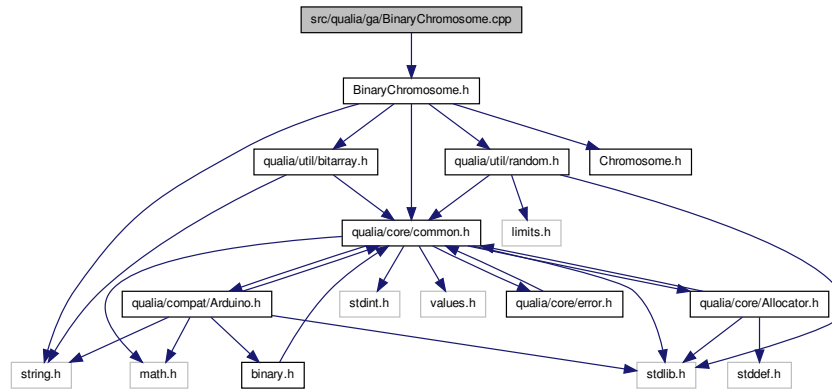
6.49.1.2 `#define` [SEEK_END](#) 0x2

6.49.1.3 `#define` [SEEK_SET](#) 0x0

6.50 src/qualia/ga/BinaryChromosome.cpp File Reference

```
#include "BinaryChromosome.h"
```

Include dependency graph for BinaryChromosome.cpp:



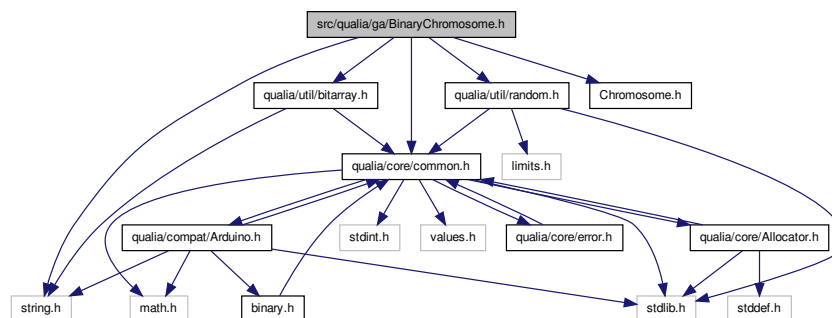
6.51 src/qualia/ga/BinaryChromosome.h File Reference

```

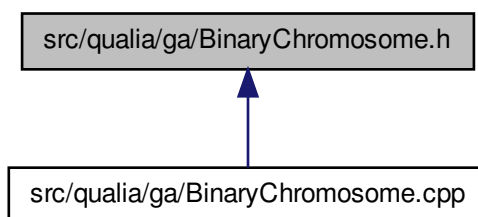
#include <qualia/core/common.h>
#include <qualia/util/bitarray.h>
#include <qualia/util/random.h>
#include "Chromosome.h"
#include <string.h>

```

Include dependency graph for BinaryChromosome.h:



This graph shows which files directly or indirectly include this file:

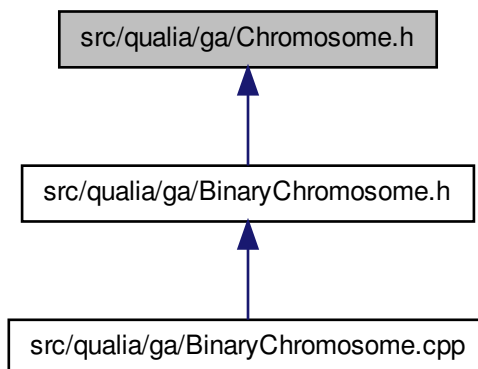


Classes

- class [BinaryChromosomeProperties](#)
- class [BinaryChromosome](#)

6.52 src/qualia/ga/Chromosome.h File Reference

This graph shows which files directly or indirectly include this file:



Classes

- class [Chromosome](#)

Typedefs

- typedef float(* [Evaluator](#))(Chromosome &)

- typedef void(* [Initializer](#))(Chromosome &)
- typedef void(* [Mutator](#))(Chromosome &, float)
- typedef int(* [Comparator](#))(const Chromosome &, const Chromosome &)
- typedef void(* [SexualCrossover](#))(const Chromosome &, const Chromosome &, Chromosome *, Chromosome *)
- typedef void(* [AsexualCrossover](#))(const Chromosome &, Chromosome *)

6.52.1 Typedef Documentation

6.52.1.1 typedef void(* [AsexualCrossover](#))(const Chromosome &, Chromosome *)

6.52.1.2 typedef int(* [Comparator](#))(const Chromosome &, const Chromosome &)

6.52.1.3 typedef float(* [Evaluator](#))(Chromosome &)

6.52.1.4 typedef void(* [Initializer](#))(Chromosome &)

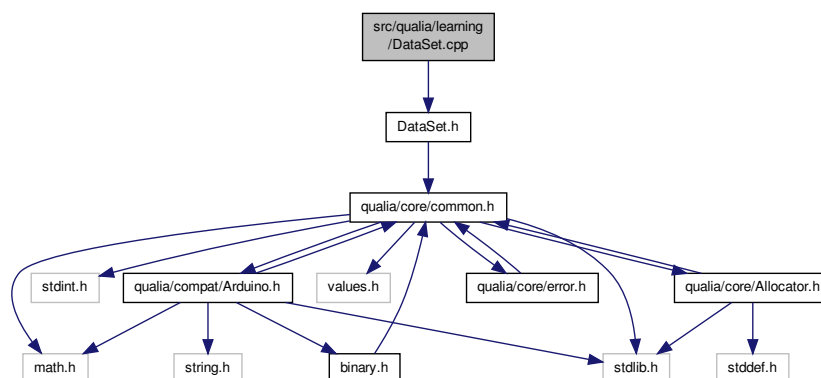
6.52.1.5 typedef void(* [Mutator](#))(Chromosome &, float)

6.52.1.6 typedef void(* [SexualCrossover](#))(const Chromosome &, const Chromosome &, Chromosome *, Chromosome *)

6.53 src/qualia/learning/DataSet.cpp File Reference

```
#include "DataSet.h"
```

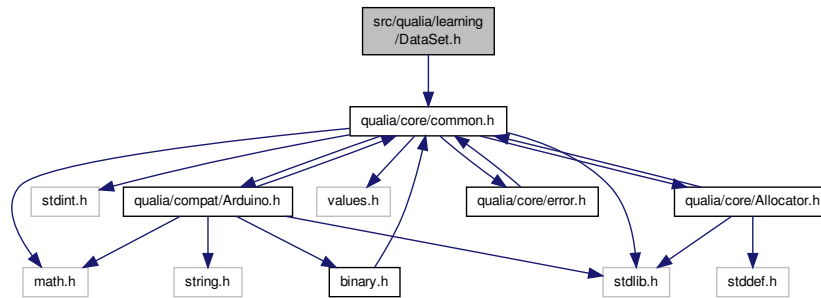
Include dependency graph for DataSet.cpp:



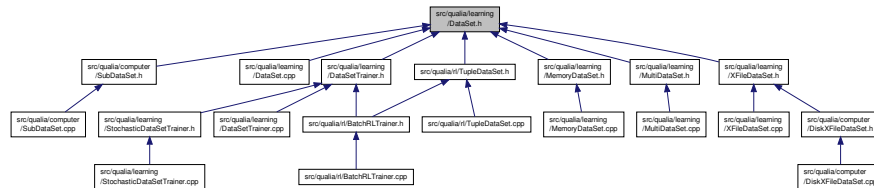
6.54 src/qualia/learning/DataSet.h File Reference

```
#include <qualia/core/common.h>
```

Include dependency graph for DataSet.h:



This graph shows which files directly or indirectly include this file:



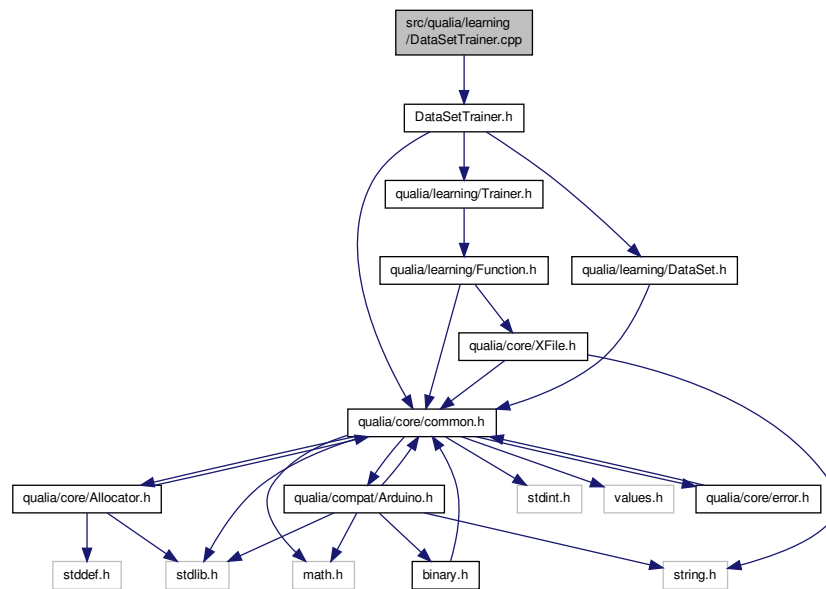
Classes

- class [DataSet](#)

6.55 src/qualia/learning/DataSetTrainer.cpp File Reference

```
#include "DataSetTrainer.h"
```

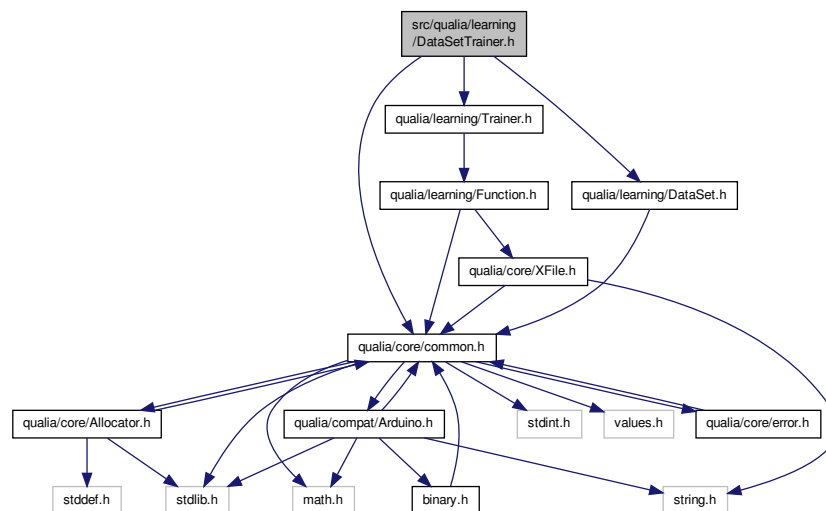

Include dependency graph for DataSetTrainer.cpp:



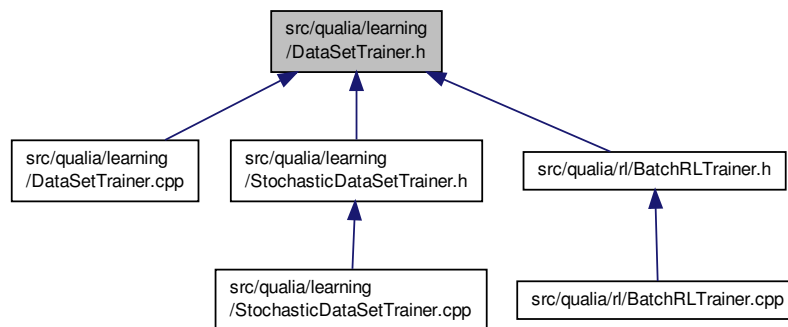
6.56 src/qualia/learning/DataSetTrainer.h File Reference

```
#include <qualia/core/common.h>
#include <qualia/learning/Trainer.h>
#include <qualia/learning/DataSet.h>
```

Include dependency graph for DataSetTrainer.h:



This graph shows which files directly or indirectly include this file:



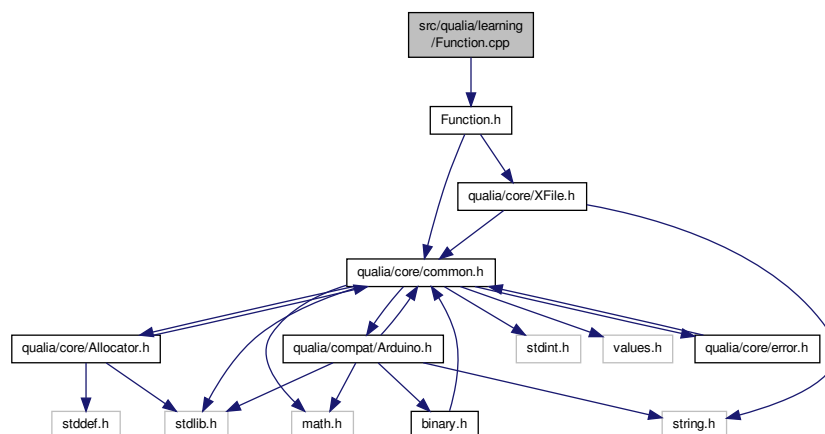
Classes

- class [DataSetTrainer](#)

6.57 src/qualia/learning/Function.cpp File Reference

```
#include "Function.h"
```

Include dependency graph for Function.cpp:

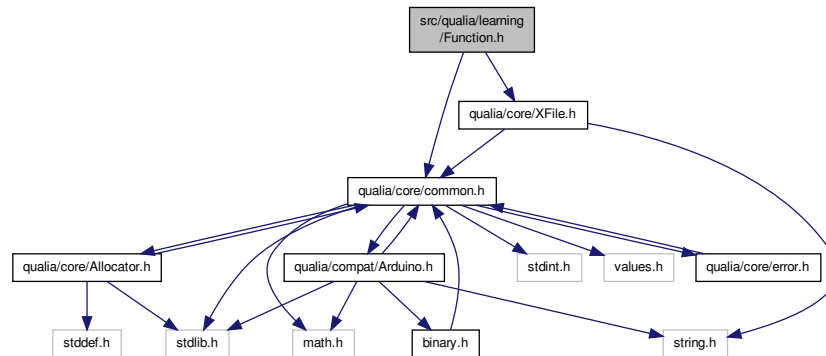


6.58 src/qualia/learning/Function.h File Reference

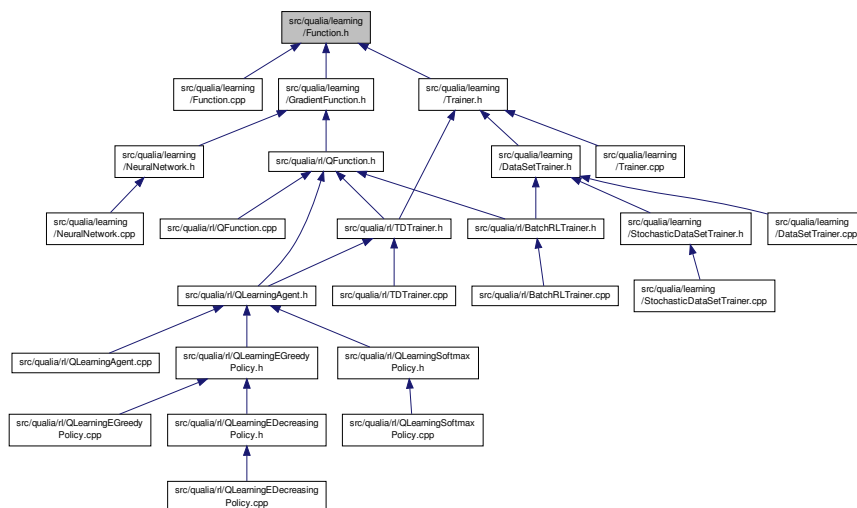
```
#include <qualia/core/common.h>
```

```
#include <qualia/core/XFile.h>
```

Include dependency graph for Function.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [Function](#)

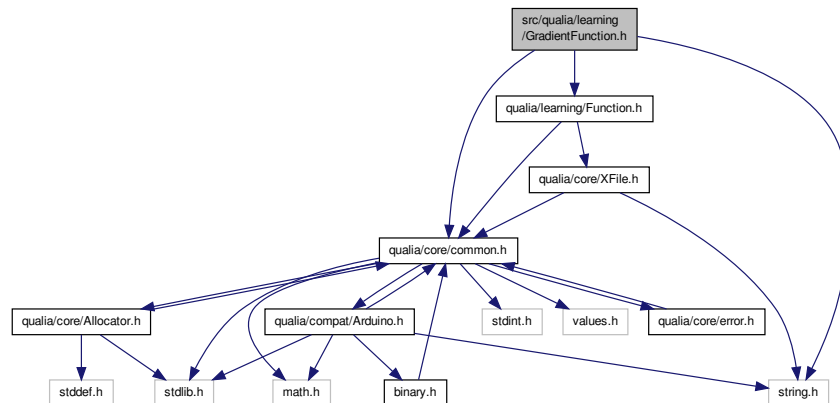
6.59 src/qualia/learning/GradientFunction.h File Reference

```

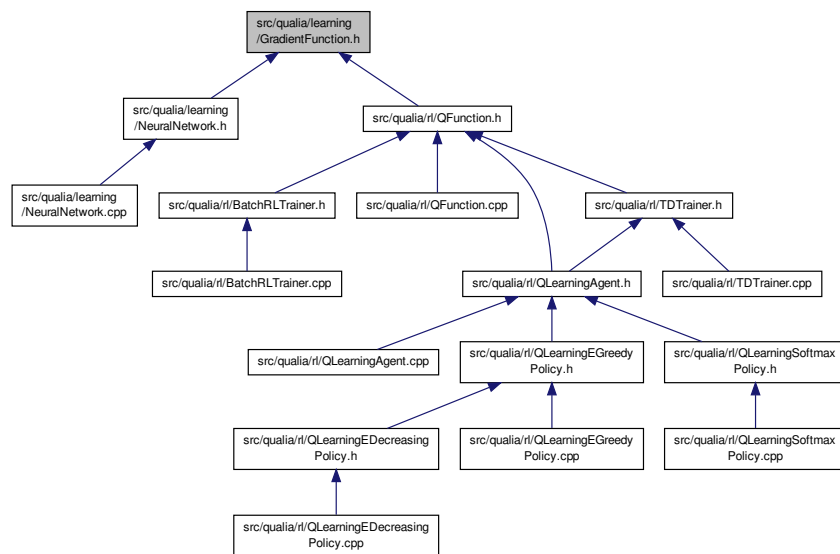
#include <qualia/core/common.h>
#include <qualia/learning/Function.h>
#include <string.h>

```

Include dependency graph for GradientFunction.h:



This graph shows which files directly or indirectly include this file:



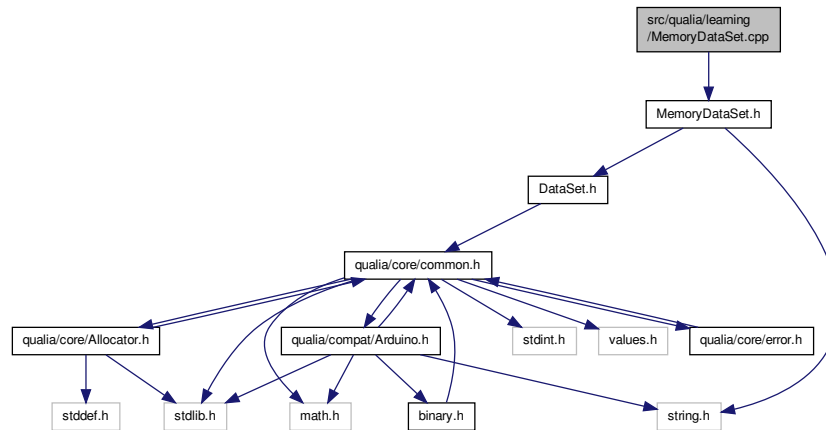
Classes

- class [GradientFunction](#)

6.60 src/qualia/learning/MemoryDataSet.cpp File Reference

```
#include "MemoryDataSet.h"
```

Include dependency graph for MemoryDataSet.cpp:

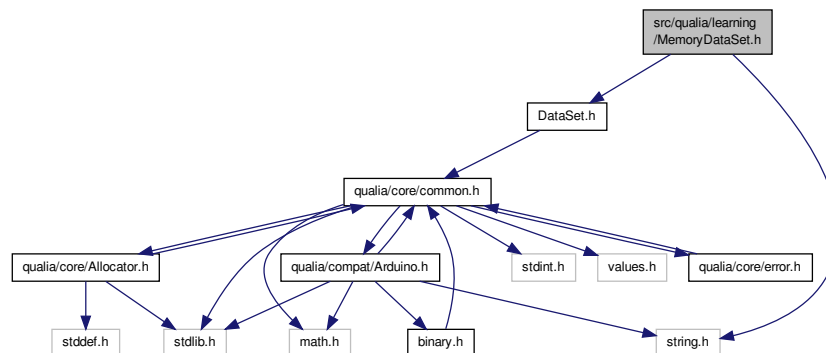


6.61 src/qualia/learning/MemoryDataSet.h File Reference

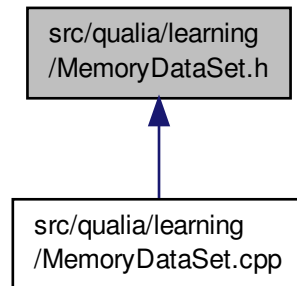
```
#include "DataSet.h"
```

```
#include <string.h>
```

Include dependency graph for MemoryDataSet.h:



This graph shows which files directly or indirectly include this file:



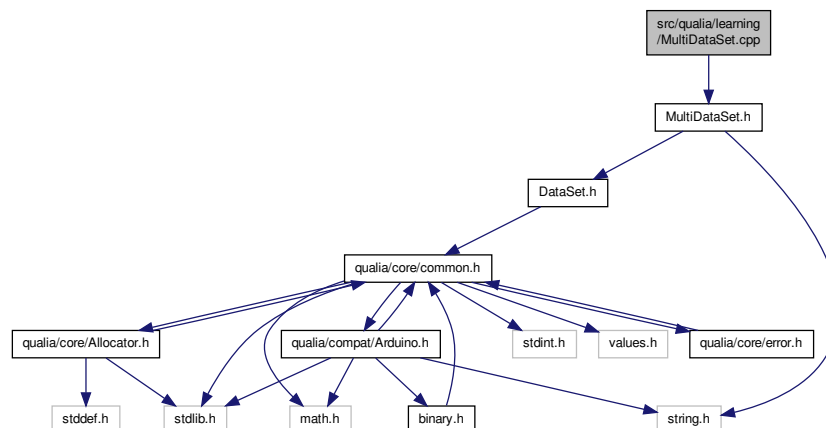
Classes

- class [MemoryDataSet](#)

6.62 src/qualia/learning/MultiDataSet.cpp File Reference

```
#include "MultiDataSet.h"
```

Include dependency graph for `MultiDataSet.cpp`:

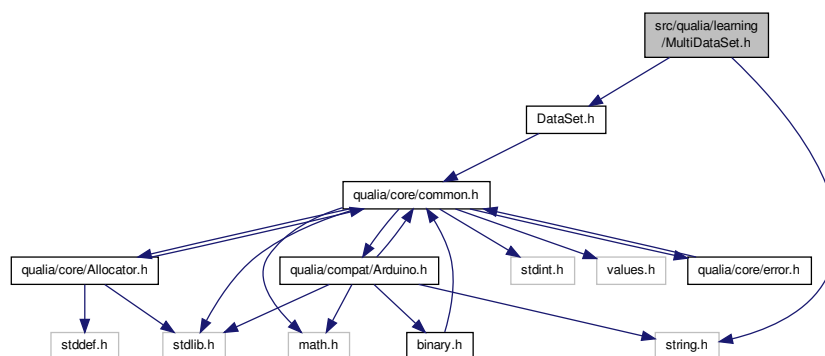


6.63 src/qualia/learning/MultiDataSet.h File Reference

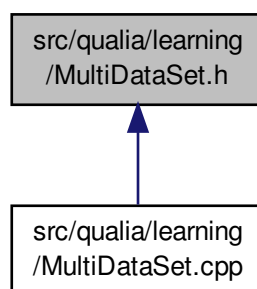
```
#include "DataSet.h"
```

```
#include <string.h>
```

Include dependency graph for MultiDataSet.h:



This graph shows which files directly or indirectly include this file:



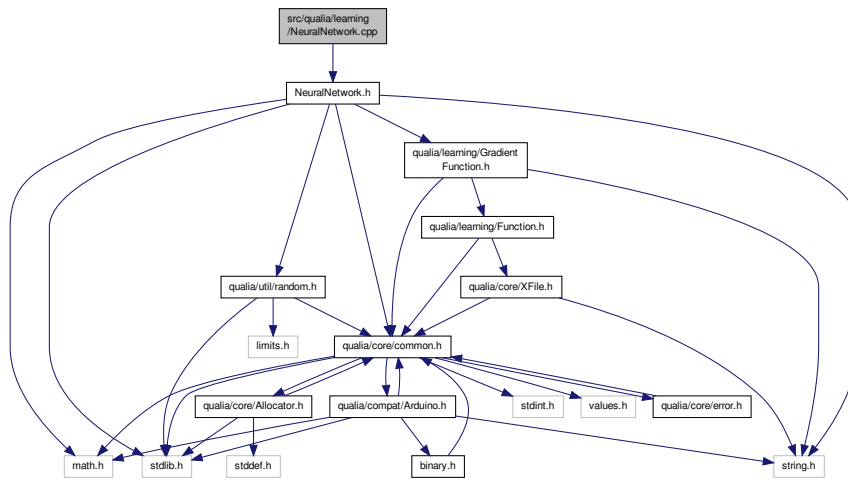
Classes

- class [MultiDataSet](#)

6.64 src/qualia/learning/NeuralNetwork.cpp File Reference

```
#include "NeuralNetwork.h"
```

Include dependency graph for NeuralNetwork.cpp:



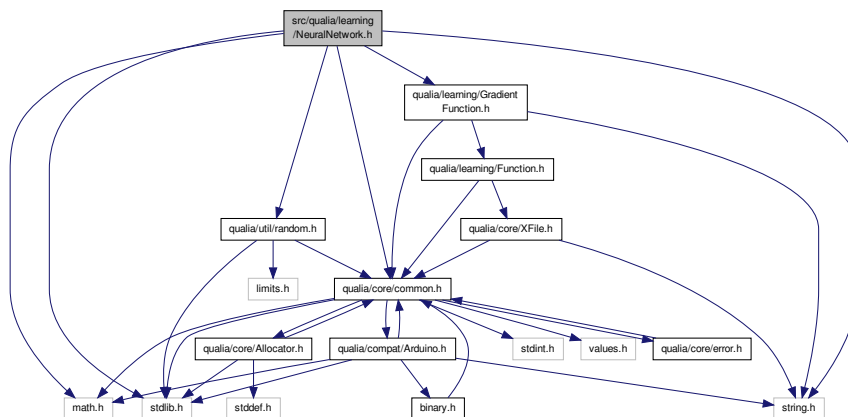
6.65 src/qualia/learning/NeuralNetwork.h File Reference

```

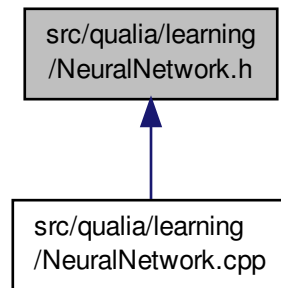
#include <stdlib.h>
#include <math.h>
#include <string.h>
#include <qualia/learning/GradientFunction.h>
#include <qualia/core/common.h>
#include <qualia/util/random.h>

```

Include dependency graph for NeuralNetwork.h:



This graph shows which files directly or indirectly include this file:



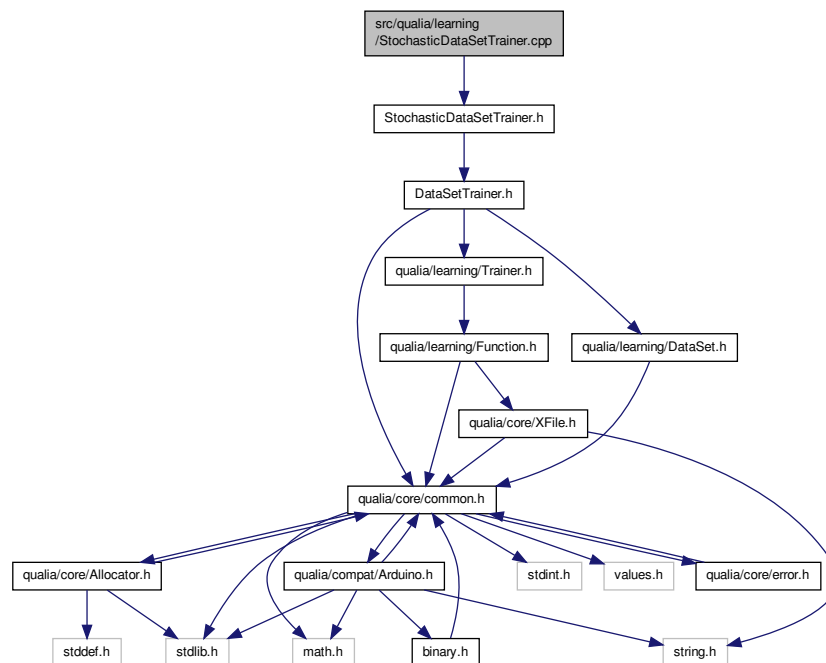
Classes

- class [NeuralNetwork](#)
- struct [NeuralNetwork::Layer](#)
[Layer](#) structure.

6.66 src/qualia/learning/StochasticDataSetTrainer.cpp File Reference

```
#include "StochasticDataSetTrainer.h"
```

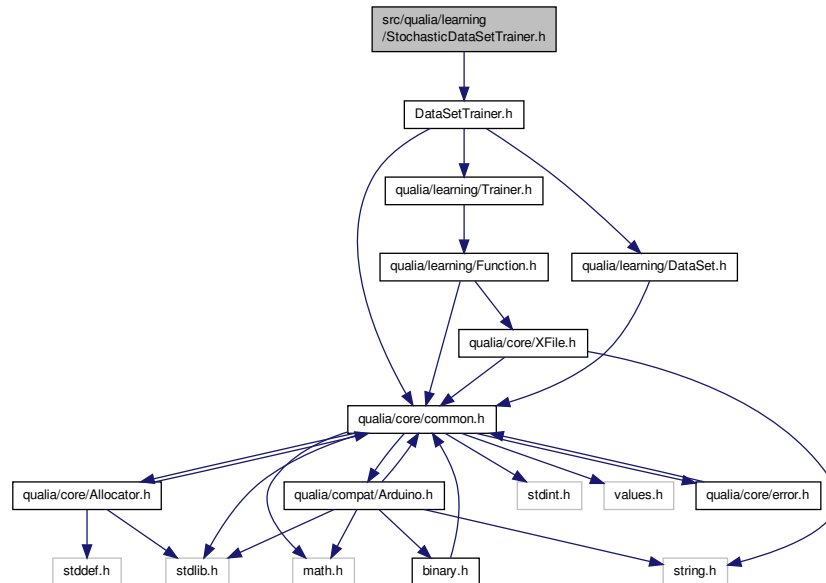
Include dependency graph for StochasticDataSetTrainer.cpp:



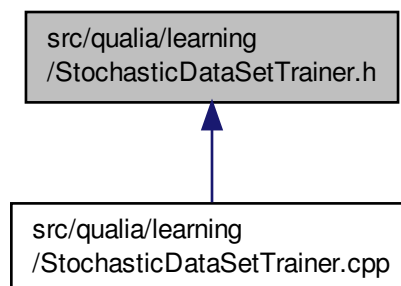
6.67 src/qualia/learning/StochasticDataSetTrainer.h File Reference

```
#include "DataSetTrainer.h"
```

Include dependency graph for StochasticDataSetTrainer.h:



This graph shows which files directly or indirectly include this file:



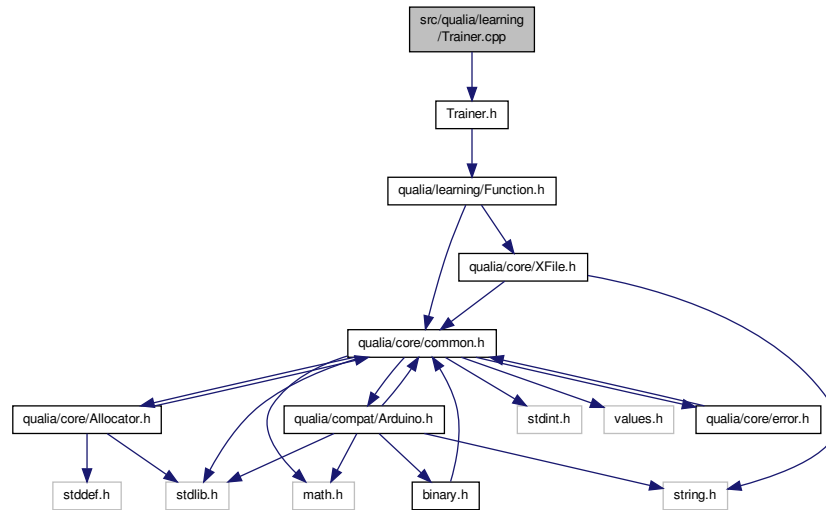
Classes

- class [StochasticDataSetTrainer](#)

6.68 src/qualia/learning/Trainer.cpp File Reference

```
#include "Trainer.h"
```

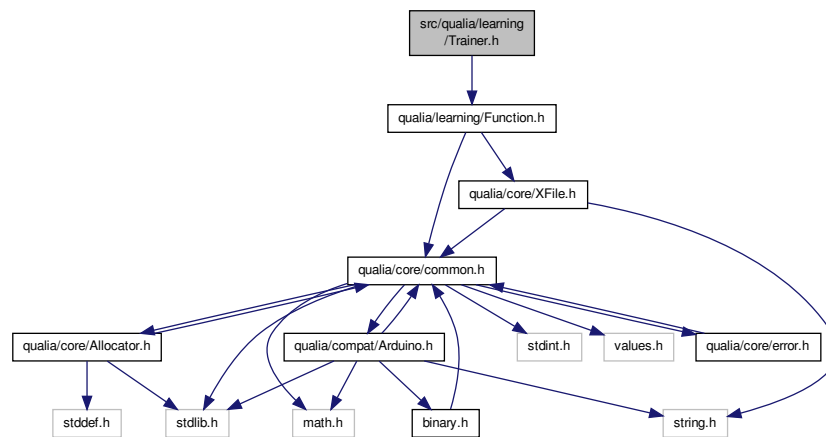
Include dependency graph for Trainer.cpp:



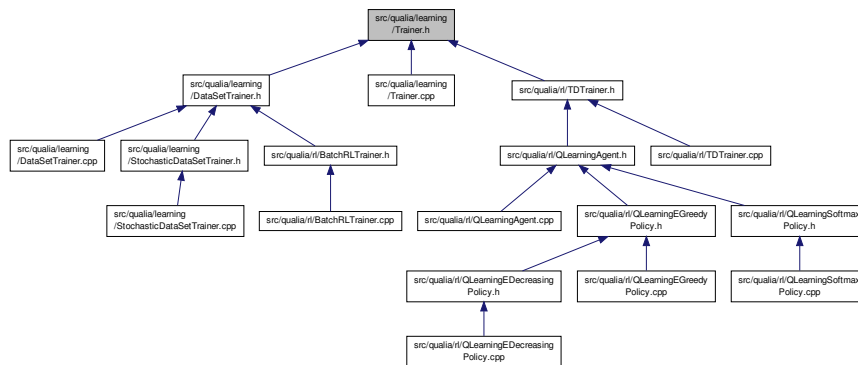
6.69 src/qualia/learning/Trainer.h File Reference

```
#include <qualia/learning/Function.h>
```

Include dependency graph for Trainer.h:



This graph shows which files directly or indirectly include this file:



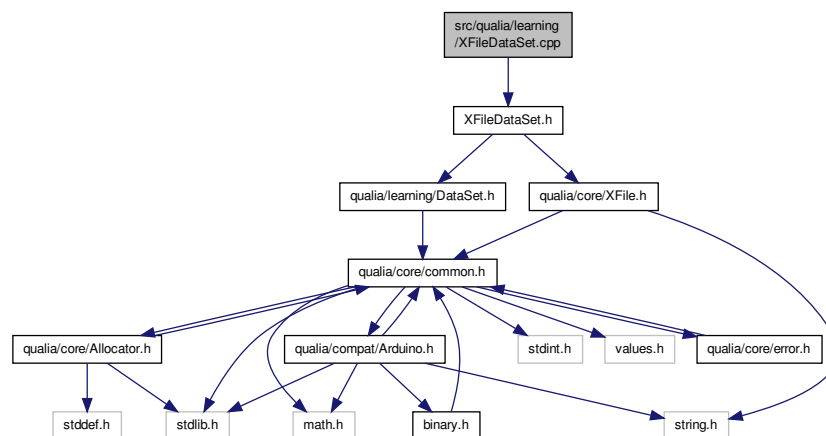
Classes

- class [Trainer](#)

6.70 src/qualia/learning/XFileDataSet.cpp File Reference

```
#include "XFileDataSet.h"
```

Include dependency graph for XFileDataSet.cpp:

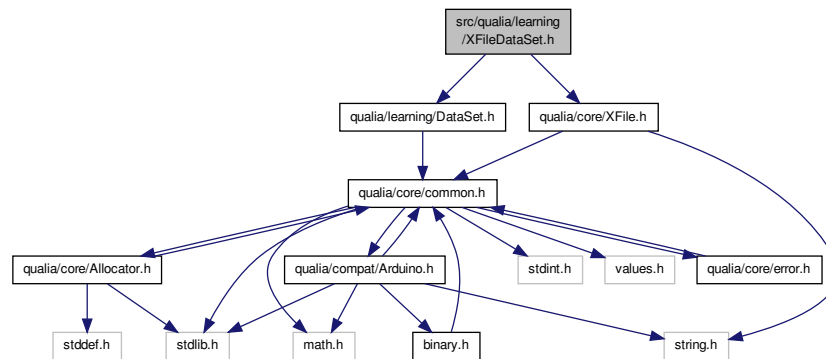


6.71 src/qualia/learning/XFileDataSet.h File Reference

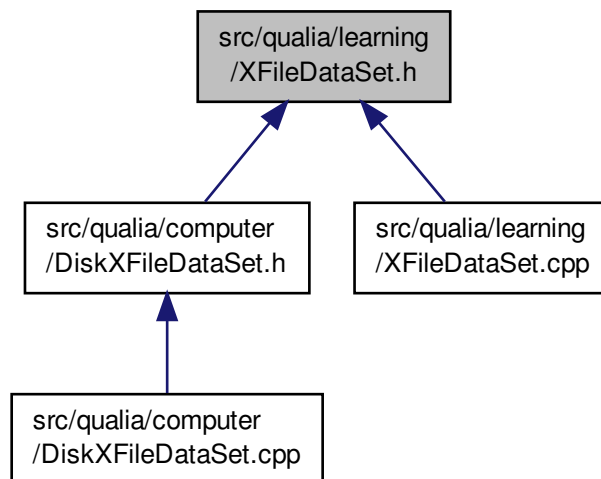
```
#include <qualia/learning/DataSet.h>
```

```
#include <qualia/core/XFile.h>
```

Include dependency graph for XFileDataSet.h:



This graph shows which files directly or indirectly include this file:



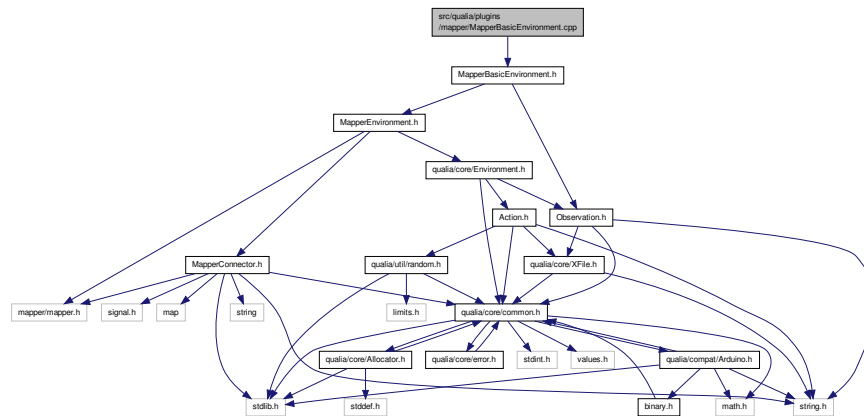
Classes

- class [XFileDataSet](#)

6.72 src/qualia/plugins/mapper/MapperBasicEnvironment.cpp File Reference

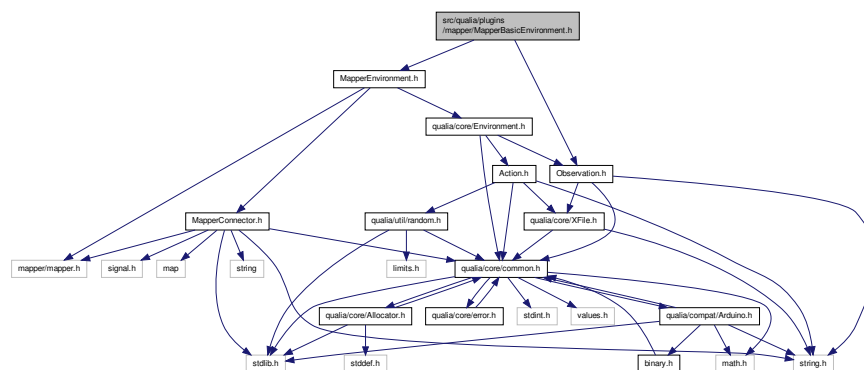
```
#include "MapperBasicEnvironment.h"
```

Include dependency graph for MapperBasicEnvironment.cpp:

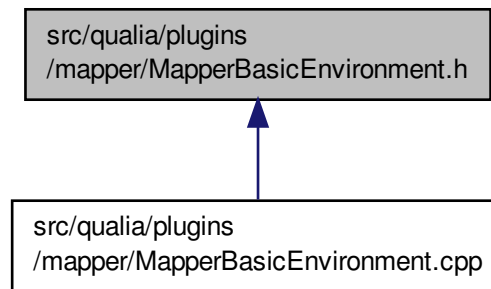


6.73 src/qualia/plugins/mapper/MapperBasicEnvironment.h File Reference

```
#include "MapperEnvironment.h"
#include <qualia/core/Observation.h>
Include dependency graph for MapperBasicEnvironment.h:
```



This graph shows which files directly or indirectly include this file:



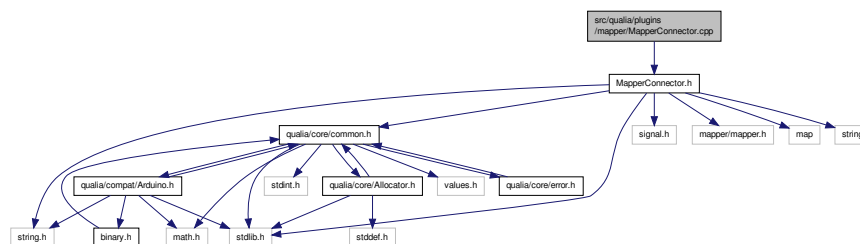
Classes

- class [MapperBasicEnvironment](#)

6.74 src/qualia/plugins/mapper/MapperConnector.cpp File Reference

```
#include "MapperConnector.h"
```

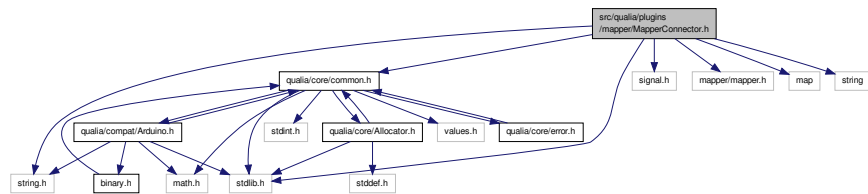
Include dependency graph for `MapperConnector.cpp`:



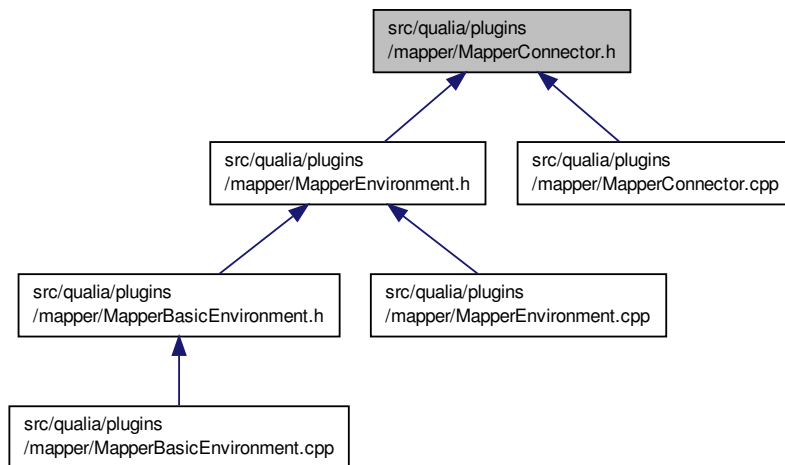
6.75 src/qualia/plugins/mapper/MapperConnector.h File Reference

```
#include <string.h>
#include <stdlib.h>
#include <signal.h>
#include <mapper/mapper.h>
#include <qualia/core/common.h>
#include <map>
#include <string>
```

Include dependency graph for MapperConnector.h:



This graph shows which files directly or indirectly include this file:



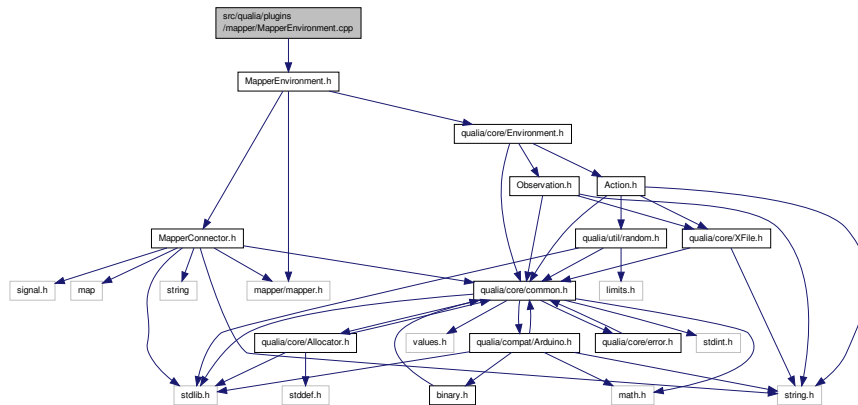
Classes

- class [MapperConnector](#)
- struct [MapperConnector::SignalData](#)

6.76 src/qualia/plugins/mapper/MapperEnvironment.cpp File Reference

```
#include "MapperEnvironment.h"
```

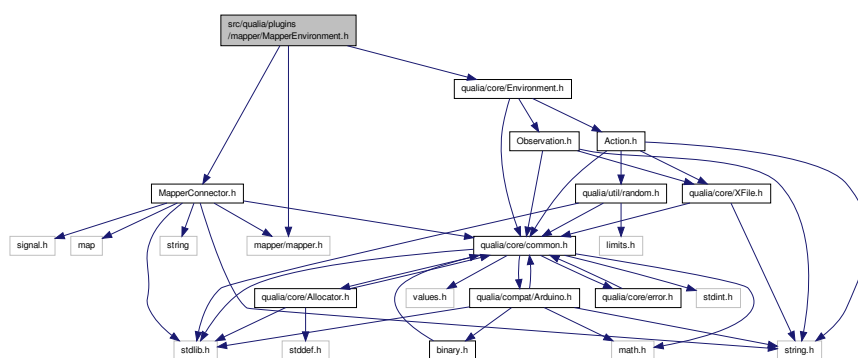

Include dependency graph for MapperEnvironment.cpp:



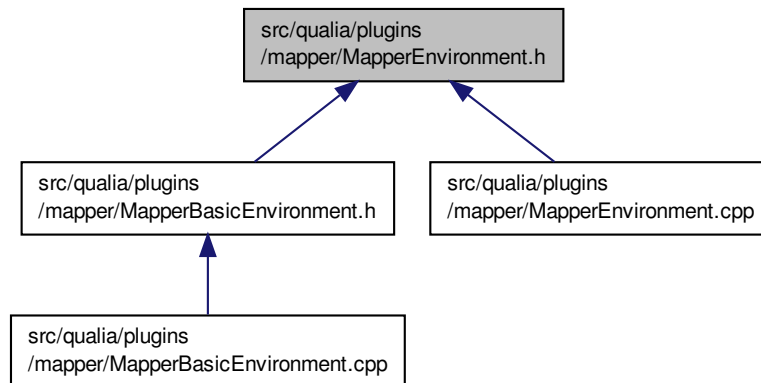
6.77 src/qualia/plugins/mapper/MapperEnvironment.h File Reference

```
#include "MapperConnector.h"
#include <qualia/core/Environment.h>
#include <mapper/mapper.h>
```

Include dependency graph for MapperEnvironment.h:



This graph shows which files directly or indirectly include this file:



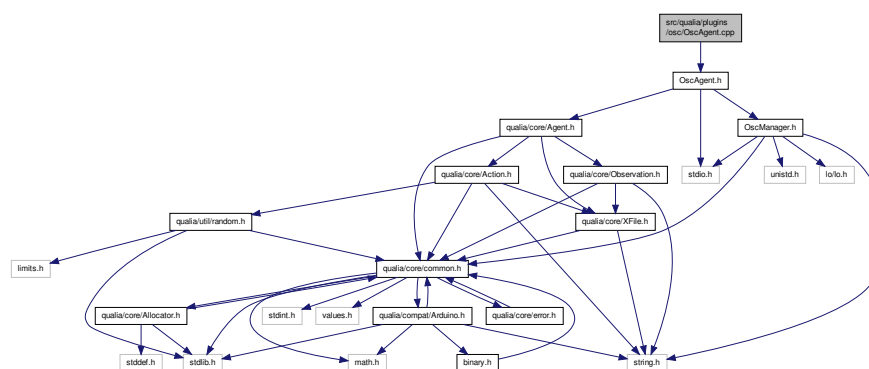
Classes

- class [MapperEnvironment](#)

6.78 src/qualia/plugins/osc/OscAgent.cpp File Reference

```
#include "OscAgent.h"
```

Include dependency graph for `OscAgent.cpp`:



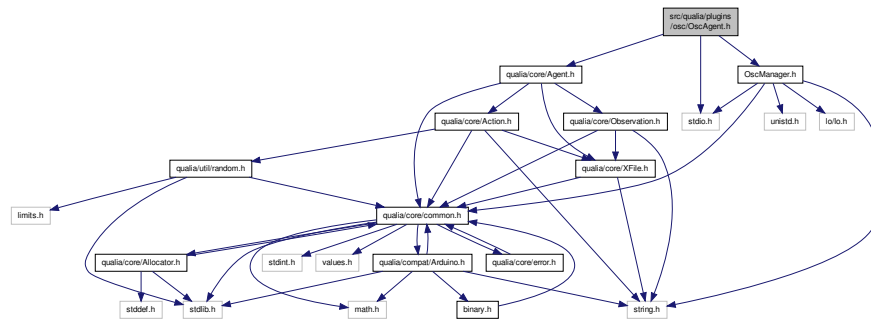
6.79 src/qualia/plugins/osc/OscAgent.h File Reference

```
#include <qualia/core/Agent.h>
```

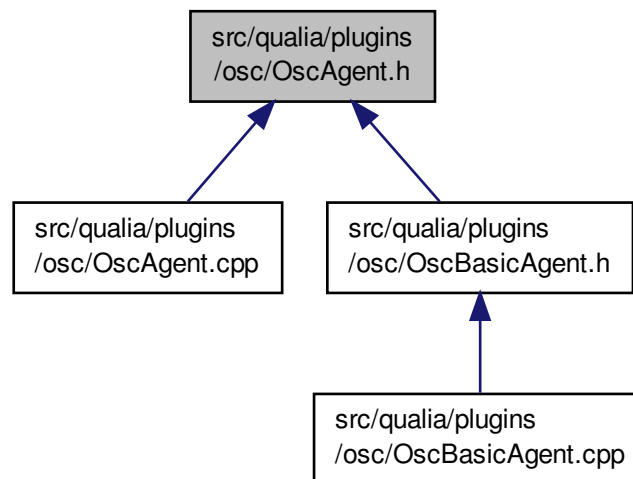
```
#include <stdio.h>
```

```
#include "OscManager.h"
```

Include dependency graph for OscAgent.h:



This graph shows which files directly or indirectly include this file:



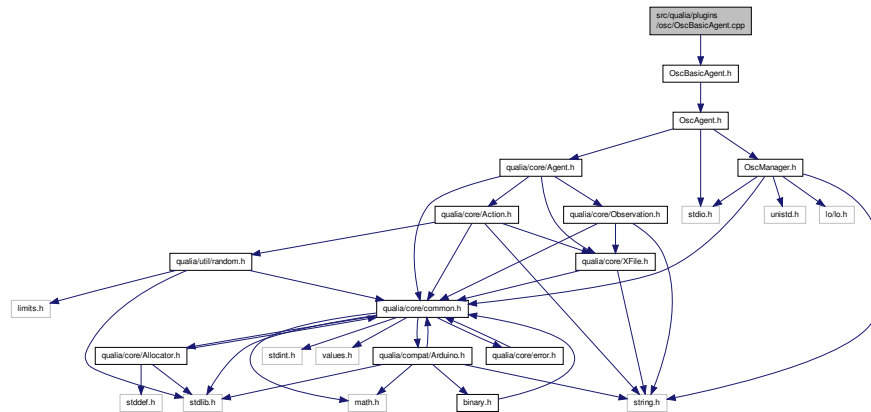
Classes

- class [OscAgent](#)

6.80 src/qualia/plugins/osc/OscBasicAgent.cpp File Reference

```
#include "OscBasicAgent.h"
```

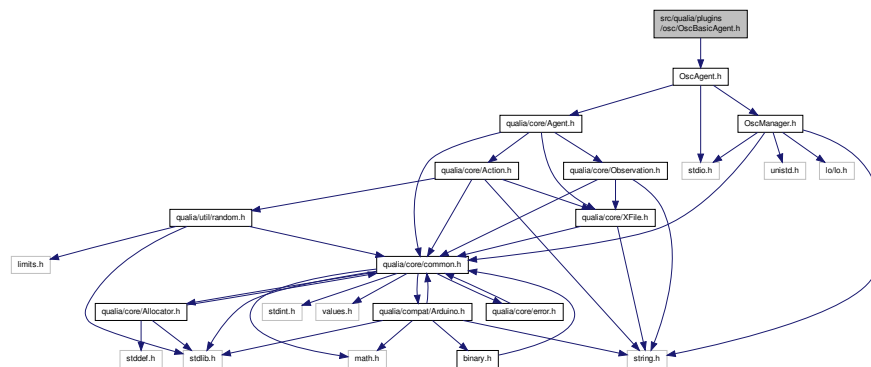
Include dependency graph for OscBasicAgent.cpp:



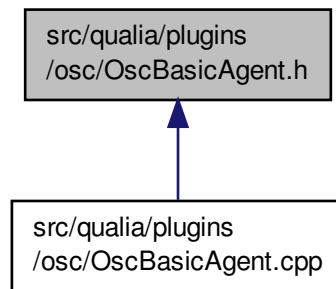
6.81 src/qualia/plugins/osc/OscBasicAgent.h File Reference

```
#include "OscAgent.h"
```

Include dependency graph for OscBasicAgent.h:



This graph shows which files directly or indirectly include this file:



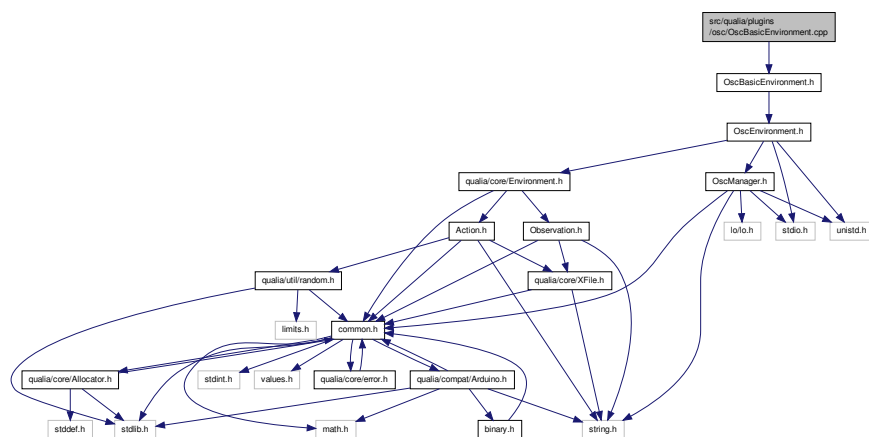
Classes

- class [OscBasicAgent](#)

6.82 src/qualia/plugins/osc/OscBasicEnvironment.cpp File Reference

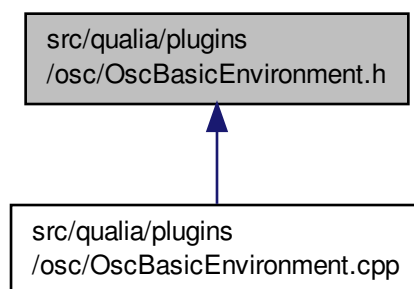
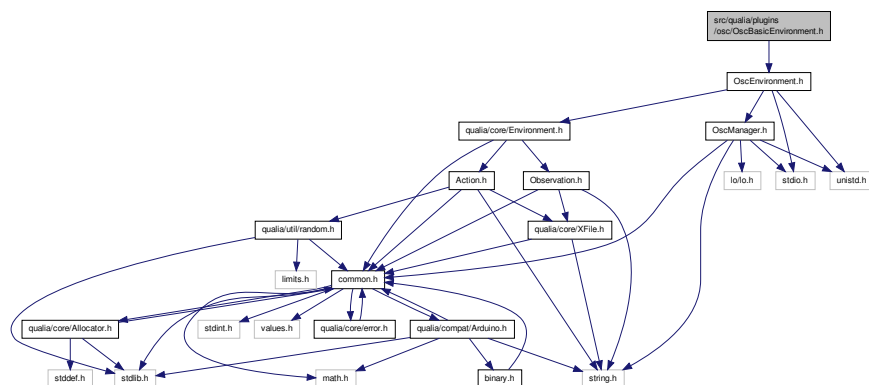
```
#include "OscBasicEnvironment.h"
```

Include dependency graph for OscBasicEnvironment.cpp:

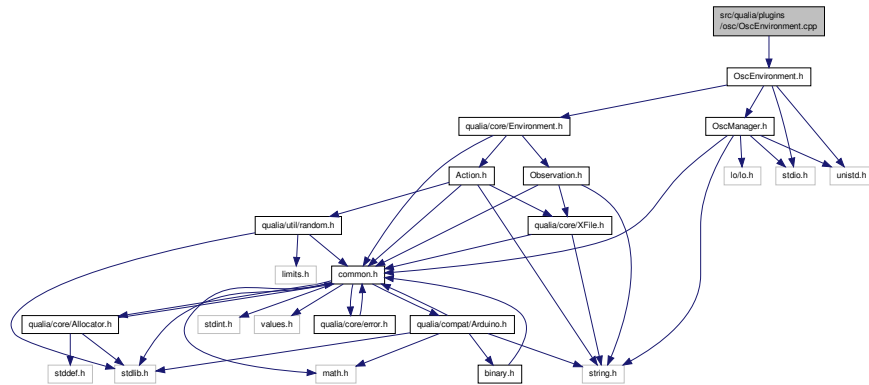


6.83 src/qualia/plugins/osc/OscBasicEnvironment.h File Reference

```
#include "OscEnvironment.h"
```



Include dependency graph for OscEnvironment.cpp:



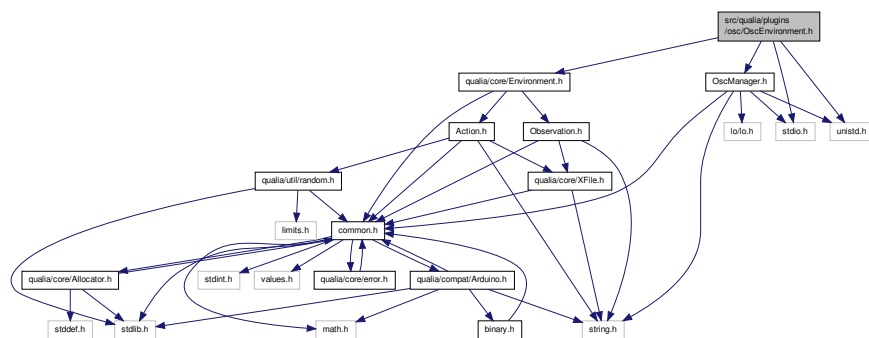
6.85 src/qualia/plugins/osc/OscEnvironment.h File Reference

```

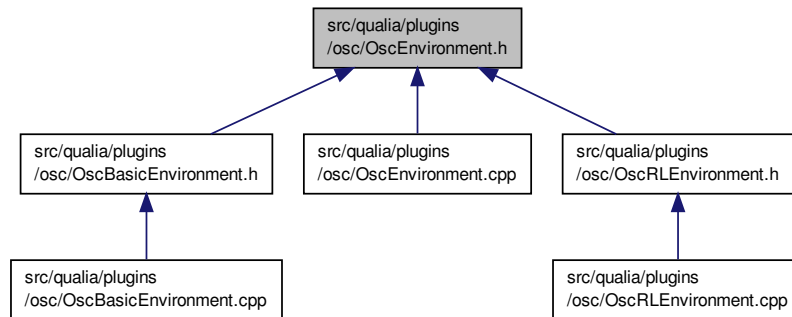
#include <qualia/core/Environment.h>
#include <stdio.h>
#include <unistd.h>
#include "OscManager.h"

```

Include dependency graph for OscEnvironment.h:



This graph shows which files directly or indirectly include this file:



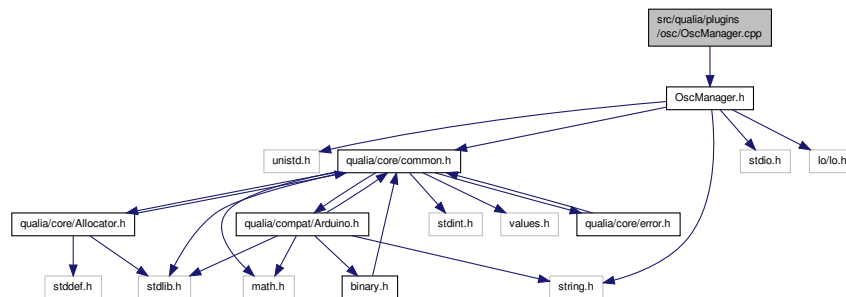
Classes

- class [OscEnvironment](#)

6.86 src/qualia/plugins/osc/OscManager.cpp File Reference

```
#include "OscManager.h"
```

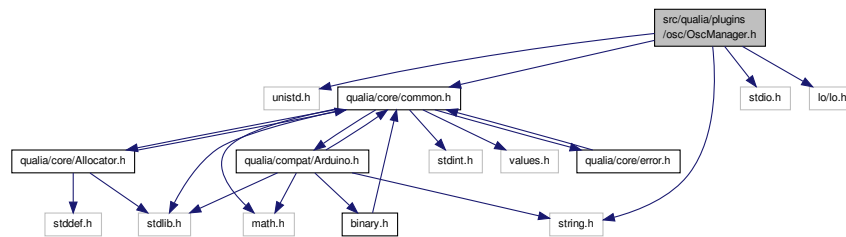
Include dependency graph for `OscManager.cpp`:



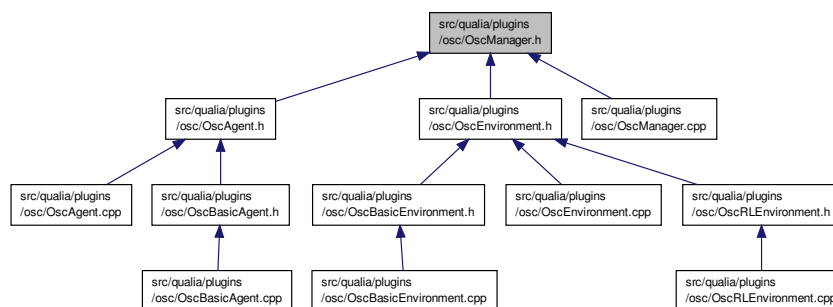
6.87 src/qualia/plugins/osc/OscManager.h File Reference

```
#include <unistd.h>
#include <qualia/core/common.h>
#include <string.h>
#include <stdio.h>
#include <lo/lo.h>
```


Include dependency graph for OscManager.h:



This graph shows which files directly or indirectly include this file:



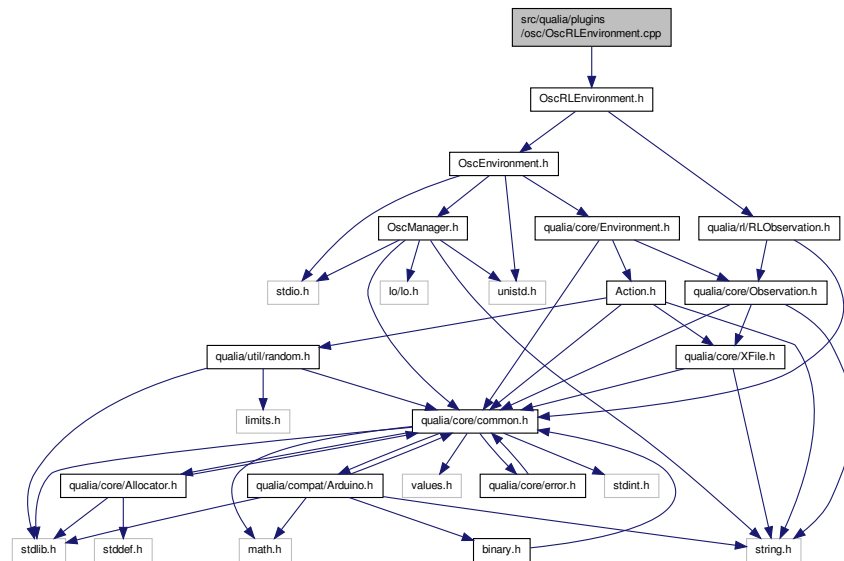
Classes

- class [OscManager](#)

6.88 src/qualia/plugins/osc/OscRLEnvironment.cpp File Reference

```
#include "OscRLEnvironment.h"
```

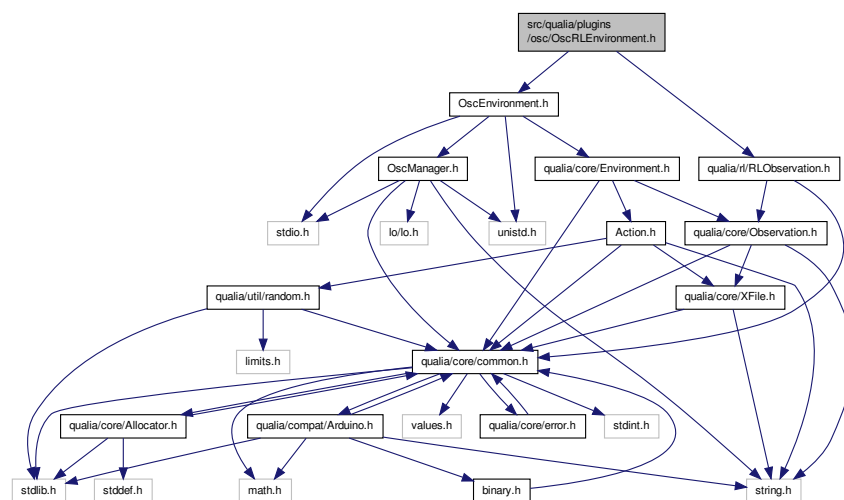
Include dependency graph for OscRLEnvironment.cpp:



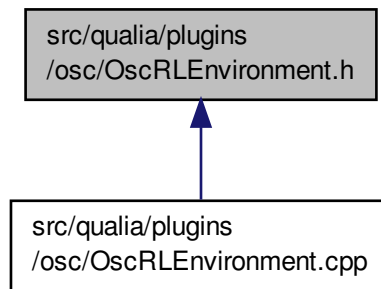
6.89 src/qualia/plugins/osc/OscRLEnvironment.h File Reference

```
#include <qualia/rl/RLObservation.h>
#include "OscEnvironment.h"
```

Include dependency graph for OscRLEnvironment.h:



This graph shows which files directly or indirectly include this file:



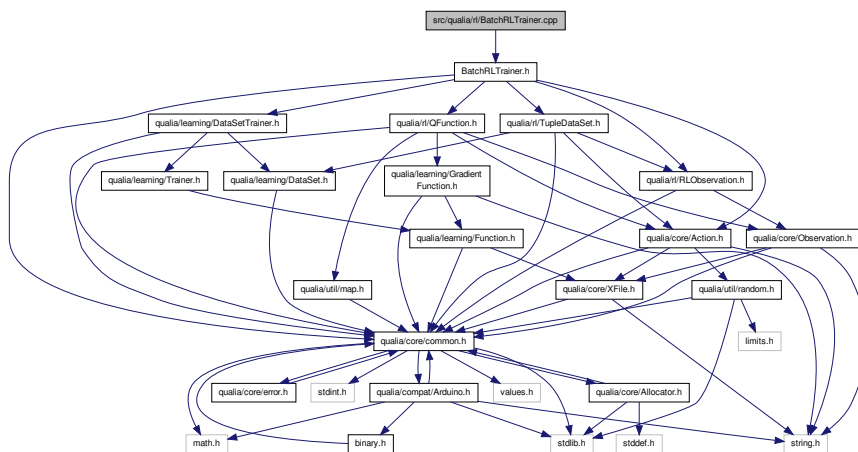
Classes

- class [OscRLEnvironment](#)

6.90 src/qualia/rl/BatchRLTrainer.cpp File Reference

```
#include "BatchRLTrainer.h"
```

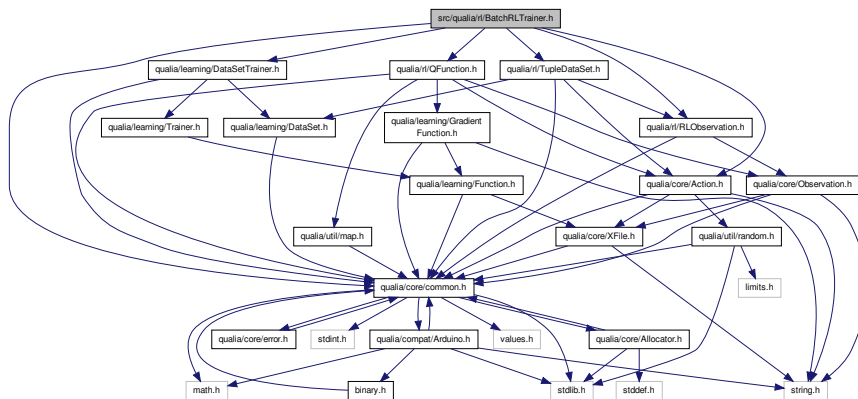
Include dependency graph for BatchRLTrainer.cpp:



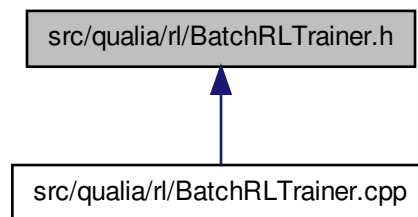
6.91 src/qualia/rl/BatchRLTrainer.h File Reference

```
#include <qualia/core/common.h>
#include <qualia/core/Action.h>
#include <qualia/learning/DataSetTrainer.h>
#include <qualia/rl/RLObservation.h>
#include <qualia/rl/QFunction.h>
#include <qualia/rl/TupleDataSet.h>
```

Include dependency graph for BatchRLTrainer.h:



This graph shows which files directly or indirectly include this file:



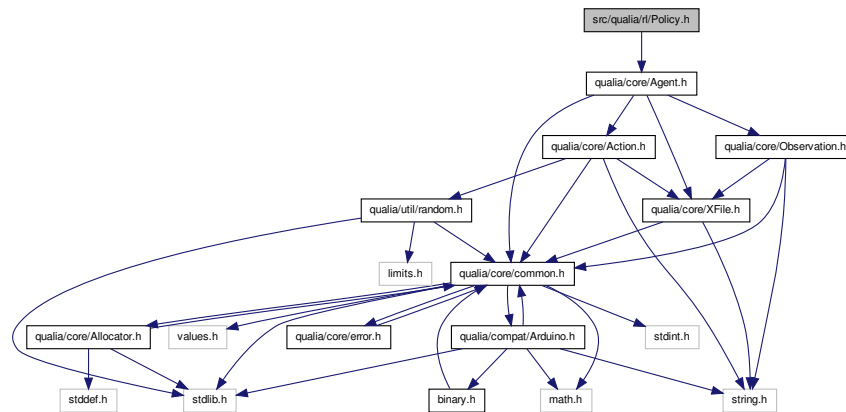
Classes

- class [BatchRLTrainer](#)

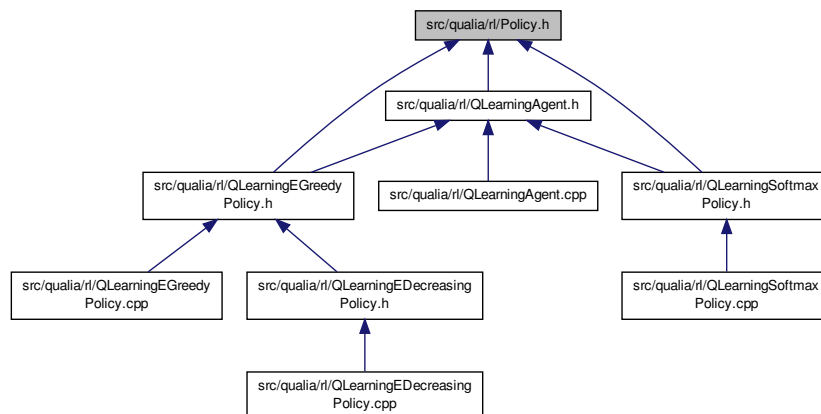
6.92 src/qualia/rl/Policy.h File Reference

```
#include <qualia/core/Agent.h>
```

Include dependency graph for Policy.h:



This graph shows which files directly or indirectly include this file:



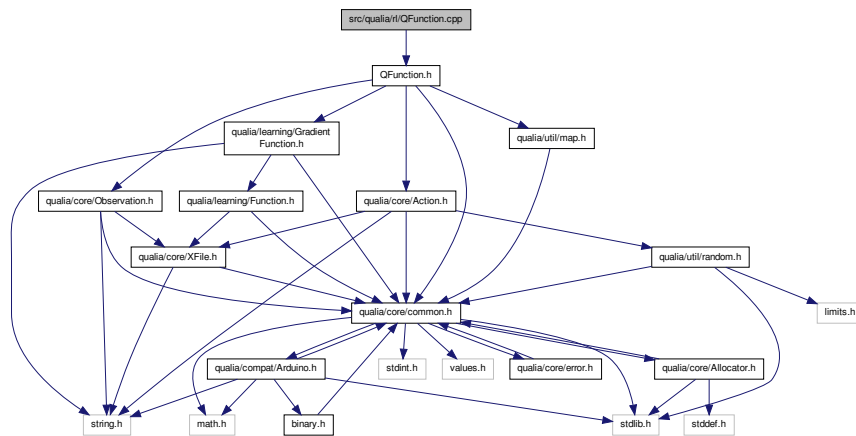
Classes

- class [Policy](#)

6.93 src/qualia/rl/QFunction.cpp File Reference

```
#include "QFunction.h"
```

Include dependency graph for QFunction.cpp:



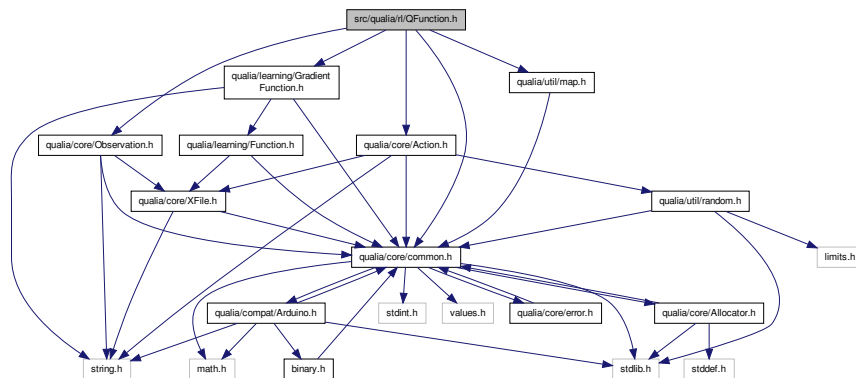
6.94 src/qualia/rl/QFunction.h File Reference

```

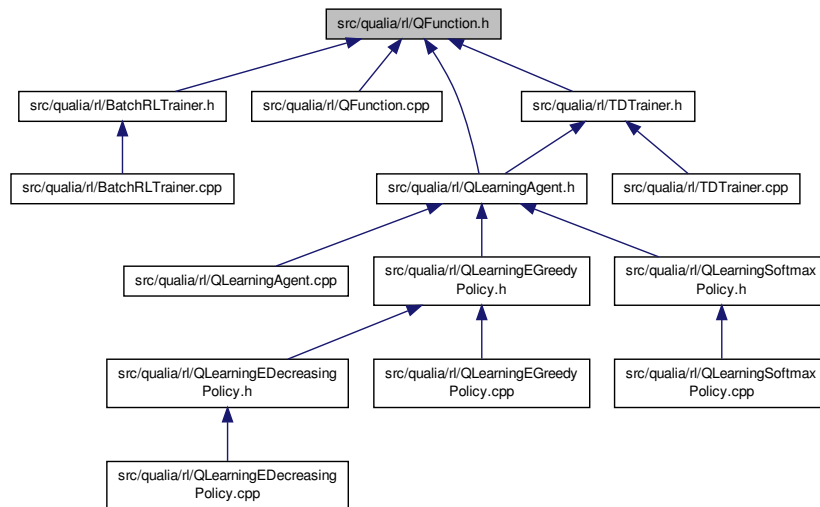
#include <qualia/core/common.h>
#include <qualia/core/Action.h>
#include <qualia/core/Observation.h>
#include <qualia/learning/GradientFunction.h>
#include <qualia/util/map.h>

```

Include dependency graph for QFunction.h:



This graph shows which files directly or indirectly include this file:



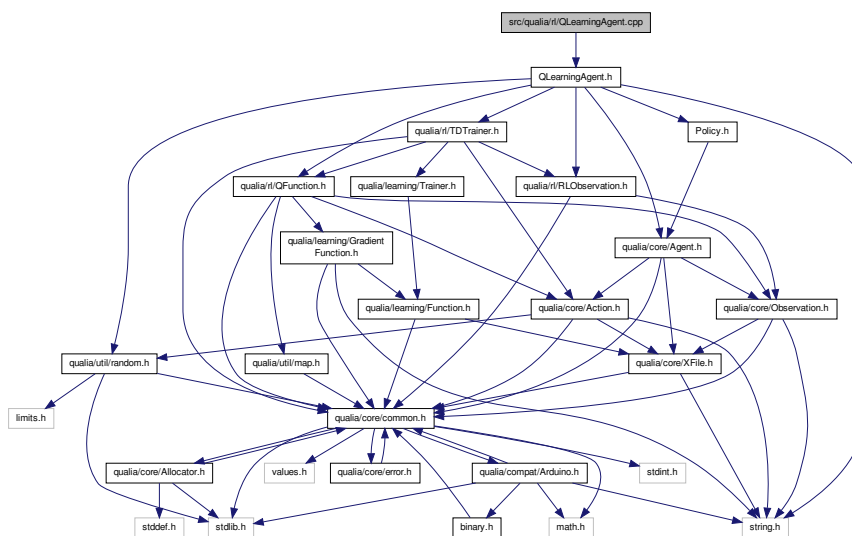
Classes

- class [QFunction](#)

6.95 src/qualia/rl/QLearningAgent.cpp File Reference

```
#include "QLearningAgent.h"
```

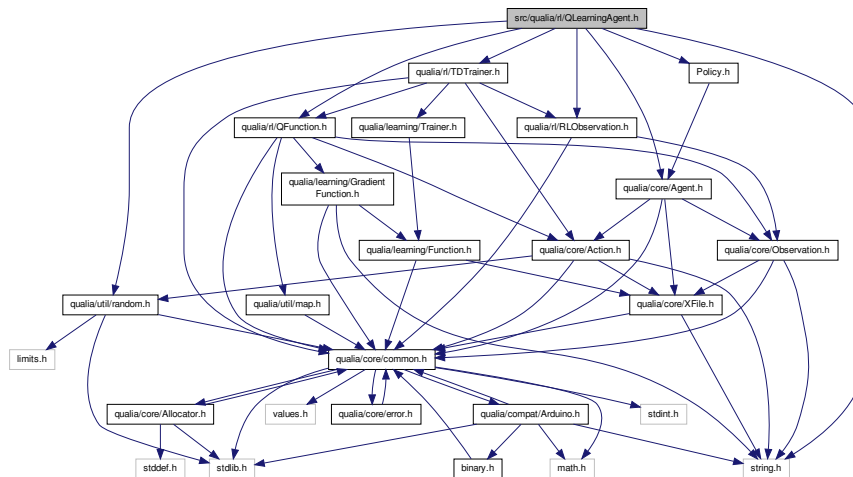
Include dependency graph for QLearningAgent.cpp:



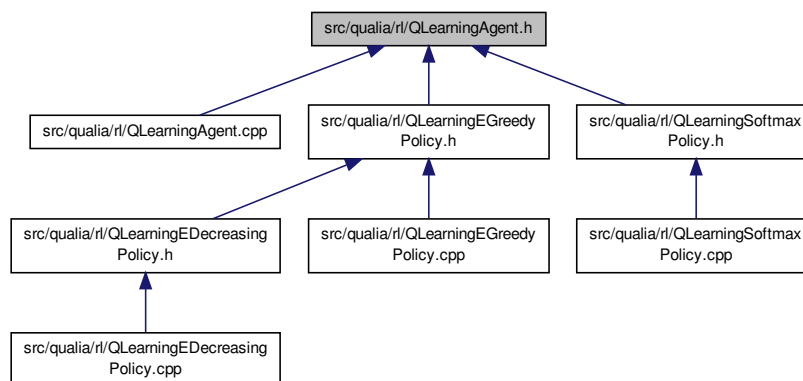
6.96 src/qualia/rl/QLearningAgent.h File Reference

```
#include <qualia/core/Agent.h>
#include <qualia/util/random.h>
#include <qualia/rl/QFunction.h>
#include <qualia/rl/TDTrainer.h>
#include "RLObservation.h"
#include "Policy.h"
#include <string.h>
```

Include dependency graph for QLearningAgent.h:



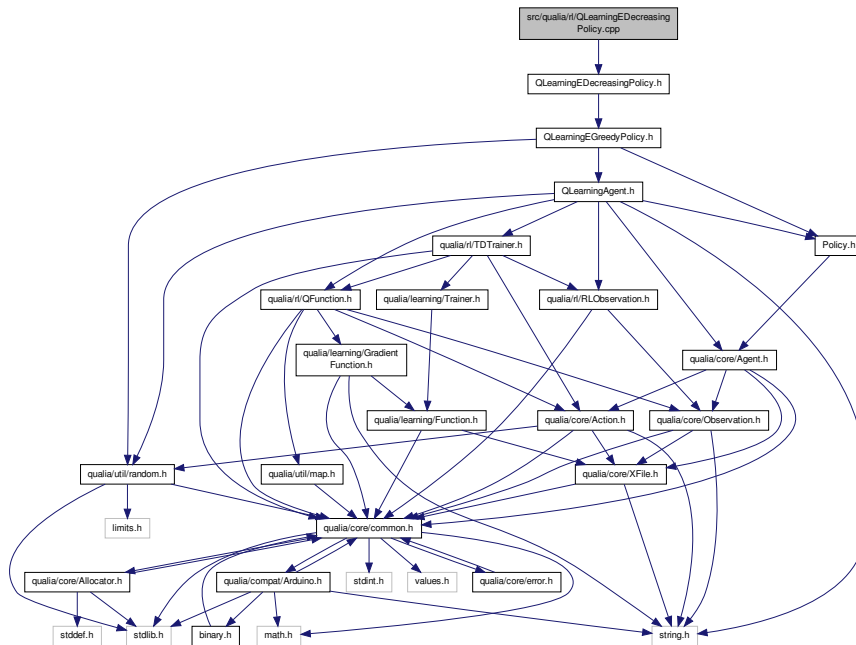
This graph shows which files directly or indirectly include this file:



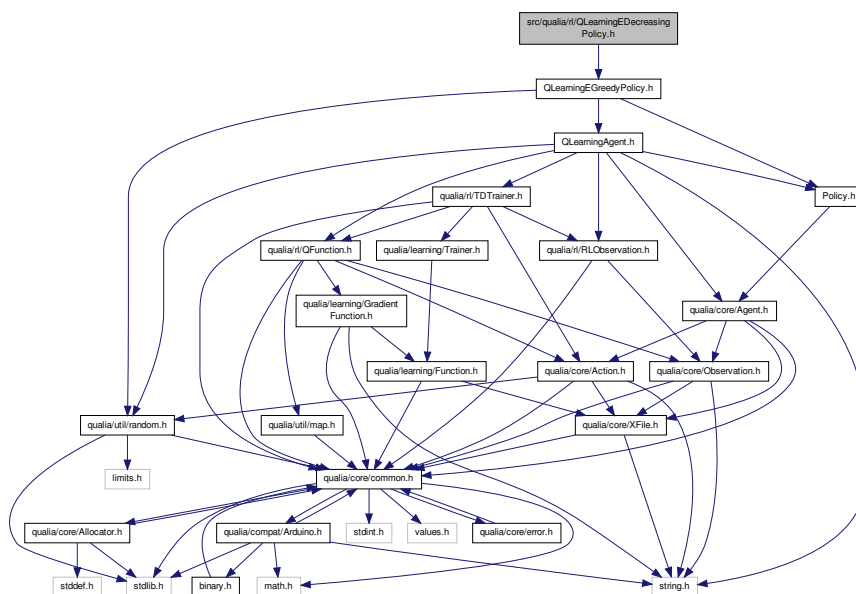
Classes

- class [QLearningAgent](#)

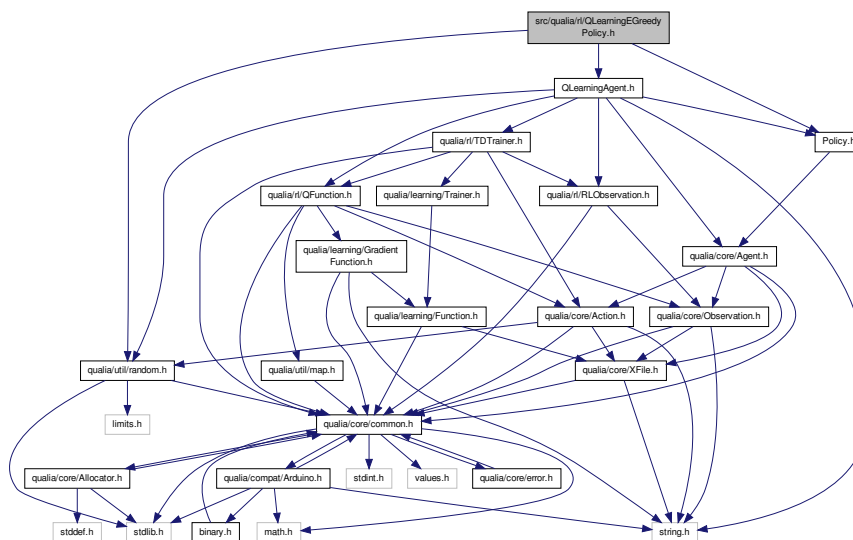

```
#include "QLearningEDecreasingPolicy.h"
Include dependency graph for QLearningEDecreasingPolicy.cpp:
```



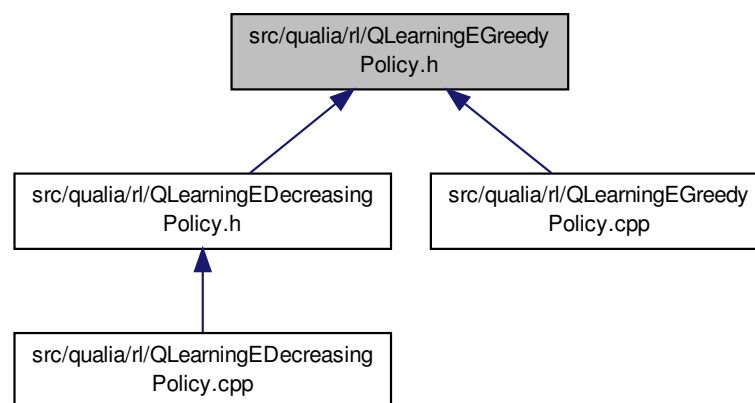
```
#include "QLearningEGreedyPolicy.h"
Include dependency graph for QLearningEDecreasingPolicy.h:
```




```
#include "Policy.h"
#include "QLearningAgent.h"
Include dependency graph for QLearningEGreedyPolicy.h:
```



This graph shows which files directly or indirectly include this file:



Classes

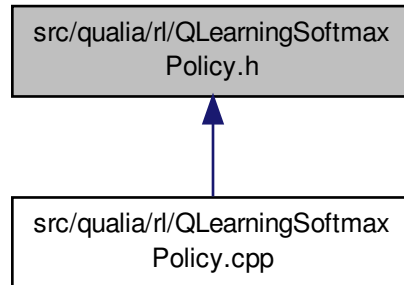
- class `QLearningEGreedyPolicy`

6.101 src/qualia/rl/QLearningSoftmaxPolicy.cpp File Reference

```
#include "QLearningSoftmaxPolicy.h"
#include <qualia/util/random.h>
```

intra-day dependency, graph is a learning algorithm. One can

This graph shows which files directly or indirectly include this file:



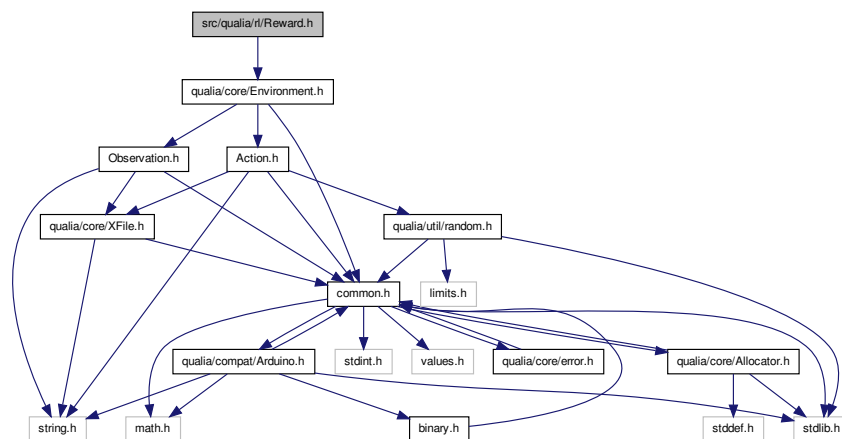
Classes

- class [QLearningSoftmaxPolicy](#)

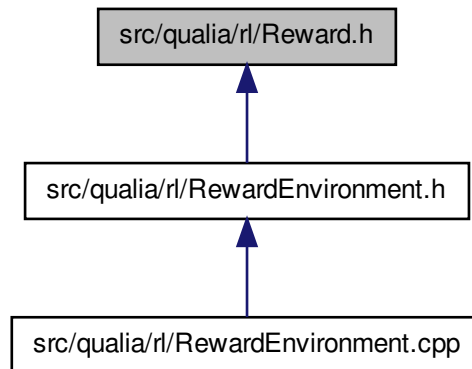
6.103 src/qualia/rl/Reward.h File Reference

```
#include <qualia/core/Environment.h>
```

Include dependency graph for Reward.h:



This graph shows which files directly or indirectly include this file:



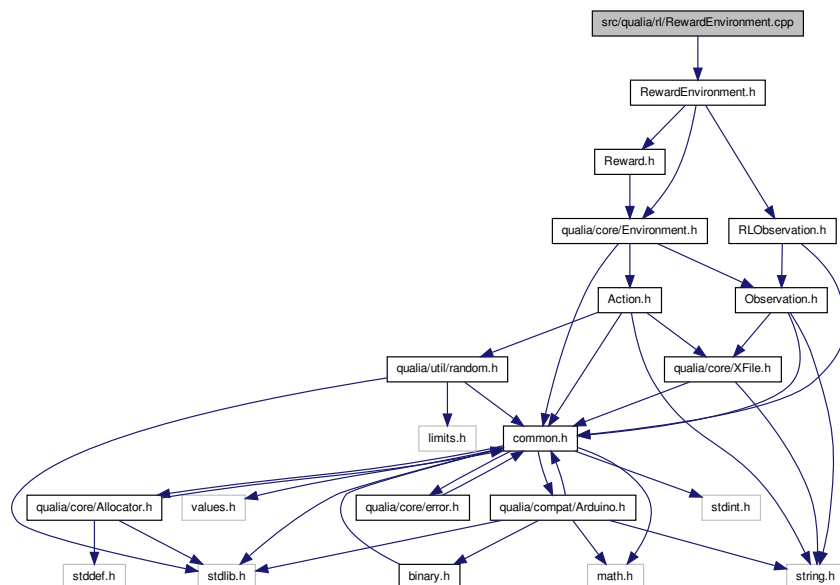
Classes

- class [Reward](#)

6.104 src/qualia/rl/RewardEnvironment.cpp File Reference

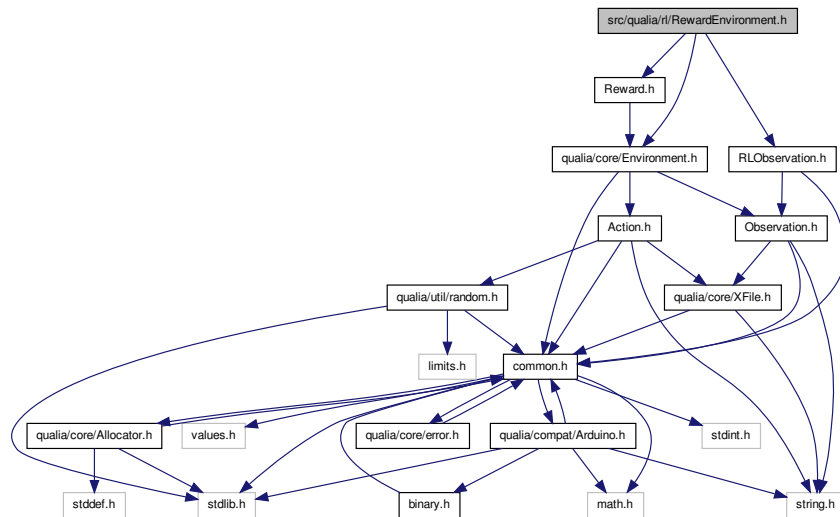
```
#include "RewardEnvironment.h"
```

Include dependency graph for RewardEnvironment.cpp:

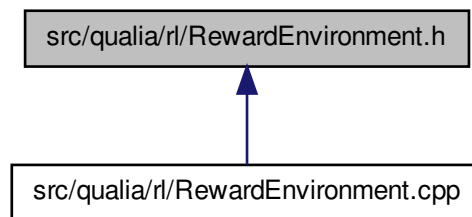


6.105 src/qualia/rl/RewardEnvironment.h File Reference

```
#include <qualia/core/Environment.h>
#include "Reward.h"
#include "RLObservation.h"
Include dependency graph for RewardEnvironment.h:
```



This graph shows which files directly or indirectly include this file:



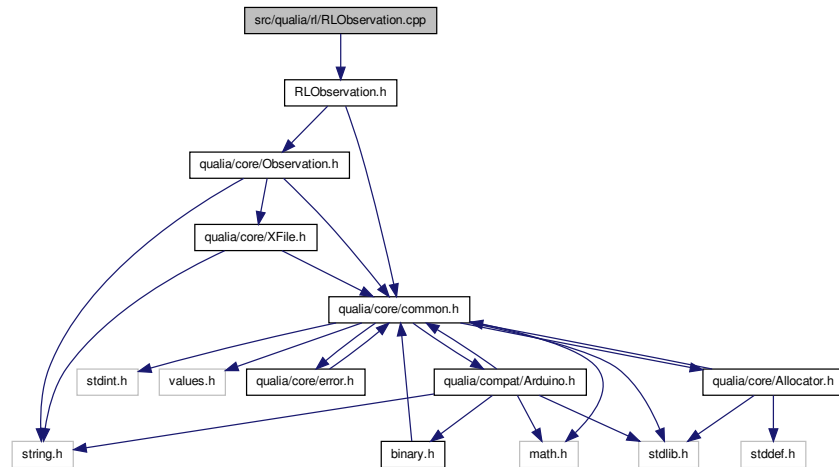
Classes

- class [RewardEnvironment](#)

6.106 src/qualia/rl/RLObservation.cpp File Reference

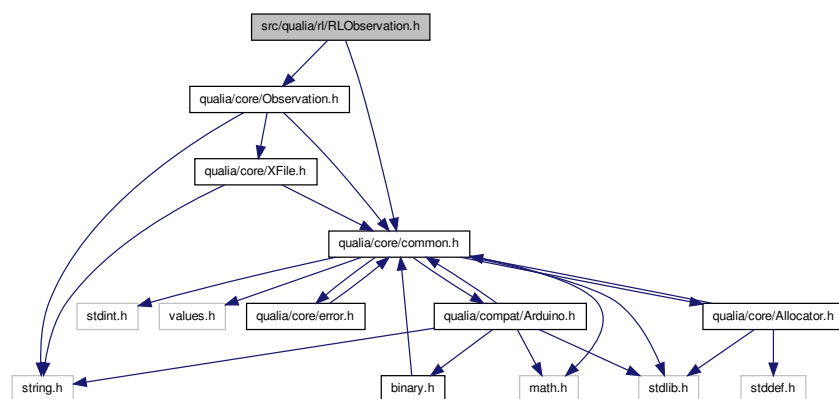
```
#include "RLObservation.h"
```

Include dependency graph for RLObservation.cpp:



6.107 src/qualia/rl/RLObservation.h File Reference

```
#include <qualia/core/common.h>
#include <qualia/core/Observation.h>
Include dependency graph for RLObservation.h:
```

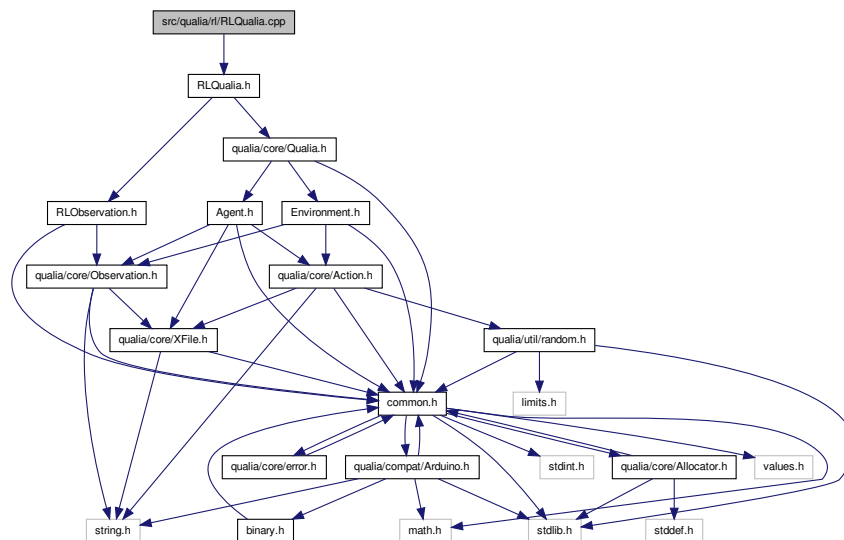


[illegible]

- class **RLObservation**

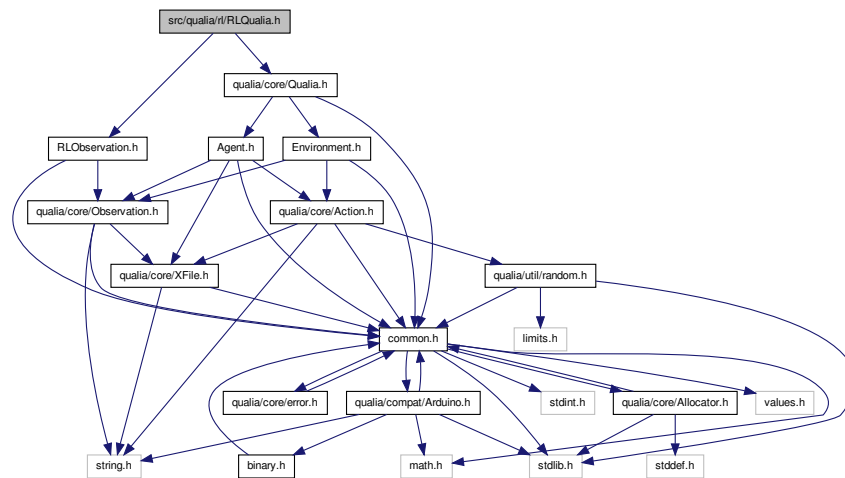
```
#include "RLQualia.h"
```

Include dependency graph for RLQualia.cpp:

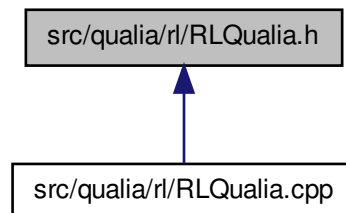


```
#include <qualia/core/Qualia.h>
#include "RLObservation.h"
```

Include dependency graph for RLQualia.h:



This graph shows which files directly or indirectly include this file:



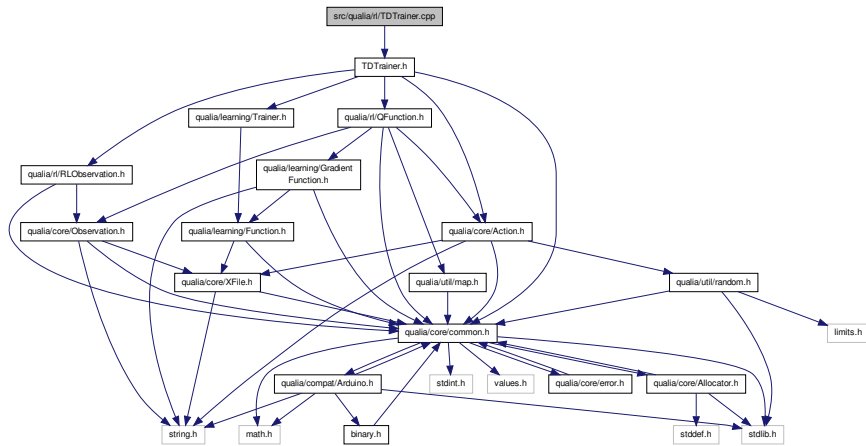
Classes

- class [RLQualia](#)

6.110 src/qualia/rl/TDTrainer.cpp File Reference

```
#include "TDTrainer.h"
```

Include dependency graph for TDTrainer.cpp:



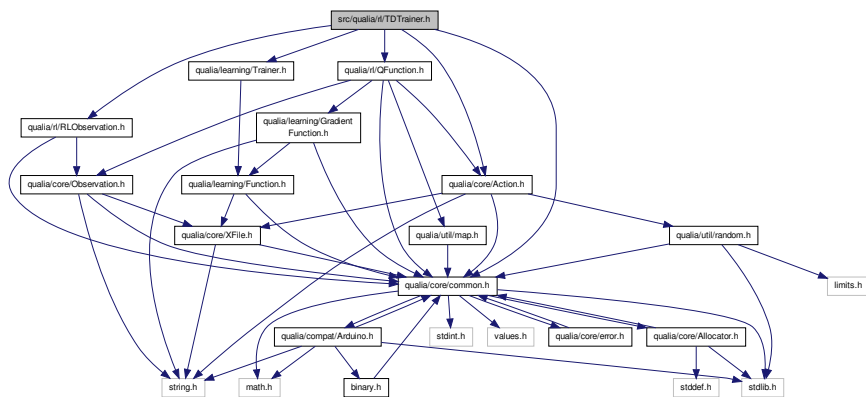
6.111 src/qualia/rl/TDTrainer.h File Reference

```

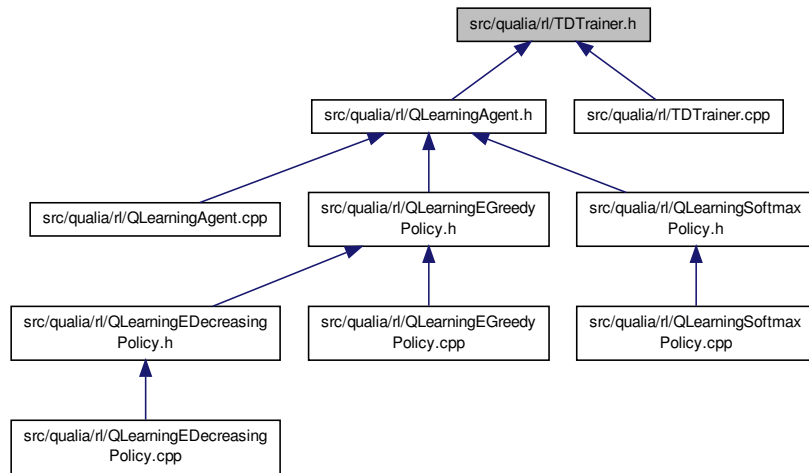
#include <qualia/core/common.h>
#include <qualia/core/Action.h>
#include <qualia/learning/Trainer.h>
#include <qualia/rl/QFunction.h>
#include <qualia/rl/RLObservation.h>

```

Include dependency graph for TDTrainer.h:



This graph shows which files directly or indirectly include this file:



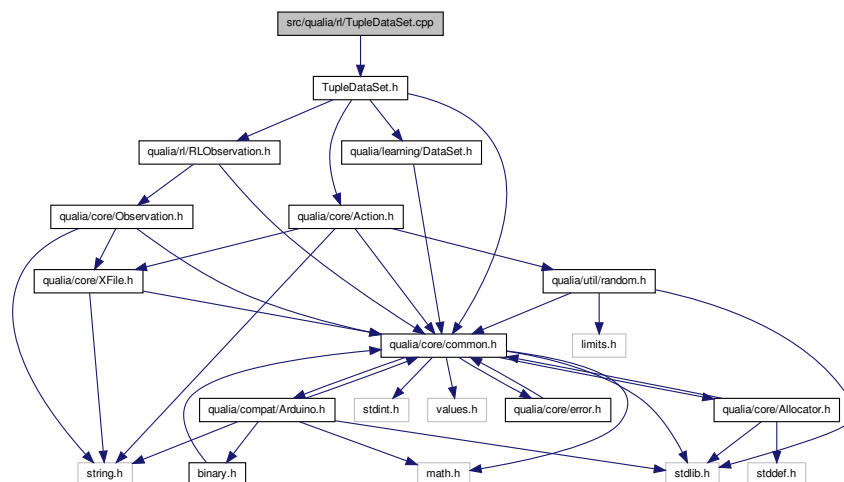
Classes

- class [TDTrainer](#)

6.112 src/qualia/rl/TupleDataSet.cpp File Reference

```
#include "TupleDataSet.h"
```

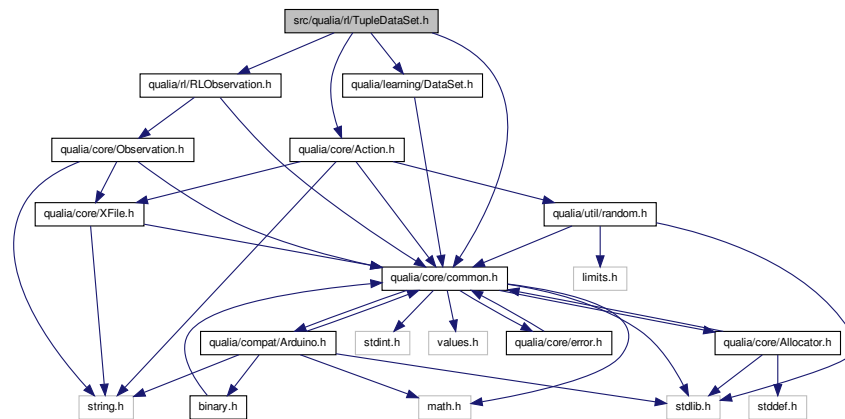
Include dependency graph for TupleDataSet.cpp:



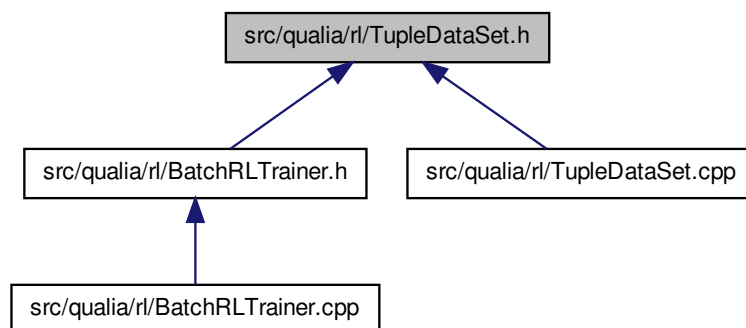
6.113 src/qualia/rl/TupleDataSet.h File Reference

```
#include <qualia/core/common.h>
```

```
#include <qualia/core/Action.h>
#include <qualia/learning/DataSet.h>
#include <qualia/rl/RLObservation.h>
Include dependency graph for TupleDataSet.h:
```



This graph shows which files directly or indirectly include this file:



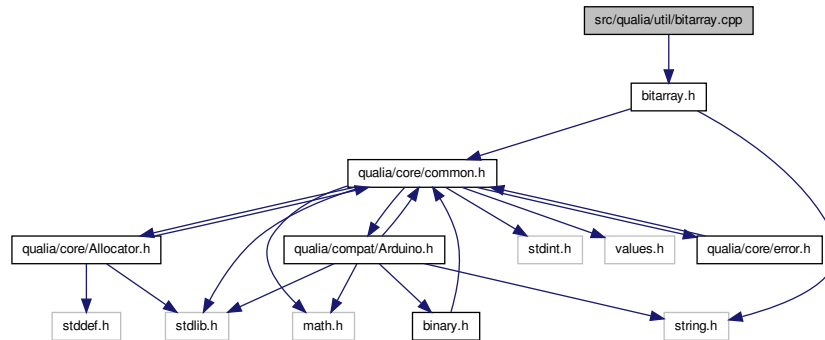
Classes

- class TupleDataSet

6.114 src/qualia/util/bitarray.cpp File Reference

```
#include "bitarray.h"
```

Include dependency graph for bitarray.cpp:



Functions

- void [arrayBitWrite](#) (uint8_t *dst, int pos, uint8_t bitValue)
- void [arrayBitSet](#) (uint8_t *dst, int pos)
- void [arrayBitClear](#) (uint8_t *dst, int pos)
- void [arrayBitFlip](#) (uint8_t *src, int pos)
- uint8_t [arrayBitRead](#) (const uint8_t *src, int pos)
- void [arrayBlockWrite](#) (void *dst, const void *src, int dstPos, int srcPos, int length)
- void [arrayBlockCopy](#) (void *dst, const void *src, int pos, int length, int dstByteSize)
- void [initPointerAndPositionForBitOperation](#) (uint8_t **array, int *pos)

6.114.1 Function Documentation

6.114.1.1 void [arrayBitClear](#) (uint8_t * *dst*, int *pos*)

6.114.1.2 void [arrayBitFlip](#) (uint8_t * *src*, int *pos*)

6.114.1.3 uint8_t [arrayBitRead](#) (const uint8_t * *src*, int *pos*)

6.114.1.4 void [arrayBitSet](#) (uint8_t * *dst*, int *pos*)

6.114.1.5 void [arrayBitWrite](#) (uint8_t * *dst*, int *pos*, uint8_t *bitValue*)

6.114.1.6 void [arrayBlockCopy](#) (void * *dst*, const void * *src*, int *pos*, int *length*, int *dstByteSize*)

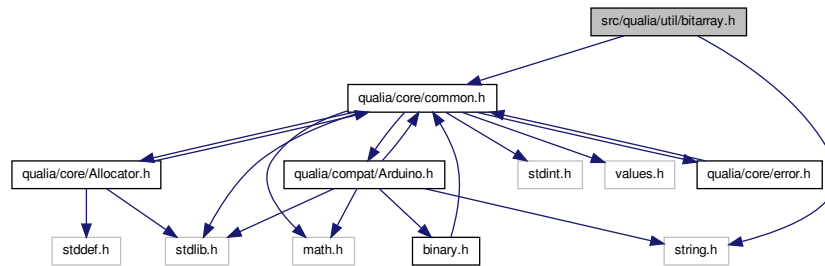
6.114.1.7 void [arrayBlockWrite](#) (void * *dst*, const void * *src*, int *dstPos*, int *srcPos*, int *length*)

6.114.1.8 void [initPointerAndPositionForBitOperation](#) (uint8_t ** *array*, int * *pos*)

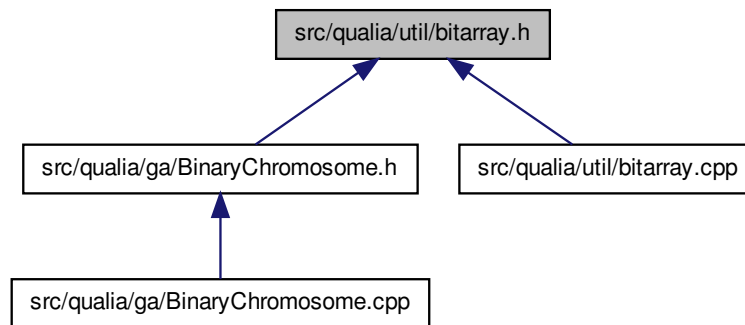
6.115 src/qualia/util/bitarray.h File Reference

```
#include <qualia/core/common.h>
#include <string.h>
```

Include dependency graph for bitarray.h:



This graph shows which files directly or indirectly include this file:



Macros

- `#define BITARRAY_BIT_TO_BYTE(bit) ((bit) >> 3)`
- `#define BITARRAY_BIT_IN_BYTE(bit) ((bit) & 7)`

Functions

- void `arrayBitWrite` (uint8_t *dst, int pos, uint8_t bitValue)
- void `arrayBitSet` (uint8_t *dst, int pos)
- void `arrayBitClear` (uint8_t *dst, int pos)
- void `arrayBitFlip` (uint8_t *dst, int pos)
- uint8_t `arrayBitRead` (const uint8_t *src, int pos)
- void `arrayBlockWrite` (void *dst, const void *src, int dstPos, int srcPos, int length)
- void `arrayBlockCopy` (void *dst, const void *src, int pos, int length, int dstByteSize)
- void `initPointerAndPositionForBitOperation` (uint8_t **array, int *pos)

6.115.1 Macro Definition Documentation

6.115.1.1 `#define BITARRAY_BIT_IN_BYTE(bit) ((bit) & 7)`

6.115.1.2 `#define BITARRAY_BIT_TO_BYTE(bit) ((bit) >> 3)`

6.115.2 Function Documentation

6.115.2.1 `void arrayBitClear (uint8_t * dst, int pos)`

6.115.2.2 `void arrayBitFlip (uint8_t * dst, int pos)`

6.115.2.3 `uint8_t arrayBitRead (const uint8_t * src, int pos)`

6.115.2.4 `void arrayBitSet (uint8_t * dst, int pos)`

6.115.2.5 `void arrayBitWrite (uint8_t * dst, int pos, uint8_t bitValue)`

6.115.2.6 `void arrayBlockCopy (void * dst, const void * src, int pos, int length, int dstByteSize)`

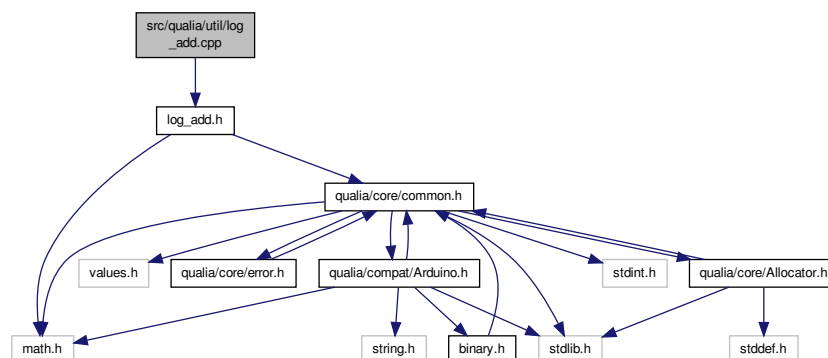
6.115.2.7 `void arrayBlockWrite (void * dst, const void * src, int dstPos, int srcPos, int length)`

6.115.2.8 `void initPointerAndPositionForBitOperation (uint8_t ** array, int * pos)`

6.116 src/qualia/util/log_add.cpp File Reference

```
#include "log_add.h"
```

Include dependency graph for log_add.cpp:



Macros

- `#define MINUS_LOG_THRESHOLD -18.42`

Functions

- `real logAdd (real log_a, real log_b)`
- `real logSub (real log_a, real log_b)`
 $\text{logSub}(\text{log_a}, \text{log_b}) = \log(a-b)$

6.116.1 Macro Definition Documentation

6.116.1.1 `#define MINUS_LOG_THRESHOLD -18.42`

6.116.2 Function Documentation

6.116.2.1 `real logAdd (real log_a, real log_b)`

`logAdd(log_a,log_b) = log(a+b) = log(exp(log_a)+exp(log_b))` but done in a smart way so that if `log_a` or `log_b` are large but not their difference the computation works correctly.

6.116.2.2 `real logSub (real log_a, real log_b)`

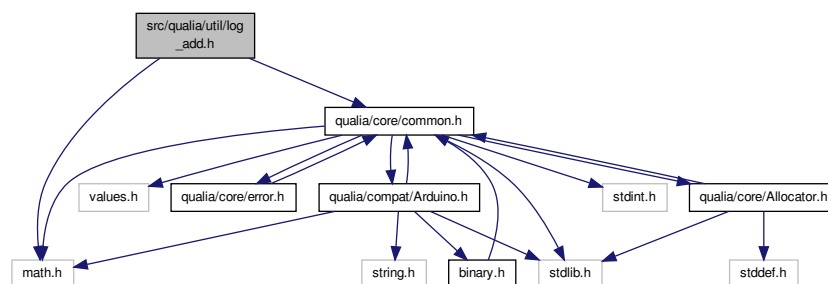
`logSub(log_a,log_b) = log(a-b)`

6.117 src/qualia/util/log_add.h File Reference

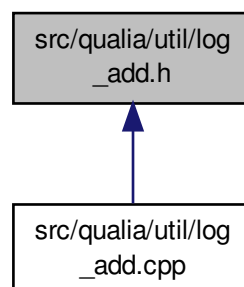
```
#include <qualia/core/common.h>
```

```
#include <math.h>
```

Include dependency graph for `log_add.h`:



This graph shows which files directly or indirectly include this file:



Macros

- `#define log1p(x) log(1+(x))`
- `#define LOG_2_PI 1.83787706640934548355`
- `#define LOG_ZERO -INF`
- `#define LOG_ONE 0`
- `real logAdd (real log_a, real log_b)`
- `real logSub (real log_a, real log_b)`

logSub(log_a,log_b) = log(a-b)

6.117.1 Macro Definition Documentation

6.117.1.1 `#define log1p(x) log(1+(x))`

6.117.1.2 `#define LOG_2_PI 1.83787706640934548355`

Some simple functions for log operations.

Author

Samy Bengio (bengio@idiap.ch)

6.117.1.3 `#define LOG_ONE 0`

6.117.1.4 `#define LOG_ZERO -INF`

6.117.2 Function Documentation

6.117.2.1 `real logAdd (real log_a, real log_b)`

`logAdd(log_a,log_b) = log(a+b) = log(exp(log_a)+exp(log_b))` but done in a smart way so that if `log_a` or `log_b` are large but not their difference the computation works correctly.

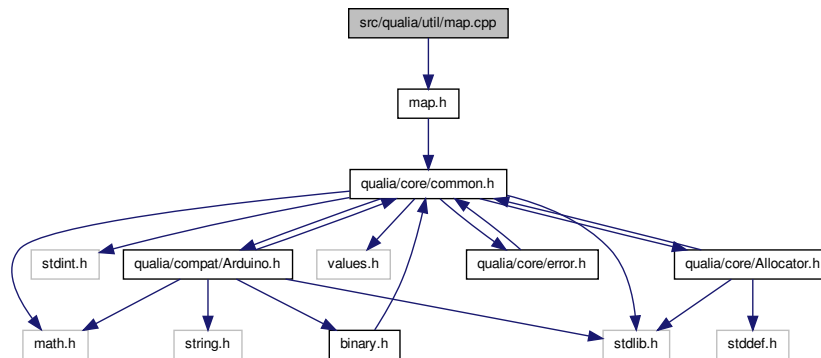
6.117.2.2 `real logSub (real log_a, real log_b)`

`logSub(log_a,log_b) = log(a-b)`

6.118 src/qualia/util/map.cpp File Reference

```
#include "map.h"
```

Include dependency graph for map.cpp:



Functions

- **real** `mapReal` (**real** `x`, **real** `in_min`, **real** `in_max`, **real** `out_min`, **real** `out_max`)

Remaps a real value in [minVal, maxVal].

6.118.1 Function Documentation

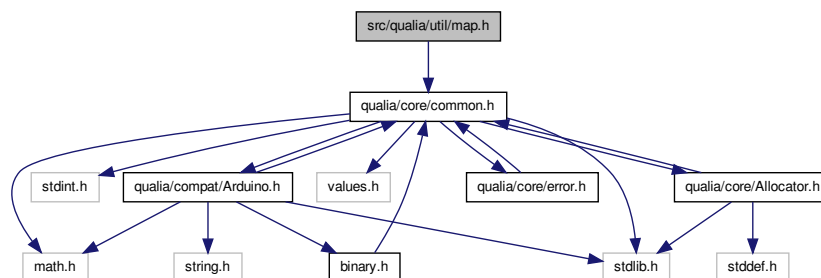
6.118.1.1 `real mapReal (real x, real in_min, real in_max, real out_min, real out_max)`

Remaps a real value in [minVal, maxVal].

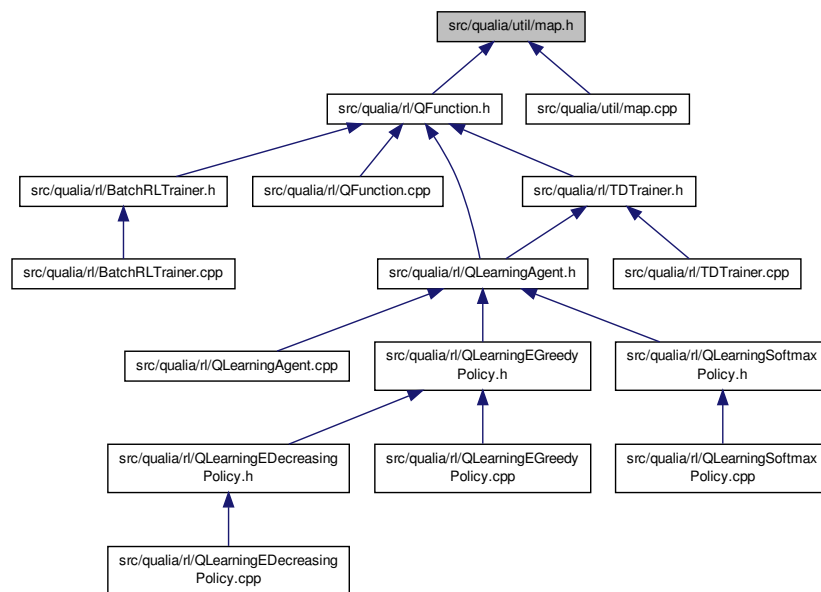
6.119 src/qualia/util/map.h File Reference

```
#include <qualia/core/common.h>
```

Include dependency graph for map.h:



This graph shows which files directly or indirectly include this file:



Functions

- [real mapReal](#) ([real](#) x, [real](#) in_min, [real](#) in_max, [real](#) out_min, [real](#) out_max)

Remaps a real value in [minVal, maxVal].

6.119.1 Function Documentation

6.119.1.1 [real mapReal](#) ([real](#) x, [real](#) in_min, [real](#) in_max, [real](#) out_min, [real](#) out_max)

Remaps a real value in [minVal, maxVal].

6.120 [src/qualia/util/MovingAverage.h](#) File Reference

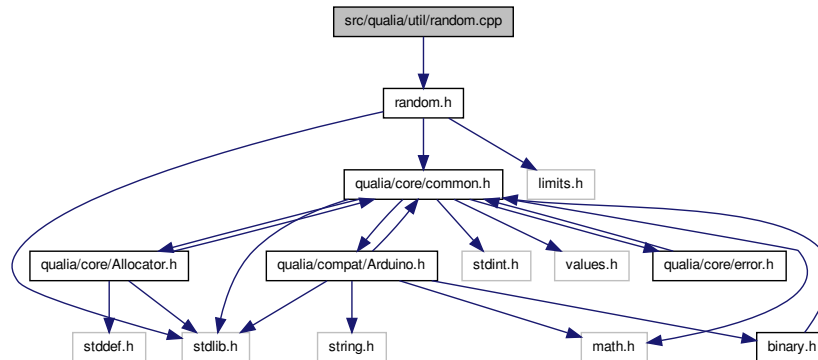
Classes

- class [MovingAverage](#)

6.121 src/qualia/util/random.cpp File Reference

```
#include "random.h"
```

Include dependency graph for random.cpp:



Functions

- float [randomUniform](#) ()
Generates a uniform random number on $[0,1[$.
- float [randomUniform](#) (float h)
Generates a uniform random number on $[0,h[$.
- float [randomUniform](#) (float a, float b)
Generates a uniform random number on $[a,b[$ ($b > a$).

6.121.1 Function Documentation

6.121.1.1 float randomUniform ()

Generates a uniform random number on $[0,1[$.

Provides methods for generating uniform random numbers.

6.121.1.2 float randomUniform (float h)

Generates a uniform random number on $[0,h[$.

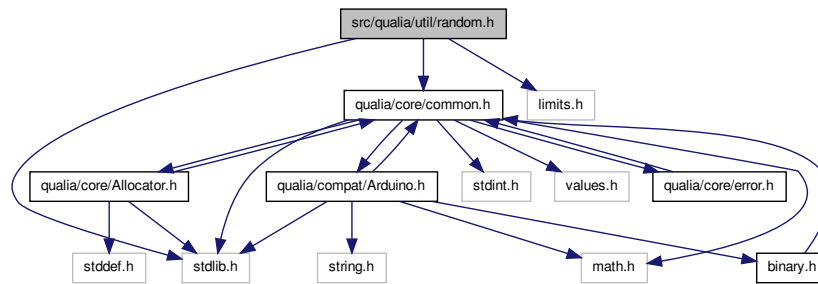
6.121.1.3 float randomUniform (float a, float b)

Generates a uniform random number on $[a,b[$ ($b > a$).

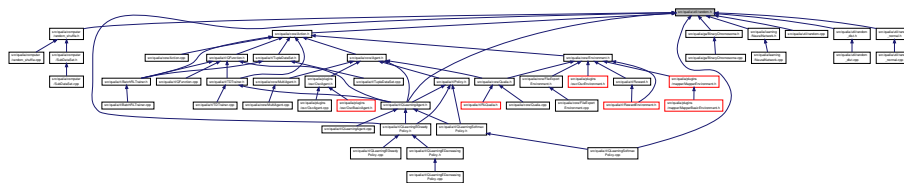
6.122 src/qualia/util/random.h File Reference

```
#include <qualia/core/common.h>
#include <limits.h>
#include <stdlib.h>
```

Include dependency graph for random.h:



This graph shows which files directly or indirectly include this file:



Functions

- float `randomUniform` ()
Generates a uniform random number on $[0,1[$.
- float `randomUniform` (float h)
Generates a uniform random number on $[0,h[$.
- float `randomUniform` (float a, float b)
Generates a uniform random number on $[a,b[$ ($b > a$).

6.122.1 Function Documentation

6.122.1.1 float randomUniform ()

Generates a uniform random number on $[0,1[$.

Provides methods for generating uniform random numbers.

6.122.1.2 float randomUniform (float h)

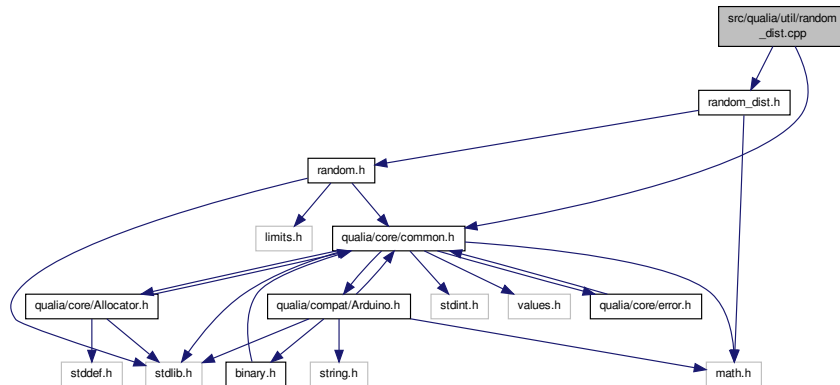
Generates a uniform random number on $[0,h[$.

6.122.1.3 float randomUniform (float a, float b)

Generates a uniform random number on $[a,b[$ ($b > a$).

6.123 src/qualia/util/random_dist.cpp File Reference

```
#include "random_dist.h"
#include <qualia/core/common.h>
Include dependency graph for random_dist.cpp:
```



Functions

- float [exponential](#) (float *lambda*)
- float [cauchy](#) (float *median*, float *sigma*)
- int [geometric](#) (float *p*)
- bool [bernouilli](#) (float *p*)

Returns true with probability p and false with probability $1-p$ ($p > 0$).

6.123.1 Function Documentation

6.123.1.1 bool [bernouilli](#) (float *p*)

Returns true with probability p and false with probability $1-p$ ($p > 0$).

6.123.1.2 float [cauchy](#) (float *median* = 0, float *sigma* = 1)

Returns a random number from a Cauchy distribution. The Cauchy density is $p(x) = \sigma / (\pi * (\sigma^2 + (x - \text{median})^2))$

6.123.1.3 float [exponential](#) (float *lambda*)

Provides methods for generating random numbers according to various probability distributions. Generates a random number from an exponential distribution. The density is $p(x) = \lambda * \exp(-\lambda * x)$, where λ is a positive number.

6.123.1.4 int [geometric](#) (float *p*)

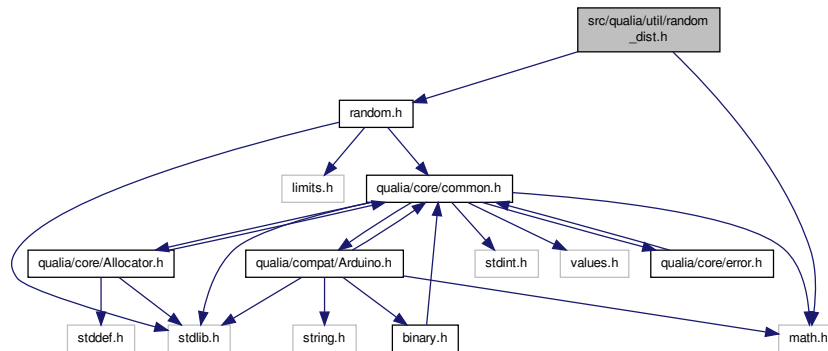
Generates a random number from a geometric distribution. It returns an integer i , where $p(i) = (1-p) * p^{(i-1)}$. p must satisfy $0 < p < 1$.

6.124 src/qualia/util/random_dist.h File Reference

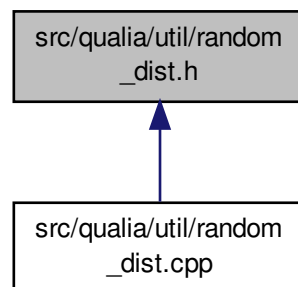
```
#include "random.h"
```

```
#include <math.h>
```

Include dependency graph for random_dist.h:



This graph shows which files directly or indirectly include this file:



Functions

- float [exponential](#) (float lambda)
- float [cauchy](#) (float median=0, float sigma=1)
- int [geometric](#) (float p)
- bool [bernoulli](#) (float p=0.5)

Returns true with probability \$p\$ and false with probability \$1-p\$ ($p > 0$).

6.124.1 Function Documentation

6.124.1.1 bool bernoulli (float $p = 0.5$)

Returns true with probability \$p\$ and false with probability \$1-p\$ ($p > 0$).

6.124.1.2 float `cauchy` (float *median* = 0, float *sigma* = 1)

Returns a random number from a Cauchy distribution. The Cauchy density is $p(x) = \sigma / (\pi * (\sigma^2 + (x - \text{median})^2))$

6.124.1.3 float `exponential` (float *lambda*)

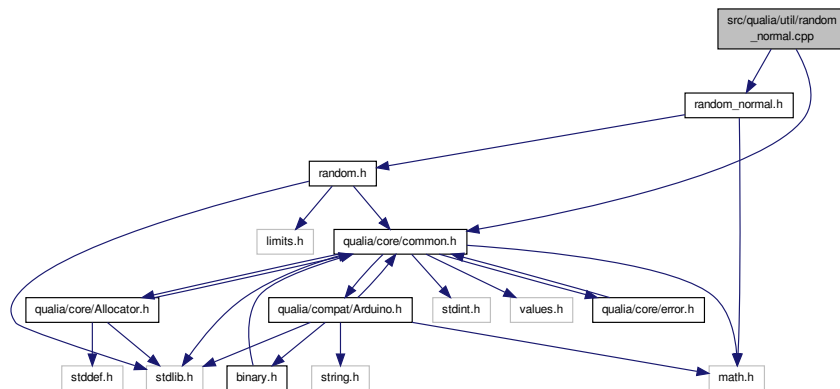
Provides methods for generating random numbers according to various probability distributions. Generates a random number from an exponential distribution. The density is $p(x) = \lambda * \exp(-\lambda * x)$, where λ is a positive number.

6.124.1.4 int `geometric` (float *p*)

Generates a random number from a geometric distribution. It returns an integer i , where $p(i) = (1-p) * p^{(i-1)}$. p must satisfy $0 < p < 1$.

6.125 src/qualia/util/random_normal.cpp File Reference

```
#include "random_normal.h"
#include <qualia/core/common.h>
Include dependency graph for random_normal.cpp:
```



Functions

- float `randomNormal` (float *mean*, float *stdv*)

6.125.1 Function Documentation

6.125.1.1 float `randomNormal` (float *mean* = 0, float *stdv* = 1)

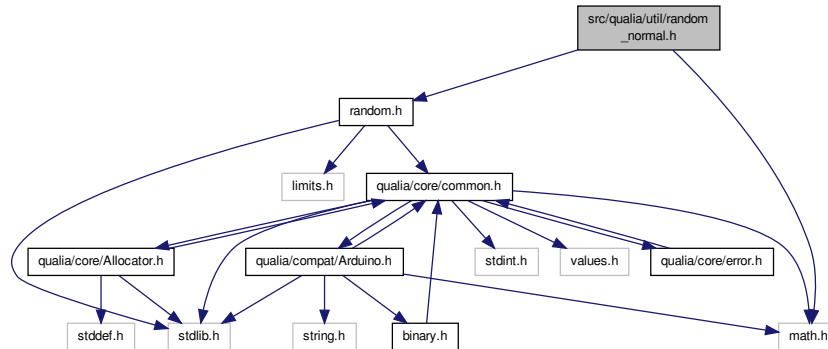
Provides methods for generating random numbers according to a normal distribution. Generates a random number from a normal distribution. (With mean *mean* and standard deviation *stdv* ≥ 0).

6.126 src/qualia/util/random_normal.h File Reference

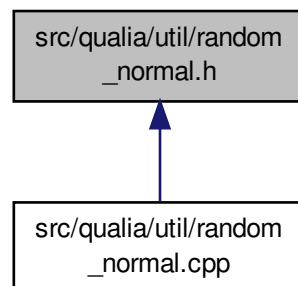
```
#include "random.h"
```

```
#include <math.h>
```

Include dependency graph for random_normal.h:



This graph shows which files directly or indirectly include this file:



Functions

- float [randomNormal](#) (float mean=0, float stdv=1)

6.126.1 Function Documentation

6.126.1.1 float randomNormal (float *mean* = 0, float *stdv* = 1)

Provides methods for generating random numbers according to a normal distribution. Generates a random number from a normal distribution. (With mean *mean* and standard deviation *stdv* ≥ 0).

Index

- ~Action
 - Action, [12](#)
- ~ActionProperties
 - ActionProperties, [15](#)
- ~Agent
 - Agent, [16](#)
- ~Allocator
 - Allocator, [19](#)
- ~BatchRLTrainer
 - BatchRLTrainer, [21](#)
- ~BinaryChromosome
 - BinaryChromosome, [24](#)
- ~BoolCmdOption
 - BoolCmdOption, [28](#)
- ~Chromosome
 - Chromosome, [29](#)
- ~CmdLine
 - CmdLine, [31](#)
- ~CmdOption
 - CmdOption, [35](#)
- ~DataSet
 - DataSet, [38](#)
- ~DataSetTrainer
 - DataSetTrainer, [40](#)
- ~DiskXFile
 - DiskXFile, [43](#)
- ~DiskXFileDataSet
 - DiskXFileDataSet, [46](#)
- ~Environment
 - Environment, [47](#)
- ~FileExportEnvironment
 - FileExportEnvironment, [49](#)
- ~Function
 - Function, [51](#)
- ~GradientFunction
 - GradientFunction, [54](#)
- ~IntCmdOption
 - IntCmdOption, [56](#)
- ~MapperBasicEnvironment
 - MapperBasicEnvironment, [59](#)
- ~MapperConnector
 - MapperConnector, [61](#)
- ~MapperEnvironment
 - MapperEnvironment, [64](#)
- ~MemoryDataSet
 - MemoryDataSet, [66](#)
- ~MultiAgent
 - MultiAgent, [68](#)
- ~MultiDataSet
 - MultiDataSet, [71](#)
- ~NeuralNetwork
 - NeuralNetwork, [74](#)
- ~Observation
 - Observation, [78](#)
- ~OscAgent
 - OscAgent, [81](#)
- ~OscBasicAgent
 - OscBasicAgent, [83](#)
- ~OscBasicEnvironment
 - OscBasicEnvironment, [85](#)
- ~OscEnvironment
 - OscEnvironment, [87](#)
- ~OscManager
 - OscManager, [88](#)
- ~OscRLEnvironment
 - OscRLEnvironment, [90](#)
- ~Policy
 - Policy, [92](#)
- ~QFunction
 - QFunction, [97](#)
- ~QLearningAgent
 - QLearningAgent, [100](#)
- ~QLearningEGreedyPolicy
 - QLearningEGreedyPolicy, [105](#)
- ~QLearningSoftmaxPolicy
 - QLearningSoftmaxPolicy, [107](#)
- ~Qualia
 - Qualia, [109](#)
- ~RealCmdOption
 - RealCmdOption, [111](#)
- ~Reward
 - Reward, [112](#)
- ~RewardEnvironment
 - RewardEnvironment, [114](#)
- ~SignalData
 - MapperConnector::SignalData, [119](#)
- ~StochasticDataSetTrainer
 - StochasticDataSetTrainer, [123](#)
- ~StringCmdOption
 - StringCmdOption, [127](#)
- ~SubDataSet
 - SubDataSet, [130](#)
- ~TDTrainer
 - TDTrainer, [132](#)
- ~Trainer
 - Trainer, [134](#)
- ~TupleDataSet
 - TupleDataSet, [137](#)

- ~XFile
 - XFile, [139](#)
- ~XFileDataSet
 - XFileDataSet, [141](#)
- __STRING
 - error.h, [202](#)
- _allocateLayer
 - NeuralNetwork, [74](#)
- _alpha
 - MovingAverage, [67](#)
- _backpropagateLayer
 - NeuralNetwork, [74](#)
- _client
 - OscManager, [89](#)
- _codeCrossoverOnePoint
 - BinaryChromosome, [24](#)
- _codeCrossoverTwoPoint
 - BinaryChromosome, [24](#)
- _currentExampleIndex
 - XFileDataSet, [142](#)
- _dataSet
 - SubDataSet, [130](#)
- _dataSets
 - MultiDataSet, [71](#)
- _deallocate
 - NeuralNetwork, [74](#)
- _deallocateLayer
 - NeuralNetwork, [74](#)
- _dim
 - ActionProperties, [15](#)
 - DataSet, [38](#)
 - Observation, [78](#)
- _diskFile
 - DiskXFileDataSet, [46](#)
- _doTrainEpisode
 - BatchRLTrainer, [21](#)
 - DataSetTrainer, [40](#)
 - StochasticDataSetTrainer, [123](#)
- _epsilonDiv
 - QLearningEDecreasingPolicy, [104](#)
- _file
 - XFileDataSet, [142](#)
- _function
 - Trainer, [135](#)
- _geneSizes
 - BinaryChromosomeProperties, [26](#)
- _indices
 - SubDataSet, [130](#)
- _lastObservation
 - BatchRLTrainer, [22](#)
- _lastOrNextAction
 - BatchRLTrainer, [22](#)
- _learningRateDiv
 - NeuralNetwork, [76](#)
- _nActions
 - ActionProperties, [15](#)
- _nConflated
 - ActionProperties, [15](#)
- _nDataSets
 - MultiDataSet, [71](#)
- _nEpisodes
 - Trainer, [135](#)
- _nExamples
 - DataSet, [38](#)
- _nGenes
 - BinaryChromosomeProperties, [26](#)
- _nParams
 - NeuralNetwork, [76](#)
- _observation
 - BatchRLTrainer, [22](#)
- _propagateLayer
 - NeuralNetwork, [74](#)
- _server
 - OscManager, [89](#)
- _startMillis
 - Stream, [126](#)
- _timeout
 - Stream, [126](#)
- _undefined
 - Action, [14](#)
- _value
 - MovingAverage, [67](#)
- abs
 - Arduino.h, [145](#)
- Action, [11](#)
 - ~Action, [12](#)
 - _undefined, [14](#)
 - Action, [12](#)
 - actions, [14](#)
 - conflated, [12](#)
 - copyFrom, [12](#)
 - dim, [13](#)
 - hasNext, [13](#)
 - loadData, [13](#)
 - nActions, [13](#)
 - nConflated, [13](#)
 - next, [13](#)
 - properties, [14](#)
 - reset, [13](#)
 - saveData, [13](#)
 - setConflated, [13](#)
 - undefined, [13](#)
- action
 - ObservationAction, [79](#)
- Action.h
 - action_dim_t, [196](#)
 - action_t, [196](#)
- action_dim_t
 - Action.h, [196](#)
- action_t
 - Action.h, [196](#)
- actionBuffer
 - OscAgent, [82](#)
- actionDim
 - FileExportEnvironment, [50](#)
 - MapperBasicEnvironment, [60](#)

- OscAgent, [82](#)
- OscEnvironment, [88](#)
- TDTrainer, [132](#)
- TupleDataSet, [137](#)
- ActionProperties, [14](#)
 - ~ActionProperties, [15](#)
 - _dim, [15](#)
 - _nActions, [15](#)
 - _nConflated, [15](#)
 - ActionProperties, [15](#)
 - ActionProperties, [15](#)
 - dim, [15](#)
 - equals, [15](#)
 - nActions, [15](#)
 - nConflated, [15](#)
 - random, [15](#)
- actionProperties
 - QFunction, [99](#)
- actionTypes
 - OscAgent, [82](#)
- actions
 - Action, [14](#)
- addBCmdArg
 - CmdLine, [31](#)
- addBCmdOption
 - CmdLine, [32](#)
- addCmdOption
 - CmdLine, [32](#)
- addICmdArg
 - CmdLine, [32](#)
- addICmdOption
 - CmdLine, [32](#)
- addInput
 - MapperConnector, [61](#)
- addMasterSwitch
 - CmdLine, [32](#)
- addOutput
 - MapperConnector, [61](#)
- addRCmdArg
 - CmdLine, [32](#)
- addRCmdOption
 - CmdLine, [32](#)
- addSCmdArg
 - CmdLine, [32](#)
- addSCmdOption
 - CmdLine, [32](#)
- addSignals
 - MapperBasicEnvironment, [59](#)
 - MapperEnvironment, [64](#)
- addText
 - CmdLine, [32](#)
- admin
 - MapperConnector, [62](#)
- Agent, [15](#)
 - ~Agent, [16](#)
 - end, [16](#)
 - init, [16](#)
 - load, [17](#)
 - save, [17](#)
 - start, [17](#)
 - step, [17](#)
- agent
 - Policy, [92](#)
 - Qualia, [109](#)
- agents
 - MultiAgent, [69](#)
- Alloc, [17](#)
 - free, [18](#)
 - init, [18](#)
 - inst, [18](#)
 - instance, [18](#)
 - malloc, [18](#)
 - realloc, [18](#)
- Allocator, [18](#)
 - ~Allocator, [19](#)
 - free, [19](#)
 - malloc, [19](#)
 - realloc, [19](#)
- Allocator.cpp
 - operator new, [197](#)
- Allocator.h
 - operator new, [198](#)
- appendAgentAction
 - MultiAgent, [69](#)
- Arduino.h
 - abs, [145](#)
 - bit, [145](#)
 - bitClear, [145](#)
 - bitRead, [145](#)
 - bitSet, [145](#)
 - bitWrite, [145](#)
 - boolean, [146](#)
 - byte, [146](#)
 - constrain, [145](#)
 - DEG_TO_RAD, [145](#)
 - degrees, [145](#)
 - delay, [146](#)
 - delayMicroseconds, [146](#)
 - false, [145](#)
 - HALF_PI, [145](#)
 - highByte, [145](#)
 - lowByte, [145](#)
 - max, [145](#)
 - micros, [146](#)
 - millis, [146](#)
 - min, [145](#)
 - PI, [145](#)
 - RAD_TO_DEG, [145](#)
 - radians, [145](#)
 - round, [145](#)
 - sq, [145](#)
 - TWO_PI, [146](#)
 - true, [145](#)
 - word, [146](#)
- argc
 - CmdLine, [33](#)

- argv
 - CmdLine, 33
- arrayBitClear
 - bitarray.cpp, 262
 - bitarray.h, 264
- arrayBitFlip
 - bitarray.cpp, 262
 - bitarray.h, 264
- arrayBitRead
 - bitarray.cpp, 262
 - bitarray.h, 264
- arrayBitSet
 - bitarray.cpp, 262
 - bitarray.h, 264
- arrayBitWrite
 - bitarray.cpp, 262
 - bitarray.h, 264
- arrayBlockCopy
 - bitarray.cpp, 262
 - bitarray.h, 264
- arrayBlockWrite
 - bitarray.cpp, 262
 - bitarray.h, 264
- ascii
 - DiskXFileDataSet, 46
- AsexualCrossover
 - Chromosome.h, 215
- associated_files
 - CmdLine, 33
- autoConnect
 - MapperConnector, 62
- available
 - Stream, 125
- B0
 - binary.h, 156
- B00
 - binary.h, 156
- B000
 - binary.h, 156
- B0000
 - binary.h, 156
- B00000
 - binary.h, 156
- B000000
 - binary.h, 156
- B0000000
 - binary.h, 156
- B00000000
 - binary.h, 156
- B000000000
 - binary.h, 156
- B000000001
 - binary.h, 156
- B0000001
 - binary.h, 156
- B00000010
 - binary.h, 156
- B00000011
 - binary.h, 156
- B000001
 - binary.h, 156
- B0000010
 - binary.h, 156
- B00000100
 - binary.h, 156
- B000001000
 - binary.h, 156
- B000001001
 - binary.h, 156
- B00000101
 - binary.h, 156
- B000001010
 - binary.h, 156
- B000001011
 - binary.h, 156
- B0000011
 - binary.h, 156
- B00000110
 - binary.h, 156
- B000001100
 - binary.h, 156
- B000001101
 - binary.h, 156
- B00000111
 - binary.h, 156
- B000001110
 - binary.h, 156
- B000001111
 - binary.h, 156
- B0001
 - binary.h, 157
- B00010
 - binary.h, 157
- B000100
 - binary.h, 157
- B0001000
 - binary.h, 157
- B00010000
 - binary.h, 157
- B00010001
 - binary.h, 157
- B0001001
 - binary.h, 157
- B00010010
 - binary.h, 157
- B0000010
 - binary.h, 156
- B00000100
 - binary.h, 156
- B00000101
 - binary.h, 156
- B0000011
 - binary.h, 156
- B00001
 - binary.h, 157
- B00010
 - binary.h, 157
- B000100
 - binary.h, 157
- B0001000
 - binary.h, 157
- B00010001
 - binary.h, 157
- B0001001
 - binary.h, 157
- B00010010
 - binary.h, 157

binary.h, 157
B00010011
binary.h, 157
B000101
binary.h, 157
B0001010
binary.h, 157
B00010100
binary.h, 157
B00010101
binary.h, 157
B0001011
binary.h, 158
B00010110
binary.h, 158
B00010111
binary.h, 158
B00011
binary.h, 158
B000110
binary.h, 158
B0001100
binary.h, 158
B00011000
binary.h, 158
B00011001
binary.h, 158
B0001101
binary.h, 158
B00011010
binary.h, 158
B00011011
binary.h, 158
B000111
binary.h, 158
B0001110
binary.h, 158
B00011100
binary.h, 158
B00011101
binary.h, 158
B0001111
binary.h, 158
B00011110
binary.h, 158
B00011111
binary.h, 158
B001
binary.h, 158
B0010
binary.h, 158
B00100
binary.h, 158
B001000
binary.h, 158
B0010000
binary.h, 158
B00100000
binary.h, 158
B00100001
binary.h, 158
B0010001
binary.h, 158
B00100010
binary.h, 158
B001000100
binary.h, 158
B001000101
binary.h, 158
B00100011
binary.h, 158
B001001
binary.h, 159
B0010010
binary.h, 159
B00100100
binary.h, 159
B00100101
binary.h, 159
B0010011
binary.h, 159
B00100110
binary.h, 159
B00100111
binary.h, 159
B00101
binary.h, 159
B001010
binary.h, 159
B0010100
binary.h, 159
B00101000
binary.h, 159
B00101001
binary.h, 159
B0010101
binary.h, 159
B00101010
binary.h, 159
B00101011
binary.h, 159
B001011
binary.h, 159
B0010110
binary.h, 159
B00101100
binary.h, 159
B00101101
binary.h, 159
B0010111
binary.h, 159
B00101110
binary.h, 159
B00101111
binary.h, 159
B0011
binary.h, 159
B00110
binary.h, 159
B001100

binary.h, 159	binary.h, 160
B0011000	B010
binary.h, 159	binary.h, 160
B00110000	B0100
binary.h, 159	binary.h, 160
B00110001	B01000
binary.h, 159	binary.h, 161
B0011001	B010000
binary.h, 160	binary.h, 161
B00110010	B0100000
binary.h, 160	binary.h, 161
B00110011	B01000000
binary.h, 160	binary.h, 161
B001101	B01000001
binary.h, 160	binary.h, 161
B0011010	B0100001
binary.h, 160	binary.h, 161
B00110100	B01000010
binary.h, 160	binary.h, 161
B00110101	B01000011
binary.h, 160	binary.h, 161
B0011011	B010001
binary.h, 160	binary.h, 161
B00110110	B0100010
binary.h, 160	binary.h, 161
B00110111	B01000100
binary.h, 160	binary.h, 161
B00111	B01000101
binary.h, 160	binary.h, 161
B001110	B0100011
binary.h, 160	binary.h, 161
B0011100	B01000110
binary.h, 160	binary.h, 161
B00111000	B01000111
binary.h, 160	binary.h, 161
B00111001	B01001
binary.h, 160	binary.h, 161
B0011101	B010010
binary.h, 160	binary.h, 161
B00111010	B0100100
binary.h, 160	binary.h, 161
B00111011	B01001000
binary.h, 160	binary.h, 161
B001111	B01001001
binary.h, 160	binary.h, 161
B0011110	B0100101
binary.h, 160	binary.h, 161
B00111100	B01001010
binary.h, 160	binary.h, 161
B00111101	B01001011
binary.h, 160	binary.h, 161
B0011111	B010011
binary.h, 160	binary.h, 161
B00111110	B0100110
binary.h, 160	binary.h, 161
B00111111	B01001100
binary.h, 160	binary.h, 161
B01	B01001101

binary.h, [161](#)
B0100111
binary.h, [161](#)
B01001110
binary.h, [162](#)
B01001111
binary.h, [162](#)
B0101
binary.h, [162](#)
B01010
binary.h, [162](#)
B010100
binary.h, [162](#)
B0101000
binary.h, [162](#)
B01010000
binary.h, [162](#)
B01010001
binary.h, [162](#)
B0101001
binary.h, [162](#)
B01010010
binary.h, [162](#)
B01010011
binary.h, [162](#)
B010101
binary.h, [162](#)
B0101010
binary.h, [162](#)
B01010100
binary.h, [162](#)
B01010101
binary.h, [162](#)
B0101011
binary.h, [162](#)
B01010110
binary.h, [162](#)
B01010111
binary.h, [162](#)
B01011
binary.h, [162](#)
B010110
binary.h, [162](#)
B0101100
binary.h, [162](#)
B01011000
binary.h, [162](#)
B01011001
binary.h, [162](#)
B0101101
binary.h, [162](#)
B01011010
binary.h, [162](#)
B01011011
binary.h, [162](#)
B010111
binary.h, [162](#)
B0101110
binary.h, [162](#)
binary.h, [162](#)
B01011100
binary.h, [162](#)
B01011101
binary.h, [163](#)
B0101111
binary.h, [163](#)
B01011110
binary.h, [163](#)
B01011111
binary.h, [163](#)
B011
binary.h, [163](#)
B0110
binary.h, [163](#)
B01100
binary.h, [163](#)
B011000
binary.h, [163](#)
B0110000
binary.h, [163](#)
B01100001
binary.h, [163](#)
B0110001
binary.h, [163](#)
B01100010
binary.h, [163](#)
B01100011
binary.h, [163](#)
B011001
binary.h, [163](#)
B0110010
binary.h, [163](#)
B01100100
binary.h, [163](#)
B01100101
binary.h, [163](#)
B0110011
binary.h, [163](#)
B01100110
binary.h, [163](#)
B01100111
binary.h, [163](#)
B01101
binary.h, [163](#)
B011010
binary.h, [163](#)
B0110100
binary.h, [163](#)
B01101000
binary.h, [163](#)
B01101001
binary.h, [163](#)
B0110101
binary.h, [163](#)
B01101010

binary.h, 164
B01101011
binary.h, 164
B011011
binary.h, 164
B0110110
binary.h, 164
B01101100
binary.h, 164
B01101101
binary.h, 164
B0110111
binary.h, 164
B01101110
binary.h, 164
B01101111
binary.h, 164
B0111
binary.h, 164
B01110
binary.h, 164
B011100
binary.h, 164
B0111000
binary.h, 164
B01110000
binary.h, 164
B01110001
binary.h, 164
B0111001
binary.h, 164
B01110010
binary.h, 164
B01110011
binary.h, 164
B011101
binary.h, 164
B0111010
binary.h, 164
B01110100
binary.h, 164
B01110101
binary.h, 164
B0111011
binary.h, 164
B01110110
binary.h, 164
B01110111
binary.h, 164
B01111
binary.h, 164
B011110
binary.h, 164
B0111100
binary.h, 164
B01111000
binary.h, 165
B01111001
binary.h, 164
binary.h, 165
B0111101
binary.h, 165
B01111010
binary.h, 165
B01111011
binary.h, 165
B0111110
binary.h, 165
B01111101
binary.h, 165
B0111111
binary.h, 165
B01111110
binary.h, 165
B01111111
binary.h, 165
B1
binary.h, 165
B10
binary.h, 165
B100
binary.h, 165
B1000
binary.h, 165
B10000
binary.h, 165
B100000
binary.h, 165
B1000000
binary.h, 165
B10000000
binary.h, 165
B10000001
binary.h, 165
B1000001
binary.h, 165
B10000010
binary.h, 165
B10000011
binary.h, 165
B100001
binary.h, 165
B1000010
binary.h, 165
B10000100
binary.h, 165
B10000101
binary.h, 165
B1000011
binary.h, 166
B10000110
binary.h, 166
B10000111

binary.h, 166
B10001
binary.h, 166
B100010
binary.h, 166
B1000100
binary.h, 166
B10001000
binary.h, 166
B10001001
binary.h, 166
B1000101
binary.h, 166
B10001010
binary.h, 166
B10001011
binary.h, 166
B100011
binary.h, 166
B1000110
binary.h, 166
B10001100
binary.h, 166
B10001101
binary.h, 166
B1000111
binary.h, 166
B10001110
binary.h, 166
B10001111
binary.h, 166
B1001
binary.h, 166
B10010
binary.h, 166
B100100
binary.h, 166
B1001000
binary.h, 166
B10010000
binary.h, 166
B10010001
binary.h, 166
B1001001
binary.h, 166
B10010010
binary.h, 166
B10010011
binary.h, 166
B100101
binary.h, 166
B1001010
binary.h, 167
B10010100
binary.h, 167
B10010101
binary.h, 167
B1001011
binary.h, 167
B10010110
binary.h, 167
B10010111
binary.h, 167
B10011
binary.h, 167
B100110
binary.h, 167
B1001100
binary.h, 167
B10011001
binary.h, 167
B1001101
binary.h, 167
B10011010
binary.h, 167
B10011011
binary.h, 167
B100111
binary.h, 167
B1001110
binary.h, 167
B10011100
binary.h, 167
B10011101
binary.h, 167
B1001111
binary.h, 167
B10011110
binary.h, 167
B10011111
binary.h, 167
B101
binary.h, 167
B1010
binary.h, 167
B10100
binary.h, 167
B101000
binary.h, 167
B1010000
binary.h, 167
B10100001
binary.h, 167
B1010001
binary.h, 168
B10100010
binary.h, 168
B10100011
binary.h, 168
B101001
binary.h, 168
B1010010

- binary.h, 168
- B10100100
 - binary.h, 168
- B10100101
 - binary.h, 168
- B1010011
 - binary.h, 168
- B10100110
 - binary.h, 168
- B10100111
 - binary.h, 168
- B10101
 - binary.h, 168
- B101010
 - binary.h, 168
- B1010100
 - binary.h, 168
- B10101000
 - binary.h, 168
- B10101001
 - binary.h, 168
- B1010101
 - binary.h, 168
- B10101010
 - binary.h, 168
- B10101011
 - binary.h, 168
- B101011
 - binary.h, 168
- B1010110
 - binary.h, 168
- B10101100
 - binary.h, 168
- B10101101
 - binary.h, 168
- B1010111
 - binary.h, 168
- B10101110
 - binary.h, 168
- B10101111
 - binary.h, 168
- B1011
 - binary.h, 168
- B10110
 - binary.h, 168
- B101100
 - binary.h, 168
- B1011000
 - binary.h, 169
- B10110000
 - binary.h, 169
- B10110001
 - binary.h, 169
- B1011001
 - binary.h, 169
- B10110010
 - binary.h, 169
- B10110011
 - binary.h, 169
- binary.h, 169
- B101101
 - binary.h, 169
- B1011010
 - binary.h, 169
- B10110100
 - binary.h, 169
- B10110101
 - binary.h, 169
- B1011011
 - binary.h, 169
- B10110110
 - binary.h, 169
- B10110111
 - binary.h, 169
- B10111
 - binary.h, 169
- B101110
 - binary.h, 169
- B1011100
 - binary.h, 169
- B10111000
 - binary.h, 169
- B10111001
 - binary.h, 169
- B1011101
 - binary.h, 169
- B10111010
 - binary.h, 169
- B10111011
 - binary.h, 169
- B101111
 - binary.h, 169
- B1011110
 - binary.h, 169
- B10111100
 - binary.h, 169
- B10111101
 - binary.h, 169
- B1011111
 - binary.h, 169
- B10111110
 - binary.h, 169
- B10111111
 - binary.h, 169
- B11
 - binary.h, 170
- B110
 - binary.h, 170
- B1100
 - binary.h, 170
- B11000
 - binary.h, 170
- B110000
 - binary.h, 170
- B1100000
 - binary.h, 170
- B11000000

binary.h, 170
B11000001
binary.h, 170
B1100001
binary.h, 170
B11000010
binary.h, 170
B11000011
binary.h, 170
B110001
binary.h, 170
B1100010
binary.h, 170
B11000100
binary.h, 170
B11000101
binary.h, 170
B1100011
binary.h, 170
B11000110
binary.h, 170
B11000111
binary.h, 170
B11001
binary.h, 170
B110010
binary.h, 170
B1100100
binary.h, 170
B11001000
binary.h, 170
B11001001
binary.h, 170
B1100101
binary.h, 170
B11001010
binary.h, 170
B11001011
binary.h, 170
B110011
binary.h, 170
B1100110
binary.h, 170
B11001100
binary.h, 171
B11001101
binary.h, 171
B1100111
binary.h, 171
B11001110
binary.h, 171
B11001111
binary.h, 171
B1101
binary.h, 171
B11010
binary.h, 171
B110100
binary.h, 171
B1101000
binary.h, 171
B11010000
binary.h, 171
B11010001
binary.h, 171
B1101001
binary.h, 171
B11010010
binary.h, 171
B11010011
binary.h, 171
B110101
binary.h, 171
B1101010
binary.h, 171
B11010100
binary.h, 171
B11010101
binary.h, 171
B1101011
binary.h, 171
B11010110
binary.h, 171
B11010111
binary.h, 171
B11011
binary.h, 171
B110110
binary.h, 171
B1101100
binary.h, 171
B11011000
binary.h, 171
B11011001
binary.h, 171
B1101101
binary.h, 171
B11011010
binary.h, 172
B11011011
binary.h, 172
B110111
binary.h, 172
B1101110
binary.h, 172
B11011100
binary.h, 172
B11011101
binary.h, 172
B1101111
binary.h, 172
B11011110
binary.h, 172
B11011111
binary.h, 172
B111

binary.h, [174](#)
 B11111101
 binary.h, [174](#)
 B1111111
 binary.h, [174](#)
 B11111110
 binary.h, [174](#)
 B11111111
 binary.h, [174](#)
 BIN
 Print.h, [177](#)
 backpropagate
 GradientFunction, [54](#)
 NeuralNetwork, [74](#)
 QFunction, [97](#)
 BatchRLTrainer, [20](#)
 ~BatchRLTrainer, [21](#)
 _doTrainEpisode, [21](#)
 _lastObservation, [22](#)
 _lastOrNextAction, [22](#)
 _observation, [22](#)
 BatchRLTrainer, [21](#)
 BatchRLTrainer, [21](#)
 gamma, [22](#)
 init, [21](#)
 maxExamples, [22](#)
 maxTrainingIterationsPerEpisode, [22](#)
 targets, [22](#)
 bernouilli
 random_dist.cpp, [271](#)
 random_dist.h, [272](#)
 binary.h
 B0, [156](#)
 B00, [156](#)
 B000, [156](#)
 B0000, [156](#)
 B00000, [156](#)
 B000000, [156](#)
 B0000000, [156](#)
 B00000000, [156](#)
 B000000001, [156](#)
 B0000001, [156](#)
 B00000010, [156](#)
 B00000011, [156](#)
 B000001, [156](#)
 B0000010, [156](#)
 B00000100, [156](#)
 B00000101, [156](#)
 B0000011, [156](#)
 B00000110, [156](#)
 B00000111, [156](#)
 B00001, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)
 B0000100, [157](#)
 B00001000, [157](#)
 B00001001, [157](#)
 B0000101, [157](#)
 B00001010, [157](#)
 B00001011, [157](#)
 B000011, [157](#)
 B000010, [157](#)

B0010100, [159](#)
B00101000, [159](#)
B00101001, [159](#)
B0010101, [159](#)
B00101010, [159](#)
B00101011, [159](#)
B001011, [159](#)
B0010110, [159](#)
B00101100, [159](#)
B00101101, [159](#)
B0010111, [159](#)
B00101110, [159](#)
B00101111, [159](#)
B0011, [159](#)
B00110, [159](#)
B001100, [159](#)
B0011000, [159](#)
B00110000, [159](#)
B00110001, [159](#)
B0011001, [160](#)
B00110010, [160](#)
B00110011, [160](#)
B001101, [160](#)
B0011010, [160](#)
B00110100, [160](#)
B00110101, [160](#)
B0011011, [160](#)
B00110110, [160](#)
B00110111, [160](#)
B00111, [160](#)
B001110, [160](#)
B0011100, [160](#)
B00111000, [160](#)
B00111001, [160](#)
B0011101, [160](#)
B00111010, [160](#)
B00111011, [160](#)
B001111, [160](#)
B0011110, [160](#)
B00111100, [160](#)
B00111101, [160](#)
B0011111, [160](#)
B00111110, [160](#)
B00111111, [160](#)
B01, [160](#)
B010, [160](#)
B0100, [160](#)
B01000, [161](#)
B010000, [161](#)
B0100000, [161](#)
B01000000, [161](#)
B01000001, [161](#)
B0100001, [161](#)
B01000010, [161](#)
B01000011, [161](#)
B010001, [161](#)
B0100010, [161](#)
B01000100, [161](#)
B01000101, [161](#)
B0100011, [161](#)
B01000110, [161](#)
B01000111, [161](#)
B01001, [161](#)
B010010, [161](#)
B0100100, [161](#)
B01001000, [161](#)
B01001001, [161](#)
B0100101, [161](#)
B01001010, [161](#)
B01001011, [161](#)
B010011, [161](#)
B0100110, [161](#)
B01001100, [161](#)
B01001101, [161](#)
B0100111, [161](#)
B01001110, [162](#)
B01001111, [162](#)
B0101, [162](#)
B01010, [162](#)
B010100, [162](#)
B0101000, [162](#)
B01010000, [162](#)
B01010001, [162](#)
B0101001, [162](#)
B01010010, [162](#)
B01010011, [162](#)
B010101, [162](#)
B0101010, [162](#)
B01010100, [162](#)
B01010101, [162](#)
B0101011, [162](#)
B01010110, [162](#)
B01010111, [162](#)
B01011, [162](#)
B010110, [162](#)
B0101100, [162](#)
B01011000, [162](#)
B01011001, [162](#)
B0101101, [162](#)
B01011010, [162](#)
B01011011, [162](#)
B010111, [162](#)
B0101110, [162](#)
B01011100, [163](#)
B01011101, [163](#)
B0101111, [163](#)
B01011110, [163](#)
B01011111, [163](#)
B011, [163](#)
B0110, [163](#)
B01100, [163](#)
B011000, [163](#)
B0110000, [163](#)
B01100000, [163](#)
B01100001, [163](#)
B0110001, [163](#)

B01100010, [163](#)
B01100011, [163](#)
B011001, [163](#)
B0110010, [163](#)
B01100100, [163](#)
B01100101, [163](#)
B0110011, [163](#)
B01100110, [163](#)
B01100111, [163](#)
B01101, [163](#)
B011010, [163](#)
B0110100, [163](#)
B01101000, [163](#)
B01101001, [163](#)
B0110101, [163](#)
B01101010, [164](#)
B01101011, [164](#)
B011011, [164](#)
B0110110, [164](#)
B01101100, [164](#)
B01101101, [164](#)
B0110111, [164](#)
B01101110, [164](#)
B01101111, [164](#)
B0111, [164](#)
B01110, [164](#)
B011100, [164](#)
B0111000, [164](#)
B01110000, [164](#)
B01110001, [164](#)
B0111001, [164](#)
B01110010, [164](#)
B01110011, [164](#)
B011101, [164](#)
B0111010, [164](#)
B01110100, [164](#)
B01110101, [164](#)
B0111011, [164](#)
B01110110, [164](#)
B01110111, [164](#)
B01111, [164](#)
B011110, [164](#)
B0111100, [164](#)
B01111000, [165](#)
B01111001, [165](#)
B0111101, [165](#)
B01111010, [165](#)
B01111011, [165](#)
B011111, [165](#)
B0111110, [165](#)
B01111100, [165](#)
B01111101, [165](#)
B0111111, [165](#)
B01111110, [165](#)
B01111111, [165](#)
B1, [165](#)
B10, [165](#)
B100, [165](#)
B1000, [165](#)
B10000, [165](#)
B100000, [165](#)
B1000000, [165](#)
B10000000, [165](#)
B10000001, [165](#)
B1000001, [165](#)
B10000010, [165](#)
B10000011, [165](#)
B100001, [165](#)
B1000010, [165](#)
B10000100, [165](#)
B10000101, [165](#)
B1000011, [166](#)
B10000110, [166](#)
B10000111, [166](#)
B10001, [166](#)
B100010, [166](#)
B1000100, [166](#)
B10001000, [166](#)
B10001001, [166](#)
B1000101, [166](#)
B10001010, [166](#)
B10001011, [166](#)
B100011, [166](#)
B1000110, [166](#)
B10001100, [166](#)
B10001101, [166](#)
B1000111, [166](#)
B10001110, [166](#)
B10001111, [166](#)
B1001, [166](#)
B10010, [166](#)
B100100, [166](#)
B1001000, [166](#)
B10010000, [166](#)
B10010001, [166](#)
B1001001, [166](#)
B10010010, [166](#)
B10010011, [166](#)
B100101, [166](#)
B1001010, [167](#)
B10010100, [167](#)
B10010101, [167](#)
B1001011, [167](#)
B10010110, [167](#)
B10010111, [167](#)
B10011, [167](#)
B100110, [167](#)
B1001100, [167](#)
B10011000, [167](#)
B10011001, [167](#)
B1001101, [167](#)
B10011010, [167](#)
B10011011, [167](#)
B100111, [167](#)
B1001110, [167](#)
B10011100, [167](#)

B10011101, [167](#)
B1001111, [167](#)
B10011110, [167](#)
B10011111, [167](#)
B101, [167](#)
B1010, [167](#)
B10100, [167](#)
B101000, [167](#)
B1010000, [167](#)
B10100000, [167](#)
B10100001, [167](#)
B1010001, [168](#)
B10100010, [168](#)
B10100011, [168](#)
B101001, [168](#)
B1010010, [168](#)
B10100100, [168](#)
B10100101, [168](#)
B1010011, [168](#)
B10100110, [168](#)
B10100111, [168](#)
B10101, [168](#)
B101010, [168](#)
B1010100, [168](#)
B10101000, [168](#)
B10101001, [168](#)
B1010101, [168](#)
B10101010, [168](#)
B10101011, [168](#)
B101011, [168](#)
B1010110, [168](#)
B10101100, [168](#)
B10101101, [168](#)
B1010111, [168](#)
B10101110, [168](#)
B10101111, [168](#)
B1011, [168](#)
B10110, [168](#)
B101100, [168](#)
B1011000, [169](#)
B10110000, [169](#)
B10110001, [169](#)
B1011001, [169](#)
B10110010, [169](#)
B10110011, [169](#)
B101101, [169](#)
B1011010, [169](#)
B10110100, [169](#)
B10110101, [169](#)
B1011011, [169](#)
B10110110, [169](#)
B10110111, [169](#)
B10111, [169](#)
B101110, [169](#)
B1011100, [169](#)
B10111000, [169](#)
B10111001, [169](#)
B10111010, [169](#)
B10111011, [169](#)
B1011101010, [169](#)
B1011101011, [169](#)
B10111011, [169](#)
B101110110, [169](#)
B101110111, [169](#)
B11, [170](#)
B110, [170](#)
B1100, [170](#)
B11000, [170](#)
B110000, [170](#)
B1100000, [170](#)
B11000000, [170](#)
B110000000, [170](#)
B11000001, [170](#)
B1100001, [170](#)
B11000010, [170](#)
B11000011, [170](#)
B110000101, [170](#)
B110000110, [170](#)
B110000111, [170](#)
B110001, [170](#)
B1100010, [170](#)
B11000100, [170](#)
B11000101, [170](#)
B1100011, [170](#)
B11000110, [170](#)
B11000111, [170](#)
B11001, [170](#)
B110010, [170](#)
B1100100, [170](#)
B11001000, [170](#)
B11001001, [170](#)
B1100101, [170](#)
B11001010, [170](#)
B11001011, [170](#)
B110011, [170](#)
B1100110, [170](#)
B11001100, [171](#)
B11001101, [171](#)
B1100111, [171](#)
B11001110, [171](#)
B11001111, [171](#)
B1101, [171](#)
B11010, [171](#)
B110100, [171](#)
B1101000, [171](#)
B11010000, [171](#)
B11010001, [171](#)
B1101001, [171](#)
B11010010, [171](#)
B11010011, [171](#)
B110101, [171](#)
B1101010, [171](#)
B11010100, [171](#)
B11010101, [171](#)
B1101011, [171](#)
B11010110, [171](#)
B11010111, [171](#)

- B11011, [171](#)
- B110110, [171](#)
- B1101100, [171](#)
- B11011000, [171](#)
- B11011001, [171](#)
- B1101101, [171](#)
- B11011010, [171](#)
- B11011011, [172](#)
- B110111, [172](#)
- B1101110, [172](#)
- B11011100, [172](#)
- B11011101, [172](#)
- B1101111, [172](#)
- B11011110, [172](#)
- B11011111, [172](#)
- B111, [172](#)
- B1110, [172](#)
- B11100, [172](#)
- B111000, [172](#)
- B1110000, [172](#)
- B11100000, [172](#)
- B11100001, [172](#)
- B1110001, [172](#)
- B11100010, [172](#)
- B11100011, [172](#)
- B111001, [172](#)
- B1110010, [172](#)
- B11100100, [172](#)
- B11100101, [172](#)
- B1110011, [172](#)
- B11100110, [172](#)
- B11100111, [172](#)
- B11101, [172](#)
- B111010, [172](#)
- B1110100, [172](#)
- B11101000, [173](#)
- B11101001, [173](#)
- B1110101, [173](#)
- B11101010, [173](#)
- B11101011, [173](#)
- B111011, [173](#)
- B1110110, [173](#)
- B11101100, [173](#)
- B11101101, [173](#)
- B1110111, [173](#)
- B11101110, [173](#)
- B11101111, [173](#)
- B1111, [173](#)
- B11110, [173](#)
- B111100, [173](#)
- B1111000, [173](#)
- B11110000, [173](#)
- B11110001, [173](#)
- B1111001, [173](#)
- B11110010, [173](#)
- B11110011, [173](#)
- B111101, [173](#)
- B1111010, [173](#)
- B11110100, [173](#)
- B11110101, [173](#)
- B1111011, [173](#)
- B11110110, [173](#)
- B11110111, [173](#)
- B11111, [174](#)
- B111110, [174](#)
- B1111100, [174](#)
- B11111000, [174](#)
- B11111001, [174](#)
- B1111101, [174](#)
- B11111010, [174](#)
- B11111011, [174](#)
- B111111, [174](#)
- B1111110, [174](#)
- B11111100, [174](#)
- B11111101, [174](#)
- B1111111, [174](#)
- B11111110, [174](#)
- B11111111, [174](#)
- BinaryChromosome, [22](#)
 - ~BinaryChromosome, [24](#)
 - _codeCrossoverOnePoint, [24](#)
 - _codeCrossoverTwoPoint, [24](#)
 - BinaryChromosome, [24](#)
 - BinaryChromosome, [24](#)
 - code, [25](#)
 - copyFrom, [24](#)
 - crossoverOnePoint, [24](#)
 - crossoverTwoPoint, [24](#)
 - equals, [24](#)
 - getGeneValue, [24](#)
 - info, [25](#)
 - init, [24](#)
 - initializeRandom, [24](#)
 - mutate, [24](#)
 - mutateFlip, [24](#)
 - setGeneValue, [24](#)
- BinaryChromosomeProperties, [25](#)
 - _geneSizes, [26](#)
 - _nGenes, [26](#)
 - BinaryChromosomeProperties, [26](#)
 - BinaryChromosomeProperties, [26](#)
 - bitSize, [26](#)
 - byteSize, [26](#)
 - equals, [26](#)
 - evaluator, [26](#)
 - geneSize, [26](#)
 - getStartBitPosition, [26](#)
 - initializer, [27](#)
 - mutator, [27](#)
 - nGenes, [26](#)
- bit
 - Arduino.h, [145](#)
- bitClear
 - Arduino.h, [145](#)
- bitFlip
 - common.h, [199](#)

- bitRead
 - Arduino.h, 145
- bitSet
 - Arduino.h, 145
- bitSize
 - BinaryChromosomeProperties, 26
- bitWrite
 - Arduino.h, 145
- bitarray.cpp
 - arrayBitClear, 262
 - arrayBitFlip, 262
 - arrayBitRead, 262
 - arrayBitSet, 262
 - arrayBitWrite, 262
 - arrayBlockCopy, 262
 - arrayBlockWrite, 262
 - initPointerAndPositionForBitOperation, 262
- bitarray.h
 - arrayBitClear, 264
 - arrayBitFlip, 264
 - arrayBitRead, 264
 - arrayBitSet, 264
 - arrayBitWrite, 264
 - arrayBlockCopy, 264
 - arrayBlockWrite, 264
 - initPointerAndPositionForBitOperation, 264
- BoolCmdOption, 27
 - ~BoolCmdOption, 28
 - BoolCmdOption, 28
 - BoolCmdOption, 28
 - init_value, 29
 - initValue, 28
 - load, 28
 - ptr, 29
 - read, 28
 - save, 28
- boolean
 - Arduino.h, 146
- buffer
 - StaticAllocator, 121
- buffer_block
 - DiskXFile, 44
- buffer_block_size
 - DiskXFile, 44
- bufferAction
 - TDTrainer, 132
- bufferIdx
 - StaticAllocator, 121
- bufferSize
 - StaticAllocator, 121
- byte
 - Arduino.h, 146
- byteSize
 - BinaryChromosomeProperties, 26
- cauchy
 - random_dist.cpp, 271
 - random_dist.h, 272
- chooseAction
 - Policy, 92
 - QLearningEDecreasingPolicy, 103
 - QLearningEGreedyPolicy, 105
 - QLearningSoftmaxPolicy, 107
- Chromosome, 29
 - ~Chromosome, 29
 - Chromosome, 29
 - copyFrom, 29
 - equals, 29
 - init, 30
 - mutate, 30
- Chromosome.h
 - AsexualCrossover, 215
 - Comparator, 215
 - Evaluator, 215
 - Initializer, 215
 - Mutator, 215
 - SexualCrossover, 215
- clearDelta
 - GradientFunction, 54
- clearWriteError
 - Print, 93
- client
 - OscManager, 88
- cmd_options
 - CmdLine, 33
- CmdLine, 30
 - ~CmdLine, 31
 - addBCmdArg, 31
 - addBCmdOption, 32
 - addCmdOption, 32
 - addICmdArg, 32
 - addICmdOption, 32
 - addMasterSwitch, 32
 - addRCmdArg, 32
 - addRCmdOption, 32
 - addSCmdArg, 32
 - addSCmdOption, 32
 - addText, 32
 - argc, 33
 - argv, 33
 - associated_files, 33
 - cmd_options, 33
 - CmdLine, 31
 - CmdLine, 31
 - getPath, 32
 - getXFile, 32
 - help, 32
 - info, 33
 - load, 33
 - master_switch, 33
 - n_associated_files, 33
 - n_cmd_options, 33
 - n_master_switches, 33
 - program_name, 33
 - read, 33
 - save, 33
 - setWorkingDirectory, 33

- text_info, 33
- working_directory, 33
- write_log, 34
- writeLog, 33
- CmdOption, 34
 - ~CmdOption, 35
 - CmdOption, 35
 - CmdOption, 35
 - help, 36
 - initValue, 35
 - is_argument, 36
 - is_master_switch, 36
 - is_option, 36
 - is_setted, 36
 - is_text, 36
 - isArgument, 35
 - isCurrent, 35
 - isMasterSwitch, 35
 - isOption, 35
 - isText, 35
 - load, 35
 - name, 36
 - needsSave, 36
 - printValue, 36
 - read, 36
 - save, 36
 - type_name, 36
- code
 - BinaryChromosome, 25
- combineAgentActions
 - MultiAgent, 69
- common.h
 - bitFlip, 199
 - INF, 199
 - is_arduino, 199
 - is_avr, 199
 - is_computer, 199
 - RANDOM_MAX, 199
 - real, 199
- Comparator
 - Chromosome.h, 215
- conflated
 - Action, 12
- connector
 - MapperEnvironment, 64
- constrain
 - Arduino.h, 145
- copyFrom
 - Action, 12
 - BinaryChromosome, 24
 - Chromosome, 29
 - Observation, 78
 - RLObservation, 116
- createConnections
 - MapperConnector, 61
- crossoverOnePoint
 - BinaryChromosome, 24
- crossoverTwoPoint
 - BinaryChromosome, 24
- currentAction
 - OscBasicAgent, 83
 - QLearningAgent, 101
- currentExampleIndex
 - TupleDataSet, 137
- currentObservation
 - MapperBasicEnvironment, 60
 - OscBasicEnvironment, 85
 - OscRLEnvironment, 90
- DEBUG_ERROR
 - error.h, 202
- DEBUG_LEVEL
 - error.h, 202
- DEBUG_LEVEL_ERROR
 - error.h, 202
- DEBUG_LEVEL_NODEBUG
 - error.h, 202
- DEBUG_LEVEL_NOTICE
 - error.h, 202
- DEBUG_LEVEL_WARNING
 - error.h, 203
- DEBUG_NOTICE
 - error.h, 203
- DEBUG_WARNING
 - error.h, 203
- DEC
 - Print.h, 177
- DEG_TO_RAD
 - Arduino.h, 145
- dWeight
 - NeuralNetwork::Layer, 58
- dWeights
 - GradientFunction, 55
- data
 - MapperConnector::SignalData, 119
 - MemoryDataSet, 66
- DataSet, 37
 - ~DataSet, 38
 - _dim, 38
 - _nExamples, 38
 - DataSet, 38
 - DataSet, 38
 - dim, 38
 - example, 38
 - init, 38
 - nExamples, 38
 - reset, 38
 - setExample, 38
- DataSetTrainer, 39
 - ~DataSetTrainer, 40
 - _doTrainEpisode, 40
 - DataSetTrainer, 40
 - DataSetTrainer, 40
 - stop, 40
 - train, 40
 - trainEpisode, 40
- db

- MapperConnector, 62
- decreaseConstant
 - NeuralNetwork, 76
 - QLearningEDecreasingPolicy, 104
- degrees
 - Arduino.h, 145
- delay
 - Arduino.h, 146
- delayMicroseconds
 - Arduino.h, 146
- dev
 - MapperConnector, 62
- devDbCallback
 - MapperConnector, 61
- deviceName
 - MapperConnector, 62
- dim
 - Action, 13
 - ActionProperties, 15
 - DataSet, 38
 - Observation, 78
- DiskXFile, 41
 - ~DiskXFile, 43
 - buffer_block, 44
 - buffer_block_size, 44
 - DiskXFile, 42
 - DiskXFile, 42
 - eof, 43
 - file, 44
 - flush, 43
 - gets, 43
 - is_native_mode, 44
 - is_opened, 44
 - isBigEndianProcessor, 43
 - isLittleEndianProcessor, 43
 - isNativeMode, 43
 - its_a_pipe, 44
 - printf, 43
 - read, 43
 - reverseMemory, 43
 - rewind, 43
 - scanf, 43
 - seek, 44
 - setBigEndianMode, 44
 - setLittleEndianMode, 44
 - setNativeMode, 44
 - tell, 44
 - write, 44
- DiskXFileDataSet, 45
 - ~DiskXFileDataSet, 46
 - _diskFile, 46
 - ascii, 46
 - DiskXFileDataSet, 46
 - DiskXFileDataSet, 46
 - init, 46
 - reset, 46
 - setExample, 46
- DiskXFileDataSet.cpp
- REAL_FORMAT, 188
- doAction
 - RewardEnvironment, 115
- doStart
 - RewardEnvironment, 115
- eTraces
 - TDTrainer, 132
- end
 - Agent, 16
 - MultiAgent, 69
 - QLearningAgent, 100
- env
 - FileExportEnvironment, 50
- Environment, 47
 - ~Environment, 47
 - init, 47
 - start, 47
 - step, 47
- environment
 - Qualia, 109
 - Reward, 113
- eof
 - DiskXFile, 43
 - XFile, 139
- episode
 - Qualia, 109
- epsilon
 - QLearningEGreedyPolicy, 106
 - QLearningSoftmaxPolicy, 107
- equals
 - ActionProperties, 15
 - BinaryChromosome, 24
 - BinaryChromosomeProperties, 26
 - Chromosome, 29
- error
 - NeuralNetwork::Layer, 58
 - OscManager, 89
- error.h
 - __STRING, 202
 - DEBUG_ERROR, 202
 - DEBUG_LEVEL, 202
 - DEBUG_LEVEL_ERROR, 202
 - DEBUG_LEVEL_NODEBUG, 202
 - DEBUG_LEVEL_NOTICE, 202
 - DEBUG_LEVEL_WARNING, 203
 - DEBUG_NOTICE, 203
 - DEBUG_WARNING, 203
 - Q_ASSERT_ERROR, 203
 - Q_ASSERT_NOTICE, 203
 - Q_ASSERT_WARNING, 203
 - Q_ERROR, 203
 - Q_MESSAGE, 203
 - Q_NOTICE, 203
 - Q_PRINT, 203
 - Q_WARNING, 203
- Evaluator
 - Chromosome.h, 215
- evaluator

- BinaryChromosomeProperties, 26
- example
 - DataSet, 38
- exponential
 - random_dist.cpp, 271
 - random_dist.h, 273
- extractAgentObservation
 - MultiAgent, 69
- false
 - Arduino.h, 145
- file
 - DiskXFile, 44
 - FileExportEnvironment, 50
 - TupleDataSet, 137
- FileExportEnvironment, 48
 - ~FileExportEnvironment, 49
 - actionDim, 50
 - env, 50
 - file, 50
 - FileExportEnvironment, 49
 - FileExportEnvironment, 49
 - init, 49
 - observationDim, 50
 - start, 49
 - step, 49
- find
 - Stream, 125
- findUntil
 - Stream, 125
- flag
 - MapperConnector::SignalData, 119
- flush
 - DiskXFile, 43
 - Stream, 125
 - XFile, 139
- free
 - Alloc, 18
 - Allocator, 19
 - StaticAllocator, 121
- freeAll
 - StaticAllocator, 121
- Function, 50
 - ~Function, 51
 - Function, 51
 - getOutput, 51
 - getOutputs, 51
 - init, 51
 - load, 51
 - nInputs, 51
 - nOutputs, 51
 - propagate, 52
 - save, 52
 - setInput, 52
 - setInputs, 52
- function
 - QFunction, 99
- gamma
 - BatchRLTrainer, 22
 - TDTrainer, 132
- geneSize
 - BinaryChromosomeProperties, 26
- geometric
 - random_dist.cpp, 271
 - random_dist.h, 273
- get
 - MovingAverage, 67
- getCurrentEpsilon
 - QLearningEDecreasingPolicy, 103
- getCurrentLearningRate
 - NeuralNetwork, 74
- getGeneValue
 - BinaryChromosome, 24
- getMaxAction
 - QFunction, 97
- getOutput
 - Function, 51
 - NeuralNetwork, 74
 - QFunction, 97
- getOutputs
 - Function, 51
 - NeuralNetwork, 74
- getPath
 - CmdLine, 32
 - OscAgent, 81
 - OscEnvironment, 87
- getShuffledDataSet
 - SubDataSet, 130
- getStandardDataSets
 - SubDataSet, 130
- getStartBitPosition
 - BinaryChromosomeProperties, 26
- getValue
 - QFunction, 97
- getWriteError
 - Print, 93
- getXFile
 - CmdLine, 32
- gets
 - DiskXFile, 43
 - XFile, 139
- GradientFunction, 52
 - ~GradientFunction, 54
 - backpropagate, 54
 - clearDelta, 54
 - dWeights, 55
 - GradientFunction, 54
 - GradientFunction, 54
 - load, 54
 - nParams, 54
 - save, 54
 - update, 54
 - weights, 55
- HALF_PI
 - Arduino.h, 145
- HEX

- Print.h, 177
- handlerInit
 - OscAgent, 81
 - OscEnvironment, 87
- handlerStartStep
 - OscAgent, 81
 - OscEnvironment, 87
- hasNext
 - Action, 13
- help
 - CmdLine, 32
 - CmdOption, 36
- hiddenLayer
 - NeuralNetwork, 76
- highByte
 - Arduino.h, 145
- INF
 - common.h, 199
- id
 - OscAgent, 82
 - OscEnvironment, 88
- info
 - BinaryChromosome, 25
 - CmdLine, 33
- init
 - Agent, 16
 - Alloc, 18
 - BatchRLTrainer, 21
 - BinaryChromosome, 24
 - Chromosome, 30
 - DataSet, 38
 - DiskXFileDataSet, 46
 - Environment, 47
 - FileExportEnvironment, 49
 - Function, 51
 - MapperConnector, 61
 - MapperEnvironment, 64
 - MultiAgent, 69
 - MultiDataSet, 71
 - NeuralNetwork, 74
 - OscAgent, 81
 - OscEnvironment, 87
 - Policy, 92
 - QFunction, 98
 - QLearningAgent, 100
 - QLearningEDecreasingPolicy, 104
 - Qualia, 109
 - RLQualia, 118
 - SubDataSet, 130
 - TDTrainer, 132
 - Trainer, 134
 - TupleDataSet, 137
 - XFileDataSet, 141
- init_value
 - BoolCmdOption, 29
 - IntCmdOption, 57
 - RealCmdOption, 111
 - StringCmdOption, 128
- initOsc
 - OscManager, 89
- initPointerAndPositionForBitOperation
 - bitarray.cpp, 262
 - bitarray.h, 264
- initValue
 - BoolCmdOption, 28
 - CmdOption, 35
 - IntCmdOption, 57
 - RealCmdOption, 111
 - StringCmdOption, 128
- initialPort
 - MapperConnector, 62
- initializeRandom
 - BinaryChromosome, 24
- Initializer
 - Chromosome.h, 215
- initializer
 - BinaryChromosomeProperties, 27
- input
 - QFunction, 99
- inputData
 - MapperConnector, 62
- inputLayer
 - NeuralNetwork, 76
- inst
 - Alloc, 18
- instance
 - Alloc, 18
- IntCmdOption, 55
 - ~IntCmdOption, 56
 - init_value, 57
 - initValue, 57
 - IntCmdOption, 56
 - IntCmdOption, 56
 - load, 57
 - printValue, 57
 - ptr, 57
 - read, 57
 - save, 57
- is_arduino
 - common.h, 199
- is_argument
 - CmdOption, 36
- is_avr
 - common.h, 199
- is_computer
 - common.h, 199
- is_master_switch
 - CmdOption, 36
- is_native_mode
 - DiskXFile, 44
- is_opened
 - DiskXFile, 44
- is_option
 - CmdOption, 36
- is_setted
 - CmdOption, 36

- is_text
 - CmdOption, 36
- isArgument
 - CmdOption, 35
- isBigEndianProcessor
 - DiskXFile, 43
- isBlocking
 - MapperConnector::SignalData, 119
- isCurrent
 - CmdOption, 35
- isLearning
 - QLearningAgent, 101
- isLittleEndianProcessor
 - DiskXFile, 43
- isMasterSwitch
 - CmdOption, 35
- isNativeMode
 - DiskXFile, 43
- isOption
 - CmdOption, 35
- isText
 - CmdOption, 35
- its_a_pipe
 - DiskXFile, 44
- LOG_2_PI
 - log_add.h, 266
- LOG_ONE
 - log_add.h, 266
- LOG_ZERO
 - log_add.h, 266
- lambda
 - TDTrainer, 132
- lastAction
 - QLearningAgent, 101
 - Qualia, 109
 - TupleDataSet, 137
- lastLeak
 - StaticAllocator, 121
- lastObservation
 - QLearningAgent, 101
 - RewardEnvironment, 115
 - TupleDataSet, 137
- learningRate
 - NeuralNetwork, 76
- linear
 - NeuralNetwork::Layer, 58
- linkDbCallback
 - MapperConnector, 61
- load
 - Agent, 17
 - BoolCmdOption, 28
 - CmdLine, 33
 - CmdOption, 35
 - Function, 51
 - GradientFunction, 54
 - IntCmdOption, 57
 - NeuralNetwork, 74
 - QFunction, 98
 - QLearningAgent, 101
 - RealCmdOption, 111
 - StringCmdOption, 128
- loadData
 - Action, 13
 - Observation, 78
 - RLObservation, 116
- locked
 - OscAgent, 82
 - OscEnvironment, 88
- log1p
 - log_add.h, 266
- log_add.cpp
 - logAdd, 265
 - logSub, 265
- log_add.h
 - LOG_2_PI, 266
 - LOG_ONE, 266
 - LOG_ZERO, 266
 - log1p, 266
 - logAdd, 266
 - logSub, 266
- logAdd
 - log_add.cpp, 265
 - log_add.h, 266
- logSub
 - log_add.cpp, 265
 - log_add.h, 266
- logout
 - MapperConnector, 61
- lowByte
 - Arduino.h, 145
- makeWord
 - WMath.cpp, 180
- malloc
 - Alloc, 18
 - Allocator, 19
 - StaticAllocator, 121
- map
 - WMath.cpp, 180
- map.cpp
 - mapReal, 267
- map.h
 - mapReal, 268
- mapReal
 - map.cpp, 267
 - map.h, 268
- MapperBasicEnvironment, 58
 - ~MapperBasicEnvironment, 59
 - actionDim, 60
 - addSignals, 59
 - currentObservation, 60
 - MapperBasicEnvironment, 59
 - MapperBasicEnvironment, 59
 - readInputs, 59
 - writeOutputs, 59
- MapperConnector, 60
 - ~MapperConnector, 61

- addInput, 61
- addOutput, 61
- admin, 62
- autoConnect, 62
- createConnections, 61
- db, 62
- dev, 62
- devDbCallback, 61
- deviceName, 62
- init, 61
- initialPort, 62
- inputData, 62
- linkDbCallback, 61
- logout, 61
- MapperConnector, 61
- MapperConnector, 61
- mon, 62
- nLinked, 62
- outputData, 62
- peerDeviceName, 62
- readInput, 61
- sendAllOutputs, 61
- SignalDataMap, 61
- updateInput, 61
- waitForBlockingInputs, 61
- writeOutput, 61, 62
- MapperConnector::SignalData, 119
 - ~SignalData, 119
 - data, 119
 - flag, 119
 - isBlocking, 119
 - n, 119
 - sig, 119
 - SignalData, 119
- MapperEnvironment, 62
 - ~MapperEnvironment, 64
 - addSignals, 64
 - connector, 64
 - init, 64
 - MapperEnvironment, 64
 - MapperEnvironment, 64
 - readInputs, 64
 - start, 64
 - step, 64
 - writeOutputs, 64
- master_switch
 - CmdLine, 33
- max
 - Arduino.h, 145
- maxExamples
 - BatchRLTrainer, 22
- maxTrainingIterationsPerEpisode
 - BatchRLTrainer, 22
- MemoryDataSet, 65
 - ~MemoryDataSet, 66
 - data, 66
 - MemoryDataSet, 66
 - MemoryDataSet, 66
- setExample, 66
- micros
 - Arduino.h, 146
- millis
 - Arduino.h, 146
- min
 - Arduino.h, 145
- mon
 - MapperConnector, 62
- MovingAverage, 66
 - _alpha, 67
 - _value, 67
 - get, 67
 - MovingAverage, 66
 - MovingAverage, 66
 - reset, 67
 - update, 67
- MultiAgent, 67
 - ~MultiAgent, 68
 - agents, 69
 - appendAgentAction, 69
 - combineAgentActions, 69
 - end, 69
 - extractAgentObservation, 69
 - init, 69
 - MultiAgent, 68
 - MultiAgent, 68
 - nAgents, 69
 - ownsAgents, 70
 - start, 69
 - step, 69
- MultiDataSet, 70
 - ~MultiDataSet, 71
 - _dataSets, 71
 - _nDataSets, 71
 - init, 71
 - MultiDataSet, 71
 - MultiDataSet, 71
 - reset, 71
 - setExample, 71
- mutate
 - BinaryChromosome, 24
 - Chromosome, 30
- mutateFlip
 - BinaryChromosome, 24
- Mutator
 - Chromosome.h, 215
- mutator
 - BinaryChromosomeProperties, 27
- n
 - MapperConnector::SignalData, 119
 - NeuralNetwork::Layer, 58
- n_associated_files
 - CmdLine, 33
- n_cmd_options
 - CmdLine, 33
- n_master_switches
 - CmdLine, 33

- nActions
 - Action, 13
 - ActionProperties, 15
- nAgents
 - MultiAgent, 69
- nConflated
 - Action, 13
 - ActionProperties, 15
- nEpisodes
 - Qualia, 109
 - Trainer, 135
- nExamples
 - DataSet, 38
- nGenes
 - BinaryChromosomeProperties, 26
- nHidden
 - NeuralNetwork, 74
- nInputs
 - Function, 51
 - NeuralNetwork, 75
 - QFunction, 98
- nLeaks
 - StaticAllocator, 121
- nLinked
 - MapperConnector, 62
- NO_SKIP_CHAR
 - Stream.cpp, 179
- nOutputs
 - Function, 51
 - NeuralNetwork, 75
 - QFunction, 98
- nParams
 - GradientFunction, 54
 - NeuralNetwork, 75
 - QFunction, 98
- nSteps
 - Qualia, 109
- name
 - CmdOption, 36
- needsSave
 - CmdOption, 36
- NeuralNetwork, 72
 - ~NeuralNetwork, 74
 - _allocateLayer, 74
 - _backpropagateLayer, 74
 - _deallocate, 74
 - _deallocateLayer, 74
 - _learningRateDiv, 76
 - _nParams, 76
 - _propagateLayer, 74
 - backpropagate, 74
 - decreaseConstant, 76
 - getCurrentLearningRate, 74
 - getOutput, 74
 - getOutputs, 74
 - hiddenLayer, 76
 - init, 74
 - inputLayer, 76
 - learningRate, 76
 - load, 74
 - nHidden, 74
 - nInputs, 75
 - nOutputs, 75
 - nParams, 75
 - NeuralNetwork, 74
 - NeuralNetwork, 74
 - outputLayer, 76
 - propagate, 75
 - save, 75
 - setInput, 75
 - setInputs, 75
 - update, 75
 - weightDecay, 76
- NeuralNetwork::Layer, 57
 - dWeight, 58
 - error, 58
 - linear, 58
 - n, 58
 - output, 58
 - weight, 58
- next
 - Action, 13
- OCT
 - Print.h, 177
- Observation, 76
 - ~Observation, 78
 - _dim, 78
 - copyFrom, 78
 - dim, 78
 - loadData, 78
 - Observation, 78
 - observations, 78
 - saveData, 78
 - terminal, 78
- observation
 - ObservationAction, 79
 - TupleDataSet, 137
- Observation.h
 - observation_t, 208
- observation_t
 - Observation.h, 208
- ObservationAction, 79
 - action, 79
 - observation, 79
- observationAction
 - Qualia, 109
- observationBuffer
 - OscEnvironment, 88
- observationBufferDim
 - OscAgent, 82
 - OscEnvironment, 88
- observationDim
 - FileExportEnvironment, 50
 - OscAgent, 82
 - OscEnvironment, 88
 - QFunction, 99

- QLearningAgent, 101
- TDTrainer, 132
- TupleDataSet, 137
- observationTypes
 - OscEnvironment, 88
- observations
 - Observation, 78
- offPolicy
 - TDTrainer, 133
- operator new
 - Allocator.cpp, 197
 - Allocator.h, 198
- operator+
 - WString.cpp, 181
- OscAgent, 79
 - ~OscAgent, 81
 - actionBuffer, 82
 - actionDim, 82
 - actionTypes, 82
 - getPath, 81
 - handlerInit, 81
 - handlerStartStep, 81
 - id, 82
 - init, 81
 - locked, 82
 - observationBufferDim, 82
 - observationDim, 82
 - OscAgent, 81
 - OscAgent, 81
 - readAction, 81
 - start, 81
 - step, 81
- OscBasicAgent, 82
 - ~OscBasicAgent, 83
 - currentAction, 83
 - OscBasicAgent, 83
 - OscBasicAgent, 83
 - readAction, 83
- OscBasicEnvironment, 84
 - ~OscBasicEnvironment, 85
 - currentObservation, 85
 - OscBasicEnvironment, 85
 - OscBasicEnvironment, 85
 - readObservation, 85
- OscEnvironment, 85
 - ~OscEnvironment, 87
 - actionDim, 88
 - getPath, 87
 - handlerInit, 87
 - handlerStartStep, 87
 - id, 88
 - init, 87
 - locked, 88
 - observationBuffer, 88
 - observationBufferDim, 88
 - observationDim, 88
 - observationTypes, 88
 - OscEnvironment, 87
 - OscEnvironment, 87
 - readObservation, 87
 - start, 87
 - step, 87
- OscManager, 88
 - ~OscManager, 88
 - _client, 89
 - _server, 89
 - client, 88
 - error, 89
 - initOsc, 89
 - OscManager, 88
 - OscManager, 88
 - repeatChar, 89
 - server, 89
- OscRLEnvironment, 89
 - ~OscRLEnvironment, 90
 - currentObservation, 90
 - OscRLEnvironment, 90
 - OscRLEnvironment, 90
 - readObservation, 90
- output
 - NeuralNetwork::Layer, 58
- outputData
 - MapperConnector, 62
- outputLayer
 - NeuralNetwork, 76
- ownsAgents
 - MultiAgent, 70
- PARSE_TIMEOUT
 - Stream.cpp, 179
- PI
 - Arduino.h, 145
- parseFloat
 - Stream, 125
- parseInt
 - Stream, 125
- peek
 - Stream, 125
- peekNextDigit
 - Stream, 125
- peerDeviceName
 - MapperConnector, 62
- Policy, 91
 - ~Policy, 92
 - agent, 92
 - chooseAction, 92
 - init, 92
 - Policy, 92
 - setAgent, 92
- policy
 - QLearningAgent, 101
- Print, 92
 - clearWriteError, 93
 - getWriteError, 93
 - Print, 93
 - print, 93, 94
 - printFloat, 94

- printNumber, [94](#)
 - println, [94](#)
 - setWriteError, [94](#)
 - write, [94](#)
 - write_error, [94](#)
- print
 - Print, [93](#), [94](#)
- Print.h
 - BIN, [177](#)
 - DEC, [177](#)
 - HEX, [177](#)
 - OCT, [177](#)
- printFloat
 - Print, [94](#)
- printNumber
 - Print, [94](#)
- printTo
 - Printable, [95](#)
- printValue
 - CmdOption, [36](#)
 - IntCmdOption, [57](#)
 - RealCmdOption, [111](#)
 - StringCmdOption, [128](#)
- Printable, [95](#)
 - printTo, [95](#)
- printf
 - DiskXFile, [43](#)
- println
 - Print, [94](#)
- program_name
 - CmdLine, [33](#)
- propagate
 - Function, [52](#)
 - NeuralNetwork, [75](#)
 - QFunction, [98](#)
- properties
 - Action, [14](#)
- ptr
 - BoolCmdOption, [29](#)
 - IntCmdOption, [57](#)
 - RealCmdOption, [111](#)
 - StringCmdOption, [128](#)
- Q_ASSERT_ERROR
 - error.h, [203](#)
- Q_ASSERT_NOTICE
 - error.h, [203](#)
- Q_ASSERT_WARNING
 - error.h, [203](#)
- Q_ERROR
 - error.h, [203](#)
- Q_MESSAGE
 - error.h, [203](#)
- Q_NOTICE
 - error.h, [203](#)
- Q_PRINT
 - error.h, [203](#)
- Q_WARNING
 - error.h, [203](#)
- QFunction, [95](#)
 - ~QFunction, [97](#)
 - actionProperties, [99](#)
 - backpropagate, [97](#)
 - function, [99](#)
 - getMaxAction, [97](#)
 - getOutput, [97](#)
 - getValue, [97](#)
 - init, [98](#)
 - input, [99](#)
 - load, [98](#)
 - nInputs, [98](#)
 - nOutputs, [98](#)
 - nParams, [98](#)
 - observationDim, [99](#)
 - propagate, [98](#)
 - QFunction, [97](#)
 - QFunction, [97](#)
 - save, [98](#)
 - setInput, [98](#)
 - update, [98](#)
- qFunction
 - QLearningAgent, [101](#)
- QLearningAgent, [99](#)
 - ~QLearningAgent, [100](#)
 - currentAction, [101](#)
 - end, [100](#)
 - init, [100](#)
 - isLearning, [101](#)
 - lastAction, [101](#)
 - lastObservation, [101](#)
 - load, [101](#)
 - observationDim, [101](#)
 - policy, [101](#)
 - qFunction, [101](#)
 - QLearningAgent, [100](#)
 - QLearningAgent, [100](#)
 - save, [101](#)
 - start, [101](#)
 - step, [101](#)
 - trainer, [101](#)
- QLearningEDecreasingPolicy, [102](#)
 - _epsilonDiv, [104](#)
 - chooseAction, [103](#)
 - decreaseConstant, [104](#)
 - getCurrentEpsilon, [103](#)
 - init, [104](#)
 - QLearningEDecreasingPolicy, [103](#)
 - QLearningEDecreasingPolicy, [103](#)
- QLearningEGreedyPolicy, [104](#)
 - ~QLearningEGreedyPolicy, [105](#)
 - chooseAction, [105](#)
 - epsilon, [106](#)
 - QLearningEGreedyPolicy, [105](#)
 - QLearningEGreedyPolicy, [105](#)
- QLearningSoftmaxPolicy, [106](#)
 - ~QLearningSoftmaxPolicy, [107](#)
 - chooseAction, [107](#)

- epsilon, 107
- QLearningSoftmaxPolicy, 107
- QLearningSoftmaxPolicy, 107
- temperature, 107
- Qualia, 108
 - ~Qualia, 109
 - agent, 109
 - environment, 109
 - episode, 109
 - init, 109
 - lastAction, 109
 - nEpisodes, 109
 - nSteps, 109
 - observationAction, 109
 - Qualia, 109
 - start, 109
 - step, 109
- RAD_TO_DEG
 - Arduino.h, 145
- RANDOM_MAX
 - common.h, 199
- REAL_FORMAT
 - DiskXFileDataSet.cpp, 188
- RLObservation, 115
 - copyFrom, 116
 - loadData, 116
 - RLObservation, 116
 - reward, 117
 - RLObservation, 116
 - saveData, 116
- RLQualia, 117
 - init, 118
 - RLQualia, 118
 - RLQualia, 118
 - start, 118
 - step, 118
 - totalReward, 119
- radians
 - Arduino.h, 145
- random
 - ActionProperties, 15
 - WMath.cpp, 180
- random.cpp
 - randomUniform, 269
- random.h
 - randomUniform, 270
- random_dist.cpp
 - bernouilli, 271
 - cauchy, 271
 - exponential, 271
 - geometric, 271
- random_dist.h
 - bernouilli, 272
 - cauchy, 272
 - exponential, 273
 - geometric, 273
- random_normal.cpp
 - randomNormal, 273
- random_normal.h
 - randomNormal, 274
- random_shuffle.cpp
 - randomShuffle, 189
 - randomShuffledIndices, 189
- random_shuffle.h
 - randomShuffle, 190
 - randomShuffledIndices, 190
- randomNormal
 - random_normal.cpp, 273
 - random_normal.h, 274
- randomSeed
 - WMath.cpp, 180
- randomShuffle
 - random_shuffle.cpp, 189
 - random_shuffle.h, 190
- randomShuffledIndices
 - random_shuffle.cpp, 189
 - random_shuffle.h, 190
- randomUniform
 - random.cpp, 269
 - random.h, 270
- read
 - BoolCmdOption, 28
 - CmdLine, 33
 - CmdOption, 36
 - DiskXFile, 43
 - IntCmdOption, 57
 - RealCmdOption, 111
 - Stream, 126
 - StringCmdOption, 128
 - XFile, 139
- readAction
 - OscAgent, 81
 - OscBasicAgent, 83
- readBytes
 - Stream, 126
- readBytesUntil
 - Stream, 126
- readInput
 - MapperConnector, 61
- readInputs
 - MapperBasicEnvironment, 59
 - MapperEnvironment, 64
- readObservation
 - OscBasicEnvironment, 85
 - OscEnvironment, 87
 - OscRLEnvironment, 90
- readString
 - Stream, 126
- readStringUntil
 - Stream, 126
- real
 - common.h, 199
- RealCmdOption, 110
 - ~RealCmdOption, 111
 - init_value, 111
 - initValue, 111

- load, 111
- printValue, 111
- ptr, 111
- read, 111
- RealCmdOption, 111
- RealCmdOption, 111
- save, 111
- realloc
 - Alloc, 18
 - Allocator, 19
 - StaticAllocator, 121
- repeatChar
 - OscManager, 89
- reset
 - Action, 13
 - DataSet, 38
 - DiskXFileDataSet, 46
 - MovingAverage, 67
 - MultiDataSet, 71
 - SubDataSet, 130
 - TupleDataSet, 137
 - XFileDataSet, 141
- reverseMemory
 - DiskXFile, 43
- Reward, 112
 - ~Reward, 112
 - environment, 113
 - Reward, 112
 - reward, 112
 - setEnvironment, 113
- reward
 - Reward, 112
 - RewardEnvironment, 115
 - RLObservation, 117
- RewardEnvironment, 113
 - ~RewardEnvironment, 114
 - doAction, 115
 - doStart, 115
 - lastObservation, 115
 - reward, 115
 - RewardEnvironment, 114
 - RewardEnvironment, 114
 - start, 115
 - step, 115
- rewind
 - DiskXFile, 43
 - XFile, 139
- round
 - Arduino.h, 145
- SEEK_CUR
 - XFile.h, 212
- SEEK_END
 - XFile.h, 212
- SEEK_SET
 - XFile.h, 212
- save
 - Agent, 17
 - BoolCmdOption, 28
 - CmdLine, 33
 - CmdOption, 36
 - Function, 52
 - GradientFunction, 54
 - IntCmdOption, 57
 - NeuralNetwork, 75
 - QFunction, 98
 - QLearningAgent, 101
 - RealCmdOption, 111
 - StringCmdOption, 128
- saveData
 - Action, 13
 - Observation, 78
 - RLObservation, 116
- scanf
 - DiskXFile, 43
- seek
 - DiskXFile, 44
 - XFile, 139
- sendAllOutputs
 - MapperConnector, 61
- server
 - OscManager, 89
- setAgent
 - Policy, 92
- setBigEndianMode
 - DiskXFile, 44
- setConflated
 - Action, 13
- setEnvironment
 - Reward, 113
- setExample
 - DataSet, 38
 - DiskXFileDataSet, 46
 - MemoryDataSet, 66
 - MultiDataSet, 71
 - SubDataSet, 130
 - TupleDataSet, 137
 - XFileDataSet, 142
- setGeneValue
 - BinaryChromosome, 24
- setInput
 - Function, 52
 - NeuralNetwork, 75
 - QFunction, 98
- setInputs
 - Function, 52
 - NeuralNetwork, 75
- setLittleEndianMode
 - DiskXFile, 44
- setNativeMode
 - DiskXFile, 44
- setTimeout
 - Stream, 126
- setWorkingDirectory
 - CmdLine, 33
- setWriteError
 - Print, 94

- SexualCrossover
 - Chromosome.h, 215
- sig
 - MapperConnector::SignalData, 119
- SignalData
 - MapperConnector::SignalData, 119
- SignalDataMap
 - MapperConnector, 61
- size
 - XFile, 139
- sq
 - Arduino.h, 145
- src/mainpage.dox, 143
- src/qualia/compat/Arduino.cpp, 143
- src/qualia/compat/Arduino.h, 143
- src/qualia/compat/Print.cpp, 175
- src/qualia/compat/Print.h, 176
- src/qualia/compat/Printable.h, 178
- src/qualia/compat/Stream.cpp, 178
- src/qualia/compat/Stream.h, 179
- src/qualia/compat/WMath.cpp, 180
- src/qualia/compat/WString.cpp, 180
- src/qualia/compat/WString.h, 181
- src/qualia/compat/binary.h, 146
- src/qualia/compat/new.cpp, 174
- src/qualia/compat/new.h, 175
- src/qualia/computer/CmdLine.cpp, 182
- src/qualia/computer/CmdLine.h, 183
- src/qualia/computer/CmdOption.cpp, 184
- src/qualia/computer/CmdOption.h, 184
- src/qualia/computer/DiskXFile.cpp, 186
- src/qualia/computer/DiskXFile.h, 186
- src/qualia/computer/DiskXFileDataSet.cpp, 187
- src/qualia/computer/DiskXFileDataSet.h, 188
- src/qualia/computer/SubDataSet.cpp, 193
- src/qualia/computer/SubDataSet.h, 193
- src/qualia/computer/random_shuffle.cpp, 189
- src/qualia/computer/random_shuffle.h, 189
- src/qualia/computer/string_utils.cpp, 191
- src/qualia/computer/string_utils.h, 192
- src/qualia/core/Action.cpp, 194
- src/qualia/core/Action.h, 195
- src/qualia/core/Agent.h, 196
- src/qualia/core/Allocator.cpp, 197
- src/qualia/core/Allocator.h, 197
- src/qualia/core/Environment.h, 200
- src/qualia/core/FileExportEnvironment.cpp, 204
- src/qualia/core/FileExportEnvironment.h, 204
- src/qualia/core/MultiAgent.cpp, 205
- src/qualia/core/MultiAgent.h, 205
- src/qualia/core/Observation.cpp, 206
- src/qualia/core/Observation.h, 207
- src/qualia/core/Qualia.cpp, 208
- src/qualia/core/Qualia.h, 208
- src/qualia/core/StaticAllocator.cpp, 209
- src/qualia/core/StaticAllocator.h, 210
- src/qualia/core/XFile.cpp, 211
- src/qualia/core/XFile.h, 211
- src/qualia/core/common.h, 198
- src/qualia/core/error.cpp, 200
- src/qualia/core/error.h, 201
- src/qualia/ga/BinaryChromosome.cpp, 212
- src/qualia/ga/BinaryChromosome.h, 213
- src/qualia/ga/Chromosome.h, 214
- src/qualia/learning/DataSet.cpp, 215
- src/qualia/learning/DataSet.h, 216
- src/qualia/learning/DataSetTrainer.cpp, 216
- src/qualia/learning/DataSetTrainer.h, 217
- src/qualia/learning/Function.cpp, 218
- src/qualia/learning/Function.h, 218
- src/qualia/learning/GradientFunction.h, 219
- src/qualia/learning/MemoryDataSet.cpp, 220
- src/qualia/learning/MemoryDataSet.h, 221
- src/qualia/learning/MultiDataSet.cpp, 222
- src/qualia/learning/MultiDataSet.h, 222
- src/qualia/learning/NeuralNetwork.cpp, 223
- src/qualia/learning/NeuralNetwork.h, 224
- src/qualia/learning/StochasticDataSetTrainer.cpp, 225
- src/qualia/learning/StochasticDataSetTrainer.h, 226
- src/qualia/learning/Trainer.cpp, 226
- src/qualia/learning/Trainer.h, 227
- src/qualia/learning/XFileDataSet.cpp, 228
- src/qualia/learning/XFileDataSet.h, 228
- src/qualia/plugins/mapper/MapperBasicEnvironment.-
cpp, 229
- src/qualia/plugins/mapper/MapperBasicEnvironment.h,
230
- src/qualia/plugins/mapper/MapperConnector.cpp, 231
- src/qualia/plugins/mapper/MapperConnector.h, 231
- src/qualia/plugins/mapper/MapperEnvironment.cpp, 232
- src/qualia/plugins/mapper/MapperEnvironment.h, 233
- src/qualia/plugins/osc/OscAgent.cpp, 234
- src/qualia/plugins/osc/OscAgent.h, 234
- src/qualia/plugins/osc/OscBasicAgent.cpp, 235
- src/qualia/plugins/osc/OscBasicAgent.h, 236
- src/qualia/plugins/osc/OscBasicEnvironment.cpp, 237
- src/qualia/plugins/osc/OscBasicEnvironment.h, 237
- src/qualia/plugins/osc/OscEnvironment.cpp, 238
- src/qualia/plugins/osc/OscEnvironment.h, 239
- src/qualia/plugins/osc/OscManager.cpp, 240
- src/qualia/plugins/osc/OscManager.h, 240
- src/qualia/plugins/osc/OscRLEnvironment.cpp, 241
- src/qualia/plugins/osc/OscRLEnvironment.h, 242
- src/qualia/rl/BatchRLTrainer.cpp, 243
- src/qualia/rl/BatchRLTrainer.h, 243
- src/qualia/rl/Policy.h, 244
- src/qualia/rl/QFunction.cpp, 245
- src/qualia/rl/QFunction.h, 246
- src/qualia/rl/QLearningAgent.cpp, 247
- src/qualia/rl/QLearningAgent.h, 248
- src/qualia/rl/QLearningEDecreasingPolicy.cpp, 249
- src/qualia/rl/QLearningEDecreasingPolicy.h, 249
- src/qualia/rl/QLearningEGreedyPolicy.cpp, 250
- src/qualia/rl/QLearningEGreedyPolicy.h, 250
- src/qualia/rl/QLearningSoftmaxPolicy.cpp, 251
- src/qualia/rl/QLearningSoftmaxPolicy.h, 252

- src/qualia/rl/RLObservation.cpp, 255
- src/qualia/rl/RLObservation.h, 256
- src/qualia/rl/RLQualia.cpp, 257
- src/qualia/rl/RLQualia.h, 257
- src/qualia/rl/Reward.h, 253
- src/qualia/rl/RewardEnvironment.cpp, 254
- src/qualia/rl/RewardEnvironment.h, 255
- src/qualia/rl/TDTrainer.cpp, 258
- src/qualia/rl/TDTrainer.h, 259
- src/qualia/rl/TupleDataSet.cpp, 260
- src/qualia/rl/TupleDataSet.h, 260
- src/qualia/util/MovingAverage.h, 268
- src/qualia/util/bitarray.cpp, 261
- src/qualia/util/bitarray.h, 262
- src/qualia/util/log_add.cpp, 264
- src/qualia/util/log_add.h, 265
- src/qualia/util/map.cpp, 267
- src/qualia/util/map.h, 267
- src/qualia/util/random.cpp, 269
- src/qualia/util/random.h, 269
- src/qualia/util/random_dist.cpp, 271
- src/qualia/util/random_dist.h, 272
- src/qualia/util/random_normal.cpp, 273
- src/qualia/util/random_normal.h, 274
- start
 - Agent, 17
 - Environment, 47
 - FileExportEnvironment, 49
 - MapperEnvironment, 64
 - MultiAgent, 69
 - OscAgent, 81
 - OscEnvironment, 87
 - QLearningAgent, 101
 - Qualia, 109
 - RewardEnvironment, 115
 - RLQualia, 118
- StaticAllocator, 120
 - buffer, 121
 - bufferIdx, 121
 - bufferSize, 121
 - free, 121
 - freeAll, 121
 - lastLeak, 121
 - malloc, 121
 - nLeaks, 121
 - realloc, 121
 - StaticAllocator, 121
 - StaticAllocator, 121
- step
 - Agent, 17
 - Environment, 47
 - FileExportEnvironment, 49
 - MapperEnvironment, 64
 - MultiAgent, 69
 - OscAgent, 81
 - OscEnvironment, 87
 - QLearningAgent, 101
 - Qualia, 109
 - RewardEnvironment, 115
 - RLQualia, 118
 - TDTrainer, 132
- StochasticDataSetTrainer, 122
 - ~StochasticDataSetTrainer, 123
 - _doTrainEpisode, 123
 - StochasticDataSetTrainer, 123
 - StochasticDataSetTrainer, 123
 - trainExample, 123
- stop
 - DataSetTrainer, 40
- strBaseName
 - string_utils.cpp, 191
 - string_utils.h, 192
- strConcat
 - string_utils.cpp, 191
 - string_utils.h, 193
- strRemoveSuffix
 - string_utils.cpp, 191
 - string_utils.h, 193
- Stream, 124
 - _startMillis, 126
 - _timeout, 126
 - available, 125
 - find, 125
 - findUntil, 125
 - flush, 125
 - parseFloat, 125
 - parseInt, 125
 - peek, 125
 - peekNextDigit, 125
 - read, 126
 - readBytes, 126
 - readBytesUntil, 126
 - readString, 126
 - readStringUntil, 126
 - setTimeout, 126
 - Stream, 125
 - timedPeek, 126
 - timedRead, 126
- Stream.cpp
 - NO_SKIP_CHAR, 179
 - PARSE_TIMEOUT, 179
- string_utils.cpp
 - strBaseName, 191
 - strConcat, 191
 - strRemoveSuffix, 191
- string_utils.h
 - strBaseName, 192
 - strConcat, 193
 - strRemoveSuffix, 193
- StringCmdOption, 126
 - ~StringCmdOption, 127
 - init_value, 128
 - initValue, 128
 - load, 128
 - printValue, 128
 - ptr, 128

- read, 128
- save, 128
- StringCmdOption, 127
- StringCmdOption, 127
- SubDataSet, 128
 - ~SubDataSet, 130
 - _dataSet, 130
 - _indices, 130
 - getShuffledDataSet, 130
 - getStandardDataSets, 130
 - init, 130
 - reset, 130
 - setExample, 130
 - SubDataSet, 130
 - SubDataSet, 130
- TDTrainer, 130
 - ~TDTrainer, 132
 - actionDim, 132
 - bufferAction, 132
 - eTraces, 132
 - gamma, 132
 - init, 132
 - lambda, 132
 - observationDim, 132
 - offPolicy, 133
 - step, 132
 - TDTrainer, 132
 - TDTrainer, 132
- TWO_PI
 - Arduino.h, 146
- taggedRead
 - XFile, 140
- taggedWrite
 - XFile, 140
- targets
 - BatchRLTrainer, 22
- tell
 - DiskXFile, 44
 - XFile, 140
- temperature
 - QLearningSoftmaxPolicy, 107
- terminal
 - Observation, 78
- text_info
 - CmdLine, 33
- timedPeek
 - Stream, 126
- timedRead
 - Stream, 126
- totalReward
 - RLQualia, 119
- train
 - DataSetTrainer, 40
- trainEpisode
 - DataSetTrainer, 40
- trainExample
 - StochasticDataSetTrainer, 123
- Trainer, 133
 - ~Trainer, 134
 - _function, 135
 - _nEpisodes, 135
 - init, 134
 - nEpisodes, 135
 - Trainer, 134
- trainer
 - QLearningAgent, 101
- true
 - Arduino.h, 145
- TupleDataSet, 135
 - ~TupleDataSet, 137
 - actionDim, 137
 - currentExampleIndex, 137
 - file, 137
 - init, 137
 - lastAction, 137
 - lastObservation, 137
 - observation, 137
 - observationDim, 137
 - reset, 137
 - setExample, 137
 - TupleDataSet, 137
 - tupleFromExample, 137
 - tupleToExample, 137
 - TupleDataSet, 137
- tupleFromExample
 - TupleDataSet, 137
- tupleToExample
 - TupleDataSet, 137
- type_name
 - CmdOption, 36
- undefined
 - Action, 13
- update
 - GradientFunction, 54
 - MovingAverage, 67
 - NeuralNetwork, 75
 - QFunction, 98
- updateInput
 - MapperConnector, 61
- WMath.cpp
 - makeWord, 180
 - map, 180
 - random, 180
 - randomSeed, 180
- WString.cpp
 - operator+, 181
- waitForBlockingInputs
 - MapperConnector, 61
- weight
 - NeuralNetwork::Layer, 58
- weightDecay
 - NeuralNetwork, 76
- weights
 - GradientFunction, 55
- word

- Arduino.h, [146](#)
- working_directory
 - CmdLine, [33](#)
- write
 - DiskXFile, [44](#)
 - Print, [94](#)
 - XFile, [140](#)
- write_error
 - Print, [94](#)
- write_log
 - CmdLine, [34](#)
- writeLog
 - CmdLine, [33](#)
- writeOutput
 - MapperConnector, [61](#), [62](#)
- writeOutputs
 - MapperBasicEnvironment, [59](#)
 - MapperEnvironment, [64](#)
- XFile, [138](#)
 - ~XFile, [139](#)
 - eof, [139](#)
 - flush, [139](#)
 - gets, [139](#)
 - read, [139](#)
 - rewind, [139](#)
 - seek, [139](#)
 - size, [139](#)
 - taggedRead, [140](#)
 - taggedWrite, [140](#)
 - tell, [140](#)
 - write, [140](#)
 - XFile, [139](#)
 - XFile, [139](#)
- XFile.h
 - SEEK_CUR, [212](#)
 - SEEK_END, [212](#)
 - SEEK_SET, [212](#)
- XFileDataSet, [140](#)
 - ~XFileDataSet, [141](#)
 - _currentExampleIndex, [142](#)
 - _file, [142](#)
 - init, [141](#)
 - reset, [141](#)
 - setExample, [142](#)
 - XFileDataSet, [141](#)
 - XFileDataSet, [141](#)