

RestaurantManager

Generated by Doxygen 1.8.13

Contents

1	Namespace Index	1
1.1	Namespace List	1
2	Hierarchical Index	3
2.1	Class Hierarchy	3
3	Class Index	5
3.1	Class List	5
4	Namespace Documentation	7
4.1	Serialization Namespace Reference	7
4.1.1	Detailed Description	7
4.1.2	Enumeration Type Documentation	8
4.1.2.1	Type	8
4.1.3	Function Documentation	8
4.1.3.1	register_models()	8
5	Class Documentation	9
5.1	App Class Reference	9
5.1.1	Detailed Description	10
5.1.2	Member Function Documentation	10
5.1.2.1	getClient()	10
5.1.2.2	getMainWindow()	11
5.1.2.3	getShiftWindow()	11
5.1.2.4	getWorkerWindow()	11

5.1.2.5	setMainWindow()	11
5.1.2.6	setShiftWindow()	11
5.1.2.7	setWorkerWindow()	12
5.2	C2S_Authorize Class Reference	12
5.2.1	Detailed Description	13
5.2.2	Constructor & Destructor Documentation	14
5.2.2.1	C2S_Authorize()	14
5.2.3	Member Function Documentation	14
5.2.3.1	deserialize()	14
5.2.3.2	getToken()	14
5.2.3.3	getType()	15
5.2.3.4	serialize()	15
5.2.3.5	setToken()	15
5.3	C2S_DeleteShift Class Reference	16
5.3.1	Detailed Description	17
5.3.2	Constructor & Destructor Documentation	17
5.3.2.1	C2S_DeleteShift()	17
5.3.3	Member Function Documentation	17
5.3.3.1	deserialize()	18
5.3.3.2	getShiftId()	18
5.3.3.3	getType()	18
5.3.3.4	serialize()	18
5.3.3.5	setShiftId()	19
5.4	C2S_DeleteWorker Class Reference	19
5.4.1	Detailed Description	21
5.4.2	Constructor & Destructor Documentation	21
5.4.2.1	C2S_DeleteWorker()	21
5.4.3	Member Function Documentation	21
5.4.3.1	deserialize()	21
5.4.3.2	getType()	22

5.4.3.3	getWorkerId()	22
5.4.3.4	serialize()	22
5.4.3.5	setWorkerId()	23
5.5	C2S_GetShiftsByDay Class Reference	23
5.5.1	Detailed Description	24
5.5.2	Constructor & Destructor Documentation	25
5.5.2.1	C2S_GetShiftsByDay()	25
5.5.3	Member Function Documentation	25
5.5.3.1	deserialize()	25
5.5.3.2	getDate() [1/2]	25
5.5.3.3	getDate() [2/2]	26
5.5.3.4	getType()	26
5.5.3.5	serialize()	26
5.5.3.6	setDate()	26
5.6	C2S_GetWorkers Class Reference	27
5.6.1	Detailed Description	28
5.6.2	Member Function Documentation	28
5.6.2.1	getType()	28
5.7	C2S_InsertShift Class Reference	29
5.7.1	Detailed Description	31
5.7.2	Constructor & Destructor Documentation	31
5.7.2.1	C2S_InsertShift()	31
5.7.3	Member Function Documentation	31
5.7.3.1	deserialize()	31
5.7.3.2	getShift() [1/2]	32
5.7.3.3	getShift() [2/2]	32
5.7.3.4	getType()	32
5.7.3.5	isModifyExisting()	32
5.7.3.6	serialize()	32
5.7.3.7	setModifyExisting()	33

5.7.3.8	setShift()	33
5.8	C2S_InsertWorker Class Reference	33
5.8.1	Detailed Description	35
5.8.2	Constructor & Destructor Documentation	35
5.8.2.1	C2S_InsertWorker()	35
5.8.3	Member Function Documentation	35
5.8.3.1	deserialize()	35
5.8.3.2	getType()	36
5.8.3.3	getWorker() [1/2]	36
5.8.3.4	getWorker() [2/2]	36
5.8.3.5	isModifyExisting()	37
5.8.3.6	serialize()	37
5.8.3.7	setModifyExisting()	37
5.8.3.8	setWorker()	37
5.9	Connection Class Reference	38
5.9.1	Detailed Description	39
5.9.2	Constructor & Destructor Documentation	39
5.9.2.1	Connection()	39
5.9.3	Member Function Documentation	40
5.9.3.1	cleanup()	40
5.9.3.2	closeError()	40
5.9.3.3	connect() [1/2]	40
5.9.3.4	connect() [2/2]	41
5.9.3.5	isAlive()	41
5.9.3.6	readAsync()	41
5.9.3.7	readSync()	42
5.9.3.8	readSyncInternal()	42
5.9.3.9	receiveFromSocket()	42
5.9.3.10	sendBuffer()	42
5.9.3.11	setReadingAsync()	43

5.9.3.12	writeSync()	43
5.10	ConnectionBase Class Reference	43
5.10.1	Detailed Description	46
5.10.2	Constructor & Destructor Documentation	46
5.10.2.1	ConnectionBase()	46
5.10.3	Member Function Documentation	46
5.10.3.1	assertConnected()	46
5.10.3.2	cleanup()	46
5.10.3.3	close()	47
5.10.3.4	closeError()	47
5.10.3.5	connect() [1/2]	47
5.10.3.6	connect() [2/2]	47
5.10.3.7	getData() [1/2]	48
5.10.3.8	getData() [2/2]	48
5.10.3.9	getId()	48
5.10.3.10	isAlive()	49
5.10.3.11	isReadingAsync()	49
5.10.3.12	newRequestId()	49
5.10.3.13	onConnected()	49
5.10.3.14	onDisconnected()	49
5.10.3.15	onPayloadReceived()	50
5.10.3.16	onPayloadSent()	50
5.10.3.17	readAsync()	50
5.10.3.18	readSync()	51
5.10.3.19	setReadingAsync()	51
5.10.3.20	subscribe()	51
5.10.3.21	unsubscribe()	51
5.10.3.22	writeReplySync() [1/2]	52
5.10.3.23	writeReplySync() [2/2]	52
5.10.3.24	writeRequestSync()	52

5.10.3.25 writeSync()	53
5.11 ConnectionEventHandler Class Reference	53
5.11.1 Detailed Description	54
5.11.2 Member Function Documentation	54
5.11.2.1 onConnected()	55
5.11.2.2 onDisconnected()	56
5.11.2.3 onPayloadSent()	56
5.12 ConnectionObserver Class Reference	56
5.12.1 Detailed Description	57
5.12.2 Member Function Documentation	57
5.12.2.1 addHandler() [1/2]	58
5.12.2.2 addHandler() [2/2]	58
5.12.2.3 onConnected()	58
5.12.2.4 onDisconnected()	59
5.12.2.5 onPayloadReceived()	59
5.12.2.6 onPayloadSent()	59
5.12.2.7 removeHandlers()	60
5.13 DataBag Class Reference	60
5.13.1 Detailed Description	61
5.13.2 Member Function Documentation	61
5.13.2.1 clear()	61
5.13.2.2 get() [1/2]	61
5.13.2.3 get() [2/2]	61
5.13.2.4 put()	62
5.13.2.5 remove()	62
5.14 Date Class Reference	63
5.14.1 Detailed Description	64
5.14.2 Constructor & Destructor Documentation	64
5.14.2.1 Date() [1/2]	65
5.14.2.2 Date() [2/2]	65

5.14.3	Member Function Documentation	65
5.14.3.1	deserialize()	65
5.14.3.2	getDay()	66
5.14.3.3	getMonth()	66
5.14.3.4	getType()	66
5.14.3.5	getYear()	66
5.14.3.6	serialize()	66
5.14.3.7	setDay()	67
5.14.3.8	setMonth()	67
5.14.3.9	setYear()	67
5.15	DateTime Class Reference	68
5.15.1	Detailed Description	70
5.15.2	Constructor & Destructor Documentation	70
5.15.2.1	DateTime() [1/4]	70
5.15.2.2	DateTime() [2/4]	70
5.15.2.3	DateTime() [3/4]	70
5.15.2.4	DateTime() [4/4]	71
5.15.3	Member Function Documentation	71
5.15.3.1	deserialize()	71
5.15.3.2	getDate()	71
5.15.3.3	getHour()	72
5.15.3.4	getMinute()	72
5.15.3.5	getSecond()	72
5.15.3.6	getType()	72
5.15.3.7	serialize()	72
5.15.3.8	setHour()	73
5.15.3.9	setMinute()	73
5.15.3.10	setSecond()	73
5.16	MainWindow Class Reference	74
5.16.1	Detailed Description	75

5.16.2	Constructor & Destructor Documentation	75
5.16.2.1	MainWindow()	75
5.16.3	Member Function Documentation	75
5.16.3.1	getCurrentDate()	75
5.16.3.2	resizeGrid()	76
5.17	memory_buf Class Reference	76
5.17.1	Detailed Description	78
5.17.2	Constructor & Destructor Documentation	78
5.17.2.1	memory_buf()	78
5.17.3	Member Function Documentation	78
5.17.3.1	setData()	78
5.17.3.2	setLength()	78
5.18	memory_stream Class Reference	79
5.18.1	Detailed Description	80
5.19	Ping Class Reference	80
5.19.1	Detailed Description	81
5.19.2	Member Function Documentation	81
5.19.2.1	getType()	81
5.20	PingReply Class Reference	81
5.20.1	Detailed Description	82
5.20.2	Member Function Documentation	82
5.20.2.1	getType()	82
5.21	PingService Class Reference	83
5.21.1	Detailed Description	84
5.21.2	Constructor & Destructor Documentation	84
5.21.2.1	PingService()	84
5.21.3	Member Function Documentation	84
5.21.3.1	getInstance()	84
5.21.3.2	subscribe()	84
5.21.3.3	unsubscribe()	85

5.21.3.4	worker()	85
5.22	RestaurantClient Class Reference	85
5.22.1	Detailed Description	88
5.22.2	Member Function Documentation	88
5.22.2.1	authorize()	88
5.22.2.2	queryDeleteShift()	89
5.22.2.3	queryDeleteWorker()	89
5.22.2.4	queryInsertShift()	89
5.22.2.5	queryInsertWorker()	89
5.22.2.6	queryShiftsByDay()	90
5.22.2.7	queryUpdateShift()	90
5.22.2.8	queryUpdateWorker()	90
5.22.2.9	queryWorkers()	91
5.23	RestaurantManager Class Reference	91
5.23.1	Detailed Description	93
5.23.2	Member Function Documentation	93
5.23.2.1	deleteShift()	93
5.23.2.2	deleteWorker()	94
5.23.2.3	deserialize()	94
5.23.2.4	getAccessTokens()	95
5.23.2.5	getShifts()	95
5.23.2.6	getShiftsByDay()	95
5.23.2.7	getType()	95
5.23.2.8	getWorkers()	96
5.23.2.9	insertShift()	96
5.23.2.10	insertWorker()	96
5.23.2.11	newIdentityId()	97
5.23.2.12	onPayloadReceived()	97
5.23.2.13	serialize()	97
5.23.2.14	verifyPermission()	98

5.23.2.15	verifyShift()	98
5.24	S2C_AuthorizeReply Class Reference	99
5.24.1	Detailed Description	100
5.24.2	Constructor & Destructor Documentation	100
5.24.2.1	S2C_AuthorizeReply() [1/2]	100
5.24.2.2	S2C_AuthorizeReply() [2/2]	101
5.24.3	Member Function Documentation	101
5.24.3.1	deserialize()	101
5.24.3.2	getPermissions()	101
5.24.3.3	getType()	102
5.24.3.4	serialize()	102
5.24.3.5	setPermissions()	102
5.25	S2C_ClientSync Class Reference	103
5.25.1	Detailed Description	104
5.25.2	Member Function Documentation	104
5.25.2.1	deserialize()	104
5.25.2.2	getChangedShifts() [1/2]	105
5.25.2.3	getChangedShifts() [2/2]	105
5.25.2.4	getChangedWorkers() [1/2]	105
5.25.2.5	getChangedWorkers() [2/2]	105
5.25.2.6	getRemovedShifts() [1/2]	106
5.25.2.7	getRemovedShifts() [2/2]	106
5.25.2.8	getRemovedWorkers() [1/2]	106
5.25.2.9	getRemovedWorkers() [2/2]	106
5.25.2.10	getType()	107
5.25.2.11	serialize()	107
5.26	S2C_DeleteShiftReply Class Reference	108
5.26.1	Detailed Description	109
5.26.2	Constructor & Destructor Documentation	110
5.26.2.1	S2C_DeleteShiftReply()	110

5.26.3	Member Function Documentation	110
5.26.3.1	deserialize()	110
5.26.3.2	getId()	110
5.26.3.3	getType()	111
5.26.3.4	serialize()	111
5.26.3.5	setId()	111
5.27	S2C_DeleteWorkerReply Class Reference	112
5.27.1	Detailed Description	113
5.27.2	Constructor & Destructor Documentation	113
5.27.2.1	S2C_DeleteWorkerReply()	113
5.27.3	Member Function Documentation	114
5.27.3.1	deserialize()	114
5.27.3.2	getId()	114
5.27.3.3	getType()	114
5.27.3.4	serialize()	114
5.27.3.5	setId()	115
5.28	S2C_GetShiftsReply Class Reference	115
5.28.1	Detailed Description	117
5.28.2	Constructor & Destructor Documentation	117
5.28.2.1	S2C_GetShiftsReply()	117
5.28.3	Member Function Documentation	118
5.28.3.1	deserialize()	118
5.28.3.2	getDate() [1/2]	118
5.28.3.3	getDate() [2/2]	118
5.28.3.4	getShifts() [1/2]	119
5.28.3.5	getShifts() [2/2]	119
5.28.3.6	getType()	119
5.28.3.7	serialize()	119
5.28.3.8	setShifts()	120
5.29	S2C_GetWorkersReply Class Reference	120

5.29.1 Detailed Description	122
5.29.2 Constructor & Destructor Documentation	122
5.29.2.1 S2C_GetWorkersReply()	122
5.29.3 Member Function Documentation	122
5.29.3.1 deserialize()	122
5.29.3.2 getType()	122
5.29.3.3 getWorkers() [1/2]	123
5.29.3.4 getWorkers() [2/2]	123
5.29.3.5 serialize()	123
5.29.3.6 setWorkers()	124
5.30 S2C_InsertShiftReply Class Reference	124
5.30.1 Detailed Description	126
5.30.2 Constructor & Destructor Documentation	126
5.30.2.1 S2C_InsertShiftReply()	126
5.30.3 Member Function Documentation	126
5.30.3.1 deserialize()	126
5.30.3.2 getShift() [1/2]	127
5.30.3.3 getShift() [2/2]	127
5.30.3.4 getType()	127
5.30.3.5 serialize()	127
5.30.3.6 setShift()	128
5.31 S2C_InsertWorkerReply Class Reference	128
5.31.1 Detailed Description	130
5.31.2 Constructor & Destructor Documentation	130
5.31.2.1 S2C_InsertWorkerReply()	130
5.31.3 Member Function Documentation	131
5.31.3.1 deserialize()	131
5.31.3.2 getType()	131
5.31.3.3 getWorker() [1/2]	131
5.31.3.4 getWorker() [2/2]	132

5.31.3.5	<code>serialize()</code>	132
5.31.3.6	<code>setWorker()</code>	132
5.32	Serialization::Serializable Class Reference	133
5.32.1	Detailed Description	133
5.32.2	Member Function Documentation	134
5.32.2.1	<code>deserialize()</code>	134
5.32.2.2	<code>getType()</code>	134
5.32.2.3	<code>serialize()</code>	134
5.33	Server Class Reference	135
5.33.1	Member Function Documentation	136
5.33.1.1	<code>clients()</code>	137
5.33.1.2	<code>deleteConnectionAsync()</code>	137
5.33.1.3	<code>isRunning()</code>	137
5.33.1.4	<code>join()</code>	137
5.33.1.5	<code>listenWorker()</code>	137
5.33.1.6	<code>onConnected()</code>	138
5.33.1.7	<code>onDisconnected()</code>	138
5.33.1.8	<code>onPayloadReceived()</code>	138
5.33.1.9	<code>onPayloadSent()</code>	139
5.33.1.10	<code>prepareClient()</code>	139
5.33.1.11	<code>shouldAcceptSocket()</code>	139
5.33.1.12	<code>start()</code>	140
5.33.1.13	<code>stop()</code>	140
5.33.1.14	<code>subscribe()</code>	140
5.33.1.15	<code>unsubscribe()</code>	140
5.33.1.16	<code>writeToAll()</code>	141
5.34	ServerObserver Class Reference	141
5.34.1	Detailed Description	142
5.34.2	Member Function Documentation	142
5.34.2.1	<code>getServer()</code>	142

5.34.2.2	setServer()	143
5.35	Shift Class Reference	143
5.35.1	Detailed Description	145
5.35.2	Constructor & Destructor Documentation	145
5.35.2.1	Shift()	145
5.35.3	Member Function Documentation	146
5.35.3.1	deserialize()	146
5.35.3.2	getEndTime()	146
5.35.3.3	getId()	146
5.35.3.4	getJobName()	147
5.35.3.5	getStartTime() [1/2]	147
5.35.3.6	getStartTime() [2/2]	147
5.35.3.7	getType()	147
5.35.3.8	getWorkerId()	148
5.35.3.9	getWorkHours()	148
5.35.3.10	serialize()	148
5.35.3.11	setId()	148
5.35.3.12	setJobName()	149
5.35.3.13	setStartTime()	149
5.35.3.14	setWorkerId()	149
5.35.3.15	setWorkHours()	150
5.36	ShiftCellRenderer Class Reference	150
5.36.1	Detailed Description	152
5.36.2	Member Function Documentation	152
5.36.2.1	Draw()	152
5.36.2.2	GetBestSize()	152
5.36.2.3	getShift()	153
5.36.2.4	getWorker()	153
5.36.2.5	setShift()	153
5.36.2.6	setWorker()	153

5.37 ShiftEditorWindow Class Reference	154
5.37.1 Detailed Description	155
5.37.2 Constructor & Destructor Documentation	155
5.37.2.1 ShiftEditorWindow()	155
5.37.3 Member Function Documentation	155
5.37.3.1 getWorkerData()	155
5.38 ShiftWorker Class Reference	156
5.38.1 Detailed Description	157
5.38.2 Member Function Documentation	157
5.38.2.1 deserialize()	157
5.38.2.2 getFirstName()	158
5.38.2.3 getId()	158
5.38.2.4 getLastName()	158
5.38.2.5 getTitle()	159
5.38.2.6 getType()	159
5.38.2.7 serialize()	159
5.38.2.8 setFirstName()	159
5.38.2.9 setId()	160
5.38.2.10 setLastName()	160
5.38.2.11 setTitle()	160
5.38.2.12 toString()	161
5.39 TrackablePacket Class Reference	161
5.39.1 Detailed Description	162
5.39.2 Constructor & Destructor Documentation	162
5.39.2.1 TrackablePacket()	162
5.39.3 Member Function Documentation	163
5.39.3.1 deserialize()	163
5.39.3.2 getRequestId()	163
5.39.3.3 serialize()	163
5.39.3.4 setRequestId()	164

5.40 TransactionReply Class Reference	164
5.40.1 Detailed Description	166
5.40.2 Constructor & Destructor Documentation	166
5.40.2.1 TransactionReply() [1/3]	166
5.40.2.2 TransactionReply() [2/3]	166
5.40.2.3 TransactionReply() [3/3]	166
5.40.3 Member Function Documentation	167
5.40.3.1 deserialize()	167
5.40.3.2 getErrorMsg()	167
5.40.3.3 isSuccess()	167
5.40.3.4 serialize()	168
5.40.3.5 setErrorMsg()	168
5.40.3.6 setSuccess()	168
5.41 Serialization::TypeInfo Struct Reference	169
5.41.1 Detailed Description	169
5.42 WorkerEditorWindow Class Reference	169
5.42.1 Detailed Description	170
5.42.2 Constructor & Destructor Documentation	170
5.42.2.1 WorkerEditorWindow()	170
Index	173

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Serialization	
Namespace that contains all serialization logic	7

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

basic_streambuf	
memory_buf	76
ConnectionBase	43
Connection	38
ConnectionObserver	56
App	9
RestaurantClient	85
Server	135
ServerObserver	141
ConnectionEventHandler	53
RestaurantManager	91
DataBag	60
iostream	
memory_stream	79
PingService	83
Serialization::Serializable	133
Date	63
DateTime	68
Ping	80
PingReply	81
RestaurantManager	91
S2C_ClientSync	103
Shift	143
ShiftWorker	156
TrackablePacket	161
C2S_Authorize	12
C2S_DeleteShift	16
C2S_DeleteWorker	19
C2S_GetShiftsByDay	23
C2S_GetWorkers	27
C2S_InsertShift	29
C2S_InsertWorker	33
TransactionReply	164
S2C_AuthorizeReply	99

S2C_DeleteShiftReply	108
S2C_DeleteWorkerReply	112
S2C_GetShiftsReply	115
S2C_GetWorkersReply	120
S2C_InsertShiftReply	124
S2C_InsertWorkerReply	128
Serialization::TypeInfo	169
wxApp	
App	9
wxFrame	
MainWindow	74
ShiftEditorWindow	154
WorkerEditorWindow	169
wxGridCellStringRenderer	
ShiftCellRenderer	150

Chapter 3

Class Index

3.1 Class List

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

App	App class represents the base of everything Inherits from wxApp and ConnectionObserver classes	9
C2S_Authorize	Client-to-server packet that authorizes the client with a token	12
C2S_DeleteShift	Client-to-server packet that deletes a Shift by its id	16
C2S_DeleteWorker	Client-to-server packet that deletes a ShiftWorker by its id	19
C2S_GetShiftsByDay	Client-to-server request to list all Shift objects in a given date	23
C2S_GetWorkers	Client-to-server request to list all ShiftWorker objects	27
C2S_InsertShift	Client-to-server request to insert the given Shift object	29
C2S_InsertWorker	Client-to-server request to insert a ShiftWorker object	33
Connection	Main class that wraps a native socket into a managed TCP Client	38
ConnectionBase	Abstract class implementing basic functionality for wrapping a PSocket into a managed TCP Client	43
ConnectionEventHandler	Logging handlers	53
ConnectionObserver	Observer that allows listening to events of a Connection object	56
DataBag	Class that stores data associatively in string-keyed dictionary	60
Date	Serializable class representing a date (year, month, day)	63
DateTime	Serializable class extending date with additional (hour, minute, second) fields	68
MainWindow	MainWindow class represents the main window of the app Inherits from wxFrame class	74
memory_buf	Provide in-memory alternatives to iostream streams/buffers	76

memory_stream	Provides an in-memory input/output stream with std::iostream interface, using memory_buf as the internal buffer	79
Ping	Basic Ping packet that forces the other endpoint to send a PingReply	80
PingReply	Response to a Ping request	81
PingService	Singleton class to broadcast Ping packets on all bound connections	83
RestaurantClient	Client interface to access resources provided by the remote RestaurantManager	85
RestaurantManager	Class that binds to a Server and provides the RestaurantManager service	91
S2C_AuthorizeReply	Server-to-client reply to a token-based authorization request	99
S2C_ClientSync	Server-to-client broadcast packet that updates the client state	103
S2C_DeleteShiftReply	Server-to-client response to deleting a Shift object	108
S2C_DeleteWorkerReply	Server-to-client response to deleting a ShiftWorker object	112
S2C_GetShiftsReply	Server-to-client response that lists all Shift objects in the given day	115
S2C_GetWorkersReply	Server-to-client response that lists all ShiftWorker objects in the database	120
S2C_InsertShiftReply	Server-to-client response to insert a Shift object	124
S2C_InsertWorkerReply	Server-to-client response to insert a ShiftWorker object	128
Serialization::Serializable	Base-class for all serializable objects	133
Server	135
ServerObserver	Observer that allows listening to events of a Server object	141
Shift	Class that represents a Shift in the schedule	143
ShiftCellRenderer	ShiftCellRenderer class represents set of settings for beautiful UI layout Inherits from wxGridCellStringRenderer	150
ShiftEditorWindow	ShiftEditorWindow class represents a Shift Editor window Inherits from wxFrame class	154
ShiftWorker	Class representing a worker associated with a Shift	156
TrackablePacket	Base-class for tracking request-response packets by id	161
TransactionReply	Base-class for trackable transaction replies with success flags and error message	164
Serialization::TypeInfo	Struct to hold the object graph of serializable types	169
WorkerEditorWindow	WorkerEditorWindow class represents the Worker Editor window Inherits from the wxFrame class	169

Chapter 4

Namespace Documentation

4.1 Serialization Namespace Reference

Namespace that contains all serialization logic.

Classes

- class [Serializable](#)
Base-class for all serializable objects.
- struct [TypeInfo](#)
Struct to hold the object graph of serializable types.

Enumerations

- enum [Type](#) : uint8_t {
 Unknown = 0, **_Ping** = 1, **_PingReply** = 2, **_TrackablePacket**,
 _TransactionReply, **_RestaurantManager**, **_Date**, **_DateTime**,
 _Shift, **_ShiftWorker**, **_C2S_Authorize**, **_S2C_AuthorizeReply**,
 _C2S_GetShiftsByDay, **_S2C_GetShiftsReply**, **_C2S_InsertShift**, **_S2C_InsertShiftReply**,
 _C2S_DeleteShift, **_S2C_DeleteShiftReply**, **_C2S_GetWorkers**, **_S2C_GetWorkersReply**,
 _C2S_InsertWorker, **_S2C_InsertWorkerReply**, **_C2S_DeleteWorker**, **_S2C_DeleteWorkerReply**,
 _S2C_ClientSync }
Enumeration of all serializable type with their ids.

Functions

- void [register_models](#) ()
Registers all [Serializable](#) models.
- std::ostream & **operator**<< (std::ostream &dst, const [Serializable](#) &obj)
- std::istream & **operator**>> (std::istream &src, [Serializable](#) &obj)

4.1.1 Detailed Description

Namespace that contains all serialization logic.

4.1.2 Enumeration Type Documentation

4.1.2.1 Type

```
enum Serialization::Type : uint8_t [strong]
```

Enumeration of all serializable type with their ids.

4.1.3 Function Documentation

4.1.3.1 `register_models()`

```
void Serialization::register\_models ( ) [inline]
```

Registers all [Serializable](#) models.

Chapter 5

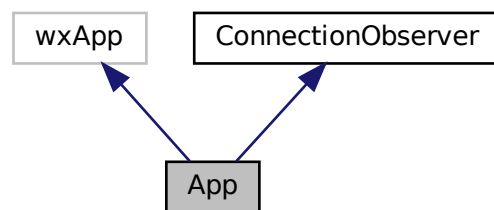
Class Documentation

5.1 App Class Reference

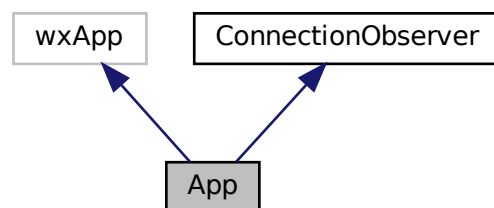
[App](#) class represents the base of everything Inherits from [wxApp](#) and [ConnectionObserver](#) classes.

```
#include <App.h>
```

Inheritance diagram for App:



Collaboration diagram for App:



Public Member Functions

- [App](#) ()
App class constructor.
- [~App](#) ()
App class destructor.
- bool [OnInit](#) () override
Method that initializes the main window.
- [RestaurantClient](#) & [getClient](#) ()
- [MainWindow](#) * [getMainWindow](#) () const
- [ShiftEditorWindow](#) * [getShiftWindow](#) () const
- [WorkerEditorWindow](#) * [getWorkerWindow](#) () const
- void [refreshUi](#) ()
Method which refreshes every window in the current context.
- void [setMainWindow](#) ([MainWindow](#) *wnd)
Method that sets Main Window pointer.
- void [setShiftWindow](#) ([ShiftEditorWindow](#) *wnd)
Method that sets Main Window pointer.
- void [setWorkerWindow](#) ([WorkerEditorWindow](#) *wnd)
Method that sets Main Window pointer.

Protected Member Functions

- void [onAuthorize](#) ([ConnectionBase](#) *connection, const [S2C_AuthorizeReply](#) &payload, size_t size)
Networking stuff.
- void [onDeleteShift](#) ([ConnectionBase](#) *connection, const [S2C_DeleteShiftReply](#) &payload, size_t size)
- void [onGetShiftsByDay](#) ([ConnectionBase](#) *connection, const [S2C_GetShiftsReply](#) &payload, size_t size)
- void [onGetWorkers](#) ([ConnectionBase](#) *connection, const [S2C_GetWorkersReply](#) &payload, size_t size)
- void [onInsertShift](#) ([ConnectionBase](#) *connection, const [S2C_InsertShiftReply](#) &payload, size_t size)
- void [onInsertWorker](#) ([ConnectionBase](#) *connection, const [S2C_InsertWorkerReply](#) &payload, size_t size)
- void [onDeleteWorker](#) ([ConnectionBase](#) *connection, const [S2C_DeleteWorkerReply](#) &payload, size_t size)
- void [onSync](#) ([ConnectionBase](#) *connection, const [S2C_ClientSync](#) &payload, size_t size)

Additional Inherited Members

5.1.1 Detailed Description

[App](#) class represents the base of everything Inherits from [wxApp](#) and [ConnectionObserver](#) classes.

5.1.2 Member Function Documentation

5.1.2.1 [getClient](#)()

```
RestaurantClient& App::getClient ( ) [inline]
```

Returns

current client connection

5.1.2.2 `getMainWindow()`

```
MainWindow* App::getMainWindow ( ) const [inline]
```

Returns

main window pointer

5.1.2.3 `getShiftWindow()`

```
ShiftEditorWindow* App::getShiftWindow ( ) const [inline]
```

Returns

[Shift](#) Editor window pointer

5.1.2.4 `getWorkerWindow()`

```
WorkerEditorWindow* App::getWorkerWindow ( ) const [inline]
```

Returns

Worker Editor window pointer

5.1.2.5 `setMainWindow()`

```
void App::setMainWindow (
    MainWindow * wnd ) [inline]
```

Method that sets Main Window pointer.

Parameters

<i>wnd</i>	- Main Window pointer
------------	-----------------------

5.1.2.6 `setShiftWindow()`

```
void App::setShiftWindow (
    ShiftEditorWindow * wnd ) [inline]
```

Method that sets Main Window pointer.

Parameters

<i>wnd</i>	- Shift Editor window pointer
------------	---

5.1.2.7 setWorkerWindow()

```
void App::setWorkerWindow (
    WorkerEditorWindow * wnd ) [inline]
```

Method that sets Main Window pointer.

Parameters

<i>wnd</i>	- Worker Editor window pointer
------------	--------------------------------

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

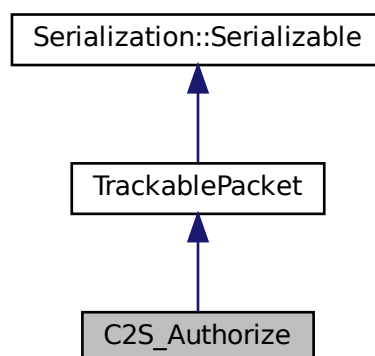
- UIPizzaClient/App.h
- UIPizzaClient/App.cpp

5.2 C2S_Authorize Class Reference

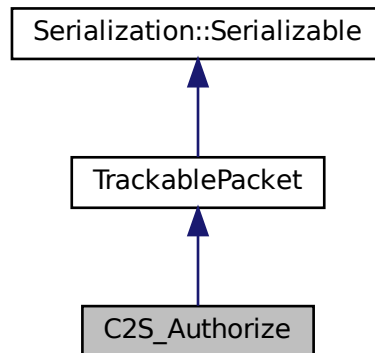
Client-to-server packet that authorizes the client with a token.

```
#include <C2S_Authorize.h>
```

Inheritance diagram for C2S_Authorize:



Collaboration diagram for C2S_Authorize:



Public Member Functions

- `Type` `getType ()` const override
- `C2S_Authorize` (`std::string` token)
Construct a new authorize packet with a token.
- `std::string` `getToken ()` const
Gets the object's auth token.
- void `setToken` (`std::string` token)
Sets the auth token.
- `std::ostream &` `serialize` (`std::ostream &`destination) const override
Serializes this object into a binary output stream.
- `std::istream &` `deserialize` (`std::istream &`source) override
Deserializes this object from a binary input stream.

Protected Attributes

- `std::string` `_token`

Friends

- bool `operator==` (const `C2S_Authorize` &lhs, const `C2S_Authorize` &rhs)
- bool `operator!=` (const `C2S_Authorize` &lhs, const `C2S_Authorize` &rhs)

5.2.1 Detailed Description

Client-to-server packet that authorizes the client with a token.

5.2.2 Constructor & Destructor Documentation

5.2.2.1 C2S_Authorize()

```
C2S_Authorize::C2S_Authorize (
    std::string token ) [inline]
```

Construct a new authorize packet with a token.

Parameters

<i>token</i>	auth token
--------------	------------

5.2.3 Member Function Documentation

5.2.3.1 deserialize()

```
std::istream& C2S_Authorize::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.2.3.2 getToken()

```
std::string C2S_Authorize::getToken ( ) const [inline]
```

Gets the object's auth token.

Returns

std::string

5.2.3.3 getType()

```
Type C2S_Authorize::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.2.3.4 serialize()

```
std::ostream& C2S_Authorize::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.2.3.5 setToken()

```
void C2S_Authorize::setToken (
    std::string token ) [inline]
```

Sets the auth token.

Parameters

<i>token</i>	
--------------	--

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

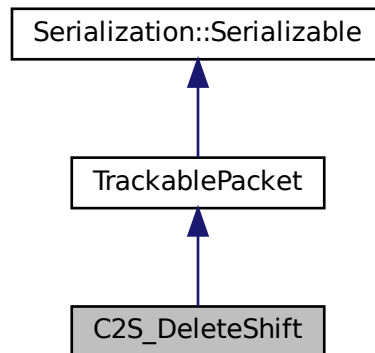
- BaseLibrary/C2S_Authorize.h

5.3 C2S_DeleteShift Class Reference

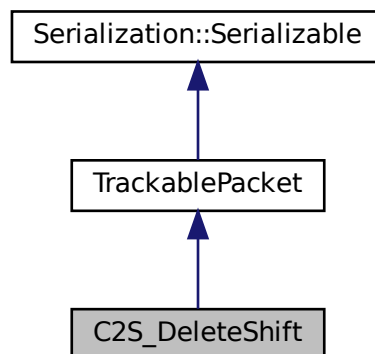
Client-to-server packet that deletes a [Shift](#) by its id.

```
#include <C2S_DeleteShift.h>
```

Inheritance diagram for C2S_DeleteShift:



Collaboration diagram for C2S_DeleteShift:



Public Member Functions

- [Type](#) `getType ()` const override
- virtual `identity_t` [getShiftId \(\)](#) const
Gets the target [Shift](#) id.

- virtual void [setShiftId](#) (const identity_t shiftId)
Sets the target [Shift](#) id.
- [C2S_DeleteShift](#) (identity_t shiftId)
Construct a new request to delete the given [Shift](#) object by its id.
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- identity_t **_shiftId**

Friends

- bool **operator==** (const [C2S_DeleteShift](#) &lhs, const [C2S_DeleteShift](#) &rhs)
- bool **operator!=** (const [C2S_DeleteShift](#) &lhs, const [C2S_DeleteShift](#) &rhs)

5.3.1 Detailed Description

Client-to-server packet that deletes a [Shift](#) by its id.

5.3.2 Constructor & Destructor Documentation

5.3.2.1 C2S_DeleteShift()

```
C2S_DeleteShift::C2S_DeleteShift (
    identity_t shiftId ) [inline]
```

Construct a new request to delete the given [Shift](#) object by its id.

Parameters

<i>shiftId</i>	target Shift id
----------------	---------------------------------

5.3.3 Member Function Documentation

5.3.3.1 deserialize()

```
std::istream& C2S_DeleteShift::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.3.3.2 getShiftId()

```
virtual identity_t C2S_DeleteShift::getShiftId ( ) const [inline], [virtual]
```

Gets the target [Shift](#) id.

Returns

identity_t

5.3.3.3 getType()

```
Type C2S_DeleteShift::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.3.3.4 serialize()

```
std::ostream& C2S_DeleteShift::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.3.3.5 setShiftId()

```
virtual void C2S_DeleteShift::setShiftId (
    const identity_t shiftId ) [inline], [virtual]
```

Sets the target [Shift](#) id.

Parameters

<i>shift↔ Id</i>	
----------------------	--

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

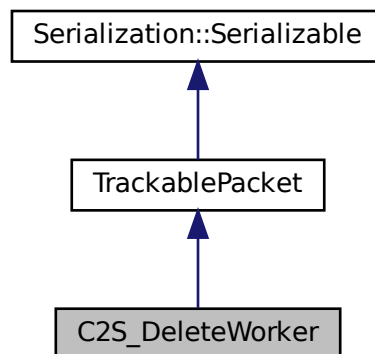
- BaseLibrary/C2S_DeleteShift.h

5.4 C2S_DeleteWorker Class Reference

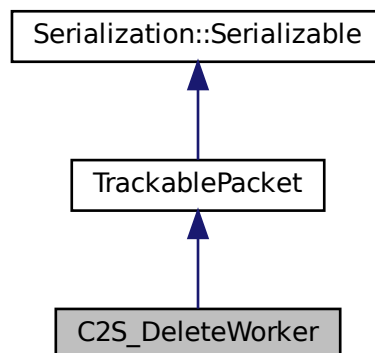
Client-to-server packet that deletes a [ShiftWorker](#) by its id.

```
#include <C2S_DeleteWorker.h>
```

Inheritance diagram for C2S_DeleteWorker:



Collaboration diagram for C2S_DeleteWorker:



Public Member Functions

- [Type](#) [getType](#) () const override
- virtual [identity_t](#) [getWorkerId](#) () const
Gets the target [ShiftWorker](#) id.
- virtual void [setWorkerId](#) (const [identity_t](#) shiftId)
Sets the target [ShiftWorker](#) id.
- [C2S_DeleteWorker](#) ([identity_t](#) workerId)
Construct a new request to delete a [ShiftWorker](#) by its id.
- [std::ostream](#) & [serialize](#) ([std::ostream](#) &destination) const override
Serializes this object into a binary output stream.
- [std::istream](#) & [deserialize](#) ([std::istream](#) &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- `identity_t _workerId`

Friends

- `bool operator==` (const [C2S_DeleteWorker](#) &lhs, const [C2S_DeleteWorker](#) &rhs)
- `bool operator!=` (const [C2S_DeleteWorker](#) &lhs, const [C2S_DeleteWorker](#) &rhs)

5.4.1 Detailed Description

Client-to-server packet that deletes a [ShiftWorker](#) by its id.

5.4.2 Constructor & Destructor Documentation

5.4.2.1 C2S_DeleteWorker()

```
C2S_DeleteWorker::C2S_DeleteWorker (
    identity_t workerId ) [inline]
```

Construct a new request to delete a ShiftWorker by its id.

Parameters

<i>workerId</i>	target ShiftWorker id
-----------------	---------------------------------------

5.4.3 Member Function Documentation

5.4.3.1 deserialize()

```
std::istream& C2S_DeleteWorker::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.4.3.2 getType()

```
Type C2S_DeleteWorker::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.4.3.3 getWorkerId()

```
virtual identity_t C2S_DeleteWorker::getWorkerId ( ) const [inline], [virtual]
```

Gets the target [ShiftWorker](#) id.

Returns

identity_t

5.4.3.4 serialize()

```
std::ostream& C2S_DeleteWorker::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.4.3.5 setWorkerId()

```
virtual void C2S_DeleteWorker::setWorkerId (
    const identity_t shiftId ) [inline], [virtual]
```

Sets the target [ShiftWorker](#) id.

Parameters

<i>shiftId</i>	
----------------	--

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

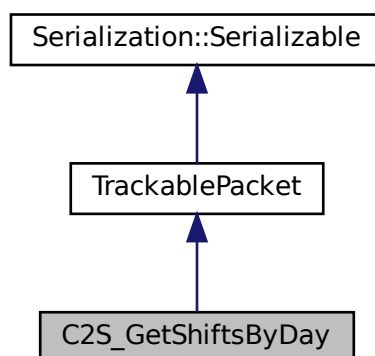
- BaseLibrary/C2S_DeleteWorker.h

5.5 C2S_GetShiftsByDay Class Reference

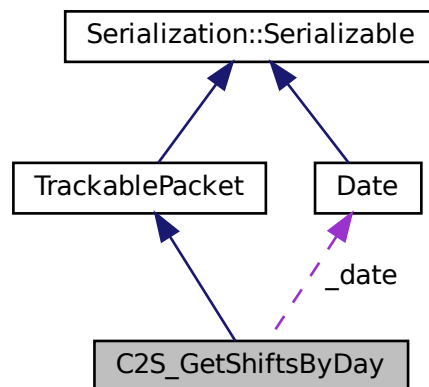
Client-to-server request to list all [Shift](#) objects in a given date.

```
#include <C2S_GetShiftsByDay.h>
```

Inheritance diagram for C2S_GetShiftsByDay:



Collaboration diagram for C2S_GetShiftsByDay:



Public Member Functions

- [Type](#) [getType](#) () const override
- [C2S_GetShiftsByDay](#) ([Date](#) date)
Construct a new request to list shifts by date.
- [Date](#) & [getDate](#) ()
Gets the target date by referece.
- const [Date](#) & [getDate](#) () const
Gets the target date.
- void [setDate](#) ([Date](#) date)
Sets the target date.
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- [Date](#) [_date](#)

Friends

- bool **operator==** (const [C2S_GetShiftsByDay](#) &lhs, const [C2S_GetShiftsByDay](#) &rhs)
- bool **operator!=** (const [C2S_GetShiftsByDay](#) &lhs, const [C2S_GetShiftsByDay](#) &rhs)

5.5.1 Detailed Description

Client-to-server request to list all [Shift](#) objects in a given date.

5.5.2 Constructor & Destructor Documentation

5.5.2.1 C2S_GetShiftsByDay()

```
C2S_GetShiftsByDay::C2S_GetShiftsByDay (
    Date date ) [inline]
```

Construct a new request to list shifts by date.

Parameters

<i>date</i>	target date
-------------	-------------

5.5.3 Member Function Documentation

5.5.3.1 deserialize()

```
std::istream& C2S_GetShiftsByDay::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.5.3.2 getDate() [1/2]

```
Date& C2S_GetShiftsByDay::getDate ( ) [inline]
```

Gets the target date by referece.

Returns

[Date](#)& target date

5.5.3.3 getDate() [2/2]

```
const Date& C2S_GetShiftsByDay::getDate ( ) const [inline]
```

Gets the target date.

Returns

const Date&

5.5.3.4 getType()

```
Type C2S_GetShiftsByDay::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.5.3.5 serialize()

```
std::ostream& C2S_GetShiftsByDay::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.5.3.6 setDate()

```
void C2S_GetShiftsByDay::setDate (
    Date date ) [inline]
```

Sets the target date.

Parameters

<i>date</i>	
-------------	--

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

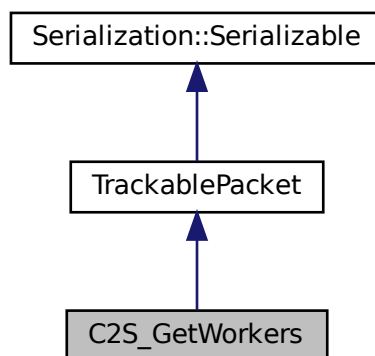
- BaseLibrary/C2S_GetShiftsByDay.h

5.6 C2S_GetWorkers Class Reference

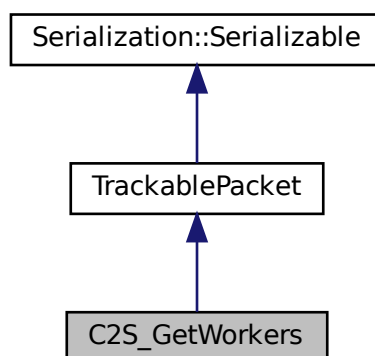
Client-to-server request to list all [ShiftWorker](#) objects.

```
#include <C2S_GetWorkers.h>
```

Inheritance diagram for C2S_GetWorkers:



Collaboration diagram for C2S_GetWorkers:



Public Member Functions

- [Type](#) `getType` () const override

Friends

- bool **operator==** (const [C2S_GetWorkers](#) &lhs, const [C2S_GetWorkers](#) &rhs)
- bool **operator!=** (const [C2S_GetWorkers](#) &lhs, const [C2S_GetWorkers](#) &rhs)

Additional Inherited Members

5.6.1 Detailed Description

Client-to-server request to list all [ShiftWorker](#) objects.

5.6.2 Member Function Documentation

5.6.2.1 `getType()`

```
Type C2S_GetWorkers::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

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

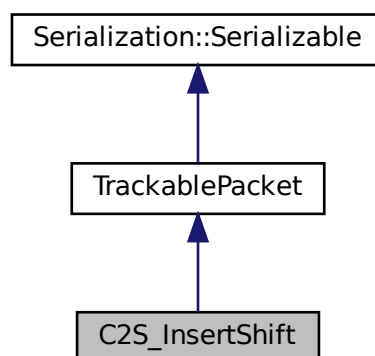
- BaseLibrary/C2S_GetWorkers.h

5.7 C2S_InsertShift Class Reference

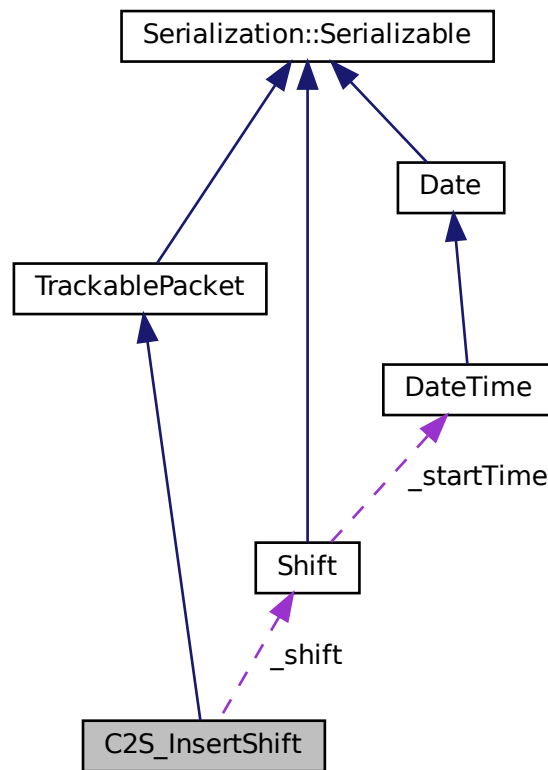
Client-to-server request to insert the given [Shift](#) object.

```
#include <C2S_InsertShift.h>
```

Inheritance diagram for C2S_InsertShift:



Collaboration diagram for C2S_InsertShift:



Public Member Functions

- [Type](#) `getType ()` const override
- virtual [Shift](#) `getShift ()` const
Gets the [Shift](#) object.
- virtual [Shift](#) & `getShift ()`
Gets the [Shift](#) object by reference.
- virtual void `setShift (const Shift &shift)`
Sets the [Shift](#) object.
- virtual bool `isModifyExisting ()` const
Returns whether the request should modify the existing records.
- virtual void `setModifyExisting (const bool modifyExisting)`
Sets whether the request should modify the existing records.
- [C2S_InsertShift](#) ([Shift](#) shift, bool modifyExisting=false)
Construct a new insert request.
- std::ostream & `serialize (std::ostream &destination)` const override
Serializes this object into a binary output stream.
- std::istream & `deserialize (std::istream &source)` override
Deserializes this object from a binary input stream.

Protected Attributes

- [Shift_shift](#)
- `bool_modifyExisting`

Friends

- `bool operator==` (const [C2S_InsertShift](#) &lhs, const [C2S_InsertShift](#) &rhs)
- `bool operator!=` (const [C2S_InsertShift](#) &lhs, const [C2S_InsertShift](#) &rhs)

5.7.1 Detailed Description

Client-to-server request to insert the given [Shift](#) object.

5.7.2 Constructor & Destructor Documentation

5.7.2.1 C2S_InsertShift()

```
C2S_InsertShift::C2S_InsertShift (
    Shift shift,
    bool modifyExisting = false ) [inline]
```

Construct a new insert request.

Parameters

<i>shift</i>	object to be inserted
<i>modifyExisting</i>	whether to update existing records

5.7.3 Member Function Documentation

5.7.3.1 deserialize()

```
std::istream& C2S_InsertShift::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.7.3.2 getShift() [1/2]

```
virtual Shift C2S_InsertShift::getShift ( ) const [inline], [virtual]
```

Gets the [Shift](#) object.

Returns

[Shift](#)

5.7.3.3 getShift() [2/2]

```
virtual Shift& C2S_InsertShift::getShift ( ) [inline], [virtual]
```

Gets the [Shift](#) object by reference.

Returns

[Shift](#)&

5.7.3.4 getType()

```
Type C2S_InsertShift::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.7.3.5 isModifyExisting()

```
virtual bool C2S_InsertShift::isModifyExisting ( ) const [inline], [virtual]
```

Returns whether the request should modify the existing records.

Returns

true edit the [Shift](#)
false insert the [Shift](#)

5.7.3.6 serialize()

```
std::ostream& C2S_InsertShift::serialize (   
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.7.3.7 setModifyExisting()

```
virtual void C2S_InsertShift::setModifyExisting (
    const bool modifyExisting ) [inline], [virtual]
```

Sets whether the request should modify the existing records.

Parameters

<i>modifyExisting</i>	
-----------------------	--

5.7.3.8 setShift()

```
virtual void C2S_InsertShift::setShift (
    const Shift & shift ) [inline], [virtual]
```

Sets the [Shift](#) object.

Parameters

<i>shift</i>	new Shift object
--------------	----------------------------------

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

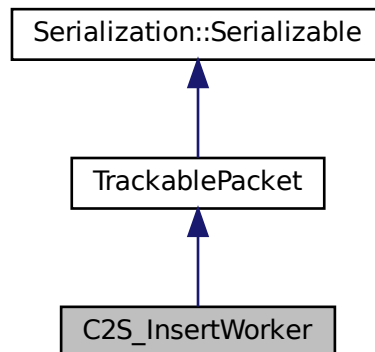
- BaseLibrary/C2S_InsertShift.h

5.8 C2S_InsertWorker Class Reference

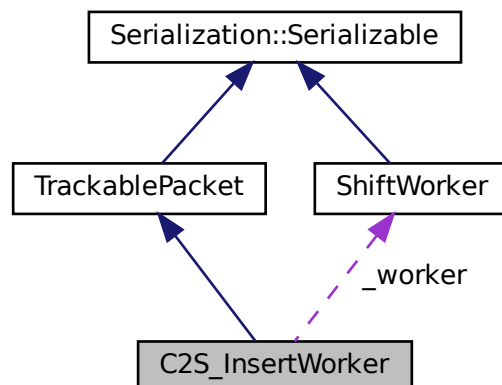
Client-to-server request to insert a [ShiftWorker](#) object.

```
#include <C2S_InsertWorker.h>
```

Inheritance diagram for C2S_InsertWorker:



Collaboration diagram for C2S_InsertWorker:



Public Member Functions

- [Type](#) `getType` () const override

Protected Attributes

- [ShiftWorker](#) `_worker`
- `bool _modifyExisting`
- virtual [ShiftWorker](#) `getWorker` () const

- Gets the code [ShiftWorker](#).*
- virtual [ShiftWorker](#) & [getWorker](#) ()
Gets the code [ShiftWorker](#) by reference.
- virtual void [setWorker](#) (const [ShiftWorker](#) &shift)
Sets the code [ShiftWorker](#).
- virtual bool [isModifyExisting](#) () const
Returns whether this request should modify existing records.
- virtual void [setModifyExisting](#) (const bool modifyExisting)
Sets whether this request should modify existing records.
- [C2S_InsertWorker](#) ([ShiftWorker](#) worker, bool modifyExisting=false)
Construct a new [ShiftWorker](#) insert request.
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.
- bool **operator==** (const [C2S_InsertWorker](#) &lhs, const [C2S_InsertWorker](#) &rhs)
- bool **operator!=** (const [C2S_InsertWorker](#) &lhs, const [C2S_InsertWorker](#) &rhs)

5.8.1 Detailed Description

Client-to-server request to insert a [ShiftWorker](#) object.

5.8.2 Constructor & Destructor Documentation

5.8.2.1 C2S_InsertWorker()

```
C2S_InsertWorker::C2S_InsertWorker (
    ShiftWorker worker,
    bool modifyExisting = false ) [inline]
```

Construct a new [ShiftWorker](#) insert request.

Parameters

<i>shift</i>	the object to be inserted
<i>modifyExisting</i>	whether to modify existing records

5.8.3 Member Function Documentation

5.8.3.1 deserialize()

```
std::istream& C2S_InsertWorker::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.8.3.2 `getType()`

```
Type C2S_InsertWorker::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.8.3.3 `getWorker()` [1/2]

```
virtual ShiftWorker C2S_InsertWorker::getWorker ( ) const [inline], [virtual]
```

Gets thecode [ShiftWorker](#){}.

Returns

[ShiftWorker](#)

5.8.3.4 `getWorker()` [2/2]

```
virtual ShiftWorker& C2S_InsertWorker::getWorker ( ) [inline], [virtual]
```

Gets thecode [ShiftWorker](#)} by reference.

Returns

[ShiftWorker](#)&

5.8.3.5 isModifyExisting()

```
virtual bool C2S_InsertWorker::isModifyExisting ( ) const [inline], [virtual]
```

Returns whether this request should modify existing records.

Returns

true edit operation
false insert operation

5.8.3.6 serialize()

```
std::ostream& C2S_InsertWorker::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.8.3.7 setModifyExisting()

```
virtual void C2S_InsertWorker::setModifyExisting (
    const bool modifyExisting ) [inline], [virtual]
```

Sets whether this request should modify existing records.

Parameters

<i>modifyExisting</i>	whether to update existing records
-----------------------	------------------------------------

5.8.3.8 setWorker()

```
virtual void C2S_InsertWorker::setWorker (
    const ShiftWorker & shift ) [inline], [virtual]
```

Sets the code `ShiftWorker`).

Parameters

<i>shift</i>	
--------------	--

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

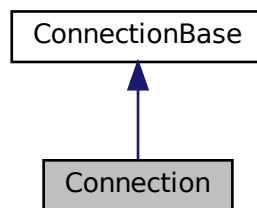
- BaseLibrary/C2S_InsertWorker.h

5.9 Connection Class Reference

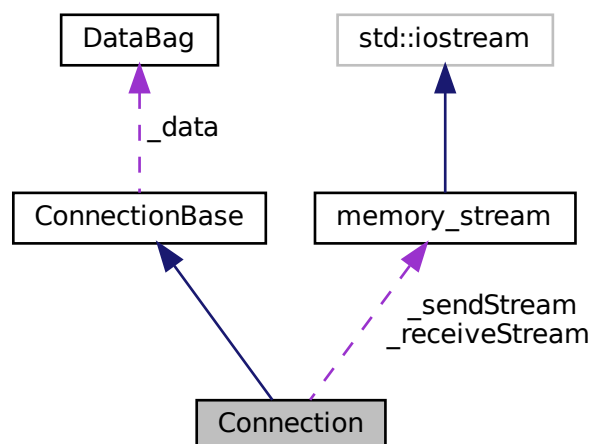
Main class that wraps a native socket into a managed TCP Client.

```
#include <Connection.h>
```

Inheritance diagram for Connection:



Collaboration diagram for Connection:



Public Member Functions

- bool `isAlive` () const override
- bool `setReadingAsync` (bool `readAsync`) override
Controls whether all input should be processed in a background thread.
- `Connection` (bool `subscribeToPing`=true)
Construct a new `Connection` object.
- int `connect` (const std::string &host, int port) override
Connects to the target endpoint.
- int `connect` (PSocket *socket) override
Wraps the existing socket object.
- void `writeSync` (const `Serializable` &payload) override
Synchronously writes the given payload into the network stream.
- std::shared_ptr< `Serializable` > `readSync` () override
Synchronously reads a single packet from the network stream.

Protected Member Functions

- void `cleanup` () override
Clean-up method called after the client is disconnected.
- void `closeError` (const std::exception &exception) override
Closes the connection as a result of the given exception.
- bool `receiveFromSocket` (size_t length)
Helper method to read length bytes from input into the buffer.
- bool `sendBuffer` ()
Helper method to push bytes stored in the send buffer.
- bool `readAsync` () override
Background task to receive payload and handle all connection events.
- virtual std::shared_ptr< `Serializable` > `readSyncInternal` ()
Internal function to read a single packet from the stream.

Protected Attributes

- `memory_stream _sendStream`
- `memory_stream _receiveStream`
- std::recursive_mutex `_readLock`
- std::thread `_readThread`

Additional Inherited Members

5.9.1 Detailed Description

Main class that wraps a native socket into a managed TCP Client.

5.9.2 Constructor & Destructor Documentation

5.9.2.1 `Connection()`

```
Connection::Connection (
    bool subscribeToPing = true ) [inline]
```

Construct a new `Connection` object.

Parameters

<i>subscribeToPing</i>	whether to subscribe to the PingService
------------------------	---

5.9.3 Member Function Documentation

5.9.3.1 cleanup()

```
void Connection::cleanup ( ) [override], [protected], [virtual]
```

Clean-up method called after the client is disconnected.

Reimplemented from [ConnectionBase](#).

5.9.3.2 closeError()

```
void Connection::closeError (
    const std::exception & exception ) [override], [protected], [virtual]
```

Closes the connection as a result of the given exception.

Parameters

<i>exception</i>	caught exception
------------------	------------------

Implements [ConnectionBase](#).

5.9.3.3 connect() [1/2]

```
int Connection::connect (
    const std::string & host,
    int port ) [override], [virtual]
```

Connects to the target endpoint.

Parameters

<i>host</i>	hostname, DNS is not supported
<i>port</i>	port number

Returns

int connection id

Implements [ConnectionBase](#).

5.9.3.4 connect() [2/2]

```
int Connection::connect (
    PSocket * socket ) [override], [virtual]
```

Wraps the existing socket object.

Parameters

<i>socket</i>	client socket
---------------	---------------

Returns

int connection id

Implements [ConnectionBase](#).

5.9.3.5 isAlive()

```
bool Connection::isAlive ( ) const [inline], [override], [virtual]
```

Returns

Whether the connection is alive and connected

Implements [ConnectionBase](#).

5.9.3.6 readAsync()

```
bool Connection::readAsync ( ) [override], [protected], [virtual]
```

Background task to receive payload and handle all connection events.

Returns

true exited properly
false exited with an exception

Implements [ConnectionBase](#).

5.9.3.7 readSync()

```
std::shared_ptr< Serializable > Connection::readSync ( ) [override], [virtual]
```

Synchronously reads a single packet from the network stream.

Returns

std::shared_ptr<Serializable> payload received

Implements [ConnectionBase](#).

5.9.3.8 readSyncInternal()

```
std::shared_ptr< Serializable > Connection::readSyncInternal ( ) [protected], [virtual]
```

Internal function to read a single packet from the stream.

Returns

std::shared_ptr<Serializable> payload received

5.9.3.9 receiveFromSocket()

```
bool Connection::receiveFromSocket (
    size_t length ) [protected]
```

Helper method to read length bytes from input into the buffer.

Parameters

<i>length</i>	number of bytes to be read
---------------	----------------------------

Returns

true success
false failure

5.9.3.10 sendBuffer()

```
bool Connection::sendBuffer ( ) [protected]
```

Helper method to push bytes stored in the send buffer.

Returns

true success
false failure

5.9.3.11 setReadingAsync()

```
bool Connection::setReadingAsync (
    bool readAsync ) [override], [virtual]
```

Controls whether all input should be processed in a background thread.

Parameters

<i>readAsync</i>	
------------------	--

Returns

whether the background receive task is now running

Implements [ConnectionBase](#).

5.9.3.12 writeSync()

```
void Connection::writeSync (
    const Serializable & payload ) [override], [virtual]
```

Synchronously writes the given payload into the network stream.

Parameters

<i>payload</i>	payload object
----------------	----------------

Implements [ConnectionBase](#).

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

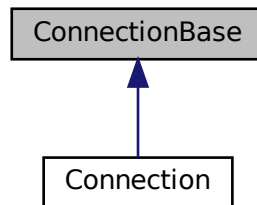
- BaseLibrary/Connection.h
- BaseLibrary/Connection.cpp

5.10 ConnectionBase Class Reference

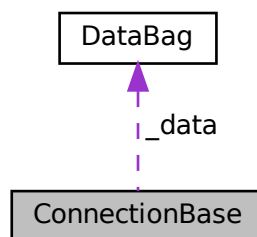
Abstract class implementing basic functionality for wrapping a PSocket into a managed TCP Client.

```
#include <ConnectionBase.h>
```

Inheritance diagram for ConnectionBase:



Collaboration diagram for ConnectionBase:



Public Member Functions

- virtual bool [isAlive](#) () const =0
- virtual [DataBag](#) & [getData](#) ()
 - Gets the [DataBag](#) associated with this connection by reference.*
- virtual const [DataBag](#) & [getData](#) () const
 - Gets the [DataBag](#) associated with this connection by const reference.*
- virtual int [getId](#) () const
 - Gets the unique id of this connection.*
- virtual bool [isReadingAsync](#) () const
- virtual bool [setReadingAsync](#) (bool [readAsync](#))=0
 - Controls whether all input should be processed in a background thread.*
- [ConnectionBase](#) (bool subscribeToPing)
 - Construct a new [ConnectionBase](#) object.*
- virtual int [connect](#) (const std::string &host, int port)=0
 - Connects to the target endpoint.*
- virtual int [connect](#) (PSocket *socket)=0

- Wraps the existing socket object.*
- virtual void `close` ()
Closes this connection and releases all resources.
- virtual void `subscribe` (`ConnectionObserver` *observer)
Subscribes an event observer to this connection.
- virtual void `unsubscribe` (`ConnectionObserver` *observer)
Unsubscribes an event observer to this connection.
- virtual void `writeSync` (const `Serializable` &payload)=0
Synchronously writes the given payload into the network stream.
- virtual int `writeRequestSync` (`TrackablePacket` &payload)
Synchronously writes the given request into the network stream.
- virtual void `writeReplySync` (`TrackablePacket` &payload, int requestId=0)
Synchronously writes the given reply into the network stream.
- virtual void `writeReplySync` (`TrackablePacket` &payload, const `TrackablePacket` &request)
Synchronously writes the given reply into the network stream.
- virtual std::shared_ptr< `Serializable` > `readSync` ()=0
Synchronously reads a single packet from the network stream.

Protected Member Functions

- virtual int `newRequestId` ()
Generates a new request id.
- virtual void `onConnected` ()
Event handler for connection established.
- virtual void `onDisconnected` (const std::exception &exception)
Event handler for connection closed.
- virtual void `onPayloadSent` (const `Serializable` &payload, size_t size)
Event handler for payload sent.
- virtual void `onPayloadReceived` (const `Serializable` &payload, size_t size)
Event handler for payload received.
- virtual void `cleanup` ()
Clean-up method called after the client is disconnected.
- virtual void `closeError` (const std::exception &exception)=0
Closes the connection as a result of the given exception.
- virtual bool `readAsync` ()=0
Background task to receive payload and handle all connection events.
- virtual void `assertConnected` () const
Throws an error if the connection is not alive.

Protected Attributes

- `PSocket` * `_socket`
- `DataBag` `_data`
- std::list< `ConnectionObserver` * > `_observers`
- std::atomic< bool > `_readingAsync`
- int `_id`
- int `_lastRequestId`

Static Protected Attributes

- static int **LastId** = 0

5.10.1 Detailed Description

Abstract class implementing basic functionality for wrapping a PSocket into a managed TCP Client.

5.10.2 Constructor & Destructor Documentation

5.10.2.1 ConnectionBase()

```
ConnectionBase::ConnectionBase (
    bool subscribeToPing )
```

Construct a new [ConnectionBase](#) object.

Parameters

<i>subscribeToPing</i>	whether to subscribe to the PingService
------------------------	---

5.10.3 Member Function Documentation

5.10.3.1 assertConnected()

```
void ConnectionBase::assertConnected ( ) const [protected], [virtual]
```

Throws an error if the connection is not alive.

5.10.3.2 cleanup()

```
void ConnectionBase::cleanup ( ) [protected], [virtual]
```

Clean-up method called after the client is disconnected.

Reimplemented in [Connection](#).

5.10.3.3 close()

```
void ConnectionBase::close ( ) [virtual]
```

Closes this connection and releases all resources.

5.10.3.4 closeError()

```
virtual void ConnectionBase::closeError (
    const std::exception & exception ) [protected], [pure virtual]
```

Closes the connection as a result of the given exception.

Parameters

<i>exception</i>	caught exception
------------------	------------------

Implemented in [Connection](#).

5.10.3.5 connect() [1/2]

```
virtual int ConnectionBase::connect (
    const std::string & host,
    int port ) [pure virtual]
```

Connects to the target endpoint.

Parameters

<i>host</i>	hostname, DNS is not supported
<i>port</i>	port number

Returns

int connection id

Implemented in [Connection](#).

5.10.3.6 connect() [2/2]

```
virtual int ConnectionBase::connect (
    PSocket * socket ) [pure virtual]
```

Wraps the existing socket object.

Parameters

<i>socket</i>	client socket
---------------	---------------

Returns

int connection id

Implemented in [Connection](#).

5.10.3.7 `getData()` [1/2]

```
virtual DataBag& ConnectionBase::getData ( ) [inline], [virtual]
```

Gets the [DataBag](#) associated with this connection by reference.

Returns

[DataBag](#)&

5.10.3.8 `getData()` [2/2]

```
virtual const DataBag& ConnectionBase::getData ( ) const [inline], [virtual]
```

Gets the [DataBag](#) associated with this connection by const reference.

Returns

const [DataBag](#)&

5.10.3.9 `getId()`

```
virtual int ConnectionBase::getId ( ) const [inline], [virtual]
```

Gets the unique id of this connection.

Returns

int connection id

5.10.3.10 isAlive()

```
virtual bool ConnectionBase::isAlive ( ) const [pure virtual]
```

Returns

Whether the connection is alive and connected

Implemented in [Connection](#).

5.10.3.11 isReadingAsync()

```
virtual bool ConnectionBase::isReadingAsync ( ) const [inline], [virtual]
```

Returns

Whether all input is being processed in a background thread

5.10.3.12 newRequestId()

```
int ConnectionBase::newRequestId ( ) [protected], [virtual]
```

Generates a new request id.

Returns

int request id

5.10.3.13 onConnected()

```
void ConnectionBase::onConnected ( ) [protected], [virtual]
```

Event handler for connection established.

5.10.3.14 onDisconnected()

```
void ConnectionBase::onDisconnected (
    const std::exception & exception ) [protected], [virtual]
```

Event handler for connection closed.

Parameters

<i>exception</i>	connection exception
------------------	----------------------

5.10.3.15 onPayloadReceived()

```
void ConnectionBase::onPayloadReceived (
    const Serializable & payload,
    size_t size ) [protected], [virtual]
```

Event handler for payload received.

Parameters

<i>payload</i>	received payload
<i>size</i>	size (in bytes) of received payload

5.10.3.16 onPayloadSent()

```
void ConnectionBase::onPayloadSent (
    const Serializable & payload,
    size_t size ) [protected], [virtual]
```

Event handler for payload sent.

Parameters

<i>payload</i>	sent payload
<i>size</i>	size (in bytes) of sent payload

5.10.3.17 readAsync()

```
virtual bool ConnectionBase::readAsync ( ) [protected], [pure virtual]
```

Background task to receive payload and handle all connection events.

Returns

true exited properly
false exited with an exception

Implemented in [Connection](#).

5.10.3.18 readSync()

```
virtual std::shared_ptr<Serializable> ConnectionBase::readSync ( ) [pure virtual]
```

Synchronously reads a single packet from the network stream.

Returns

std::shared_ptr<Serializable> payload received

Implemented in [Connection](#).

5.10.3.19 setReadingAsync()

```
virtual bool ConnectionBase::setReadingAsync (
    bool readAsync ) [pure virtual]
```

Controls whether all input should be processed in a background thread.

Parameters

<i>readAsync</i>	
------------------	--

Returns

whether the background receive task is now running

Implemented in [Connection](#).

5.10.3.20 subscribe()

```
void ConnectionBase::subscribe (
    ConnectionObserver * observer ) [virtual]
```

Subscribes an event observer to this connection.

Parameters

<i>observer</i>	event observer
-----------------	----------------

5.10.3.21 unsubscribe()

```
void ConnectionBase::unsubscribe (
```

```
ConnectionObserver * observer ) [virtual]
```

Unsubscribes an event observer to this connection.

Parameters

<i>observer</i>	event observer
-----------------	----------------

5.10.3.22 writeReplySync() [1/2]

```
void ConnectionBase::writeReplySync (
    TrackablePacket & payload,
    int requestId = 0 ) [virtual]
```

Synchronously writes the given reply into the network stream.

Parameters

<i>payload</i>	response packet
<i>requestId</i>	id of the request

5.10.3.23 writeReplySync() [2/2]

```
void ConnectionBase::writeReplySync (
    TrackablePacket & payload,
    const TrackablePacket & request ) [virtual]
```

Synchronously writes the given reply into the network stream.

Parameters

<i>payload</i>	response packet
<i>request</i>	request packet

5.10.3.24 writeRequestSync()

```
int ConnectionBase::writeRequestSync (
    TrackablePacket & payload ) [virtual]
```

Synchronously writes the given request into the network stream.

Parameters

<i>payload</i>	request packet
----------------	----------------

Returns

int unique id of the sent request

5.10.3.25 writeSync()

```
virtual void ConnectionBase::writeSync (
    const Serializable & payload ) [pure virtual]
```

Synchronously writes the given payload into the network stream.

Parameters

<i>payload</i>	payload object
----------------	----------------

Implemented in [Connection](#).

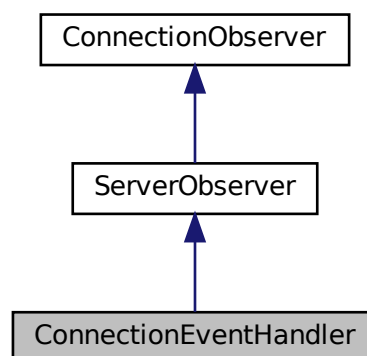
The documentation for this class was generated from the following files:

- BaseLibrary/ConnectionBase.h
- BaseLibrary/ConnectionBase.cpp

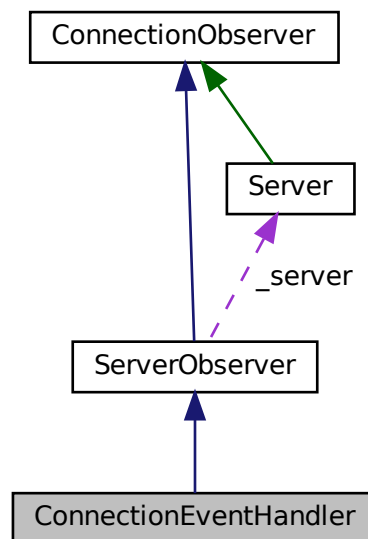
5.11 ConnectionEventHandler Class Reference

Logging handlers.

Inheritance diagram for ConnectionEventHandler:



Collaboration diagram for ConnectionEventHandler:



Public Member Functions

- void [onConnected](#) ([ConnectionBase](#) *connection) override
Event handler for connection established.
- void [onDisconnected](#) ([ConnectionBase](#) *connection, std::exception exception) override
Event handler for connection closed.
- void [onPayloadSent](#) ([ConnectionBase](#) *connection, const [Serializable](#) &payload, size_t size) override
Event handler for payload sent.

Protected Member Functions

- std::string **strPermissions** ([ConnectionBase](#) *connection)

Additional Inherited Members

5.11.1 Detailed Description

Logging handlers.

5.11.2 Member Function Documentation

5.11.2.1 onConnected()

```
void ConnectionEventHandler::onConnected (
    ConnectionBase * connection ) [inline], [override], [virtual]
```

Event handler for connection established.

Parameters

<i>connection</i>	event source
-------------------	--------------

Reimplemented from [ConnectionObserver](#).

5.11.2.2 onDisconnected()

```
void ConnectionEventHandler::onDisconnected (
    ConnectionBase * connection,
    std::exception exception ) [inline], [override], [virtual]
```

Event handler for connection closed.

Parameters

<i>connection</i>	event source
<i>exception</i>	connection exception

Reimplemented from [ConnectionObserver](#).

5.11.2.3 onPayloadSent()

```
void ConnectionEventHandler::onPayloadSent (
    ConnectionBase * connection,
    const Serializable & payload,
    size_t size ) [inline], [override], [virtual]
```

Event handler for payload sent.

Parameters

<i>connection</i>	event source
<i>payload</i>	sent payload
<i>size</i>	size (in bytes) of sent payload

Reimplemented from [ConnectionObserver](#).

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

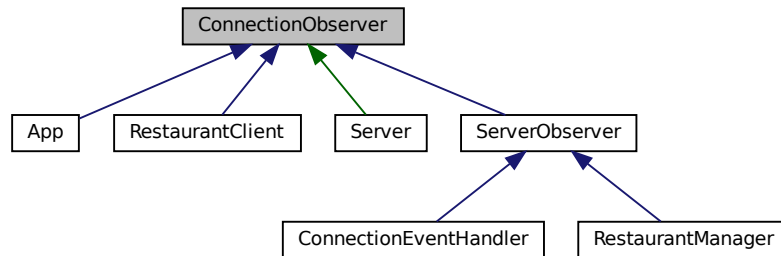
- Server/Source.cpp

5.12 ConnectionObserver Class Reference

Observer that allows listening to events of a [Connection](#) object.

```
#include <ConnectionBase.h>
```

Inheritance diagram for ConnectionObserver:



Public Member Functions

- template<typename T >
void **addHandler** (std::function< void([ConnectionBase](#) *, const T &, size_t)> callback)
Registers a payload handler.
- template<typename TBase , typename T >
void **addHandler** (void(TBase::*)(callback))([ConnectionBase](#) *, const T &, size_t)
Registers a payload handler (wrapper for inherited member methods)
- template<typename T >
void **removeHandlers** ()
Clears handlers for the specified payload type.
- virtual void **onConnected** ([ConnectionBase](#) *connection)
Event handler for connection established.
- virtual void **onDisconnected** ([ConnectionBase](#) *connection, std::exception exception)
Event handler for connection closed.
- virtual void **onPayloadSent** ([ConnectionBase](#) *connection, const [Serializable](#) &payload, size_t size)
Event handler for payload sent.
- virtual void **onPayloadReceived** ([ConnectionBase](#) *connection, const [Serializable](#) &payload, size_t size)
Event handler for payload received.

Protected Attributes

- std::map< [Type](#), std::list< std::function< void([ConnectionBase](#) *, const [Serializable](#) &, size_t)> > > **handlers**

5.12.1 Detailed Description

Observer that allows listening to events of a [Connection](#) object.

5.12.2 Member Function Documentation

5.12.2.1 addHandler() [1/2]

```
template<typename T >
void ConnectionObserver::addHandler (
    std::function< void(ConnectionBase *, const T &, size_t)> callback ) [inline]
```

Registers a payload handler.

Template Parameters

<i>T</i>	type of payload
----------	-----------------

Parameters

<i>callback</i>	handler
-----------------	---------

5.12.2.2 addHandler() [2/2]

```
template<typename TBase , typename T >
void ConnectionObserver::addHandler (
    void(TBase::*)(ConnectionBase *, const T &, size_t) callback ) [inline]
```

Registers a payload handler (wrapper for inherited member methods)

Template Parameters

<i>TBase</i>	type containing the method
<i>T</i>	type of payload

Parameters

<i>callback</i>	handler (member method)
-----------------	-------------------------

5.12.2.3 onConnected()

```
virtual void ConnectionObserver::onConnected (
    ConnectionBase * connection ) [inline], [virtual]
```

Event handler for connection established.

Parameters

<i>connection</i>	event source
-------------------	--------------

Reimplemented in [Server](#), and [ConnectionEventHandler](#).

5.12.2.4 onDisconnected()

```
virtual void ConnectionObserver::onDisconnected (
    ConnectionBase * connection,
    std::exception exception ) [inline], [virtual]
```

Event handler for connection closed.

Parameters

<i>connection</i>	event source
<i>exception</i>	connection exception

Reimplemented in [Server](#), and [ConnectionEventHandler](#).

5.12.2.5 onPayloadReceived()

```
virtual void ConnectionObserver::onPayloadReceived (
    ConnectionBase * connection,
    const Serializable & payload,
    size_t size ) [inline], [virtual]
```

Event handler for payload received.

Parameters

<i>connection</i>	event source
<i>payload</i>	received payload
<i>size</i>	size (in bytes) of received payload

Reimplemented in [Server](#), and [RestaurantManager](#).

5.12.2.6 onPayloadSent()

```
virtual void ConnectionObserver::onPayloadSent (
    ConnectionBase * connection,
    const Serializable & payload,
    size_t size ) [inline], [virtual]
```

Event handler for payload sent.

Parameters

<i>connection</i>	event source
<i>payload</i>	sent payload
<i>size</i>	size (in bytes) of sent payload

Reimplemented in [Server](#), and [ConnectionEventHandler](#).

5.12.2.7 removeHandlers()

```
template<typename T >
void ConnectionObserver::removeHandlers ( ) [inline]
```

Clears handlers for the specified payload type.

Template Parameters

<i>T</i>	type of payload
----------	-----------------

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

- BaseLibrary/ConnectionBase.h
- BaseLibrary/ConnectionBase.cpp

5.13 DataBag Class Reference

Class that stores data associatively in string-keyed dictionary.

```
#include <DataBag.h>
```

Public Member Functions

- void [clear](#) ()
Clears all stored data.
- void [remove](#) (std::string key)
Remove the object stored in the given key.
- template<typename T >
void [put](#) (std::string key, const T &value)
Stores a copy of the given object.
- template<typename T >
T & [get](#) (std::string key, T &fallback) const
Retrieves a stored object, using a fallback if not found.
- template<typename T >
const T & [get](#) (std::string key, const T &fallback) const
Retrieves a stored object, using a fallback if not found.

5.13.1 Detailed Description

Class that stores data associatively in string-keyed dictionary.

5.13.2 Member Function Documentation

5.13.2.1 clear()

```
void DataBag::clear ( )
```

Clears all stored data.

5.13.2.2 get() [1/2]

```
template<typename T >
T& DataBag::get (
    std::string key,
    T & fallback ) const [inline]
```

Retrieves a stored object, using a fallback if not found.

Template Parameters

<i>T</i>	type to be retrieved
----------	----------------------

Parameters

<i>key</i>	dictionary key
<i>fallback</i>	fallback value

Returns

T& reference to the retrieved object, or fallback if not found

5.13.2.3 get() [2/2]

```
template<typename T >
const T& DataBag::get (
    std::string key,
    const T & fallback ) const [inline]
```

Retrieves a stored object, using a fallback if not found.

Template Parameters

<i>T</i>	type to be retrieved
----------	----------------------

Parameters

<i>key</i>	dictionary key
<i>fallback</i>	fallback value

Returns

T& const reference to the retrieved object, or fallback if not found

5.13.2.4 put()

```
template<typename T >
void DataBag::put (
    std::string key,
    const T & value ) [inline]
```

Stores a copy of the given object.

Template Parameters

<i>T</i>	type to be stored
----------	-------------------

Parameters

<i>key</i>	dictionary key
<i>value</i>	object to be stored

5.13.2.5 remove()

```
void DataBag::remove (
    std::string key )
```

Remove the object stored in the given key.

Parameters

<i>key</i>	key to be deleted
------------	-------------------

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

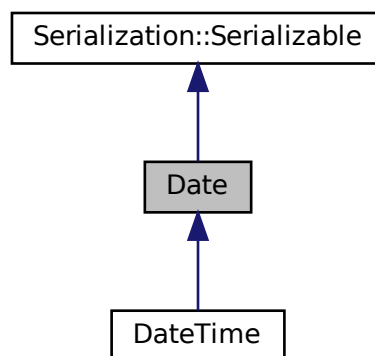
- BaseLibrary/DataBag.h
- BaseLibrary/DataBag.cpp

5.14 Date Class Reference

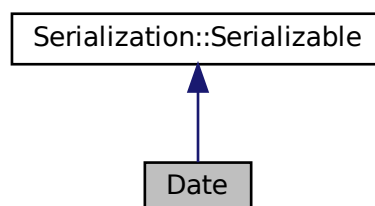
Serializable class representing a date (year, month, day)

```
#include <Date.h>
```

Inheritance diagram for Date:



Collaboration diagram for Date:



Public Member Functions

- `Type` `getType ()` const override
- `Date ()`

Construct a new `Date` object with default date (1st January 1970)

- [Date](#) (int year, int month, int day)
Construct a new [Date](#) object.
- uint32_t [getYear](#) () const
Gets the object's year.
- void [setYear](#) (int year)
Sets the year.
- uint8_t [getMonth](#) () const
Gets the object's month.
- void [setMonth](#) (int month)
Sets the month.
- uint8_t [getDay](#) () const
Gets the object's day.
- void [setDay](#) (int day)
Sets the day.
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.
- **Date** (const [Date](#) &other)
- **Date** ([Date](#) &&other) noexcept
- [Date](#) & **operator=** (const [Date](#) &other)
- [Date](#) & **operator=** ([Date](#) &&other) noexcept

Protected Attributes

- uint32_t **_year**
- uint8_t **_month**
- uint8_t **_day**

Friends

- bool **operator==** (const [Date](#) &lhs, const [Date](#) &rhs)
- bool **operator!=** (const [Date](#) &lhs, const [Date](#) &rhs)
- bool **operator<** (const [Date](#) &lhs, const [Date](#) &rhs)
- bool **operator<=** (const [Date](#) &lhs, const [Date](#) &rhs)
- bool **operator>** (const [Date](#) &lhs, const [Date](#) &rhs)
- bool **operator>=** (const [Date](#) &lhs, const [Date](#) &rhs)

5.14.1 Detailed Description

Serializable class representing a date (year, month, day)

5.14.2 Constructor & Destructor Documentation

5.14.2.1 `Date()` [1/2]

```
Date::Date ( ) [inline]
```

Construct a new [Date](#) object with default date (1st January 1970)

5.14.2.2 `Date()` [2/2]

```
Date::Date (
    int year,
    int month,
    int day ) [inline]
```

Construct a new [Date](#) object.

Parameters

<i>year</i>	year
<i>month</i>	month
<i>day</i>	day

5.14.3 Member Function Documentation

5.14.3.1 `deserialize()`

```
std::istream& Date::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

Reimplemented in [DateTime](#).

5.14.3.2 `getDay()`

```
uint8_t Date::getDay ( ) const [inline]
```

Gets the object's day.

Returns

`uint8_t`

5.14.3.3 `getMonth()`

```
uint8_t Date::getMonth ( ) const [inline]
```

Gets the object's month.

Returns

`uint8_t`

5.14.3.4 `getType()`

```
Type Date::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

Reimplemented in [DateTime](#).

5.14.3.5 `getYear()`

```
uint32_t Date::getYear ( ) const [inline]
```

Gets the object's year.

Returns

`uint32_t`

5.14.3.6 `serialize()`

```
std::ostream& Date::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

Reimplemented in [DateTime](#).

5.14.3.7 `setDay()`

```
void Date::setDay (  
    int day ) [inline]
```

Sets the day.

Parameters

<i>day</i>	
------------	--

5.14.3.8 `setMonth()`

```
void Date::setMonth (  
    int month ) [inline]
```

Sets the month.

Parameters

<i>month</i>	
--------------	--

5.14.3.9 `setYear()`

```
void Date::setYear (  
    int year ) [inline]
```

Sets the year.

Parameters

<i>year</i>	
-------------	--

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

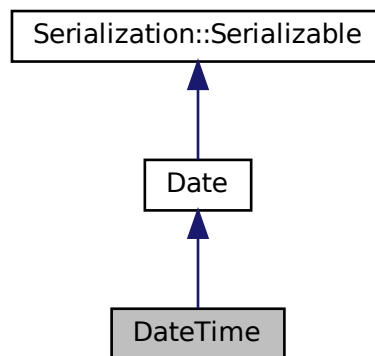
- BaseLibrary/Date.h

5.15 DateTime Class Reference

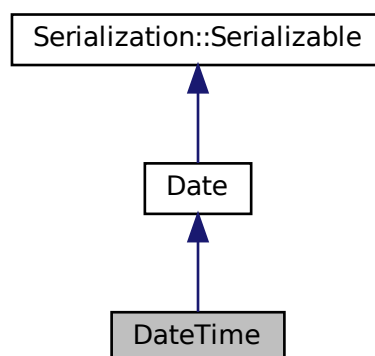
Serializable class extending date with additional (hour, minute, second) fields.

```
#include <DateTime.h>
```

Inheritance diagram for DateTime:



Collaboration diagram for DateTime:



Public Member Functions

- [Type](#) [getType](#) () const override
- [DateTime](#) ()
Construct a new [DateTime](#) object with default date (00:00 1st January 1970)
- [DateTime](#) (const [Date](#) &date)
Construct a new [DateTime](#) object from the given [Date](#).
- [DateTime](#) (const [Date](#) &date, int hour, int minute, int second)
Construct a new [DateTime](#) object from the given date and time.
- [DateTime](#) (int year, int month, int day, int hour, int minute, int second)
Construct a new [Date](#) Time object.
- uint8_t [getHour](#) () const
Gets the object's hour.
- void [setHour](#) (int hour)
Sets the hour.
- uint8_t [getMinute](#) () const
Gets the object's minute.
- void [setMinute](#) (int minute)
Sets the minute.
- uint8_t [getSecond](#) () const
Gets the object's second.
- void [setSecond](#) (int second)
Sets the second.
- [Date](#) [getDate](#) () const
Copies this [DateTime](#) as a [Date](#).
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.
- **DateTime** (const [DateTime](#) &other)
- **DateTime** ([DateTime](#) &&other) noexcept
- [DateTime](#) & **operator=** (const [DateTime](#) &other)
- [DateTime](#) & **operator=** ([DateTime](#) &&other) noexcept

Protected Attributes

- uint8_t **_hour** {}
- uint8_t **_minute** {}
- uint8_t **_second** {}

Friends

- bool **operator==** (const [DateTime](#) &lhs, const [DateTime](#) &rhs)
- bool **operator!=** (const [DateTime](#) &lhs, const [DateTime](#) &rhs)
- bool **operator<** (const [DateTime](#) &lhs, const [DateTime](#) &rhs)
- bool **operator<=** (const [DateTime](#) &lhs, const [DateTime](#) &rhs)
- bool **operator>** (const [DateTime](#) &lhs, const [DateTime](#) &rhs)
- bool **operator>=** (const [DateTime](#) &lhs, const [DateTime](#) &rhs)

5.15.1 Detailed Description

Serializable class extending date with additional (hour, minute, second) fields.

5.15.2 Constructor & Destructor Documentation

5.15.2.1 `DateTime()` [1/4]

```
DateTime::DateTime ( ) [inline]
```

Construct a new [DateTime](#) object with default date (00:00 1st January 1970)

5.15.2.2 `DateTime()` [2/4]

```
DateTime::DateTime (
    const Date & date ) [inline]
```

Construct a new [DateTime](#) object from the given [Date](#).

Parameters

<i>date</i>	date to be used
-------------	-----------------

5.15.2.3 `DateTime()` [3/4]

```
DateTime::DateTime (
    const Date & date,
    int hour,
    int minute,
    int second ) [inline]
```

Construct a new [DateTime](#) object from the given date and time.

Parameters

<i>date</i>	Date object
<i>hour</i>	hour
<i>minute</i>	minute
<i>second</i>	second

5.15.2.4 DateTime() [4/4]

```
DateTime::DateTime (
    int year,
    int month,
    int day,
    int hour,
    int minute,
    int second ) [inline]
```

Construct a new [Date](#) Time object.

Parameters

<i>year</i>	year
<i>month</i>	month
<i>day</i>	day
<i>hour</i>	hour
<i>minute</i>	minute
<i>second</i>	second

5.15.3 Member Function Documentation

5.15.3.1 deserialize()

```
std::istream& DateTime::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Date](#).

5.15.3.2 getDate()

```
Date DateTime::getDate ( ) const [inline]
```

Copies this [DateTime](#) as a [Date](#).

Returns

[Date](#)

5.15.3.3 getHour()

```
uint8_t DateTime::getHour ( ) const [inline]
```

Gets the object's hour.

Returns

uint8_t

5.15.3.4 getMinute()

```
uint8_t DateTime::getMinute ( ) const [inline]
```

Gets the object's minute.

Returns

uint8_t

5.15.3.5 getSecond()

```
uint8_t DateTime::getSecond ( ) const [inline]
```

Gets the object's second.

Returns

uint8_t

5.15.3.6 getType()

```
Type DateTime::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Reimplemented from [Date](#).

5.15.3.7 serialize()

```
std::ostream& DateTime::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Date](#).

5.15.3.8 setHour()

```
void DateTime::setHour (
    int hour ) [inline]
```

Sets the hour.

Parameters

<i>hour</i>	
-------------	--

5.15.3.9 setMinute()

```
void DateTime::setMinute (
    int minute ) [inline]
```

Sets the minute.

Parameters

<i>minute</i>	
---------------	--

5.15.3.10 setSecond()

```
void DateTime::setSecond (
    int second ) [inline]
```

Sets the second.

Parameters

<i>second</i>	
---------------	--

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

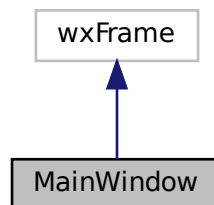
- BaseLibrary/DateTime.h

5.16 MainWindow Class Reference

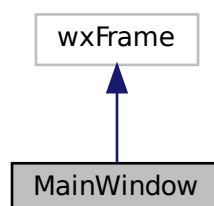
[MainWindow](#) class represents the main window of the app Inherits from wxFrame class.

```
#include <MainWindow.h>
```

Inheritance diagram for MainWindow:



Collaboration diagram for MainWindow:



Public Member Functions

- [MainWindow](#) ([App](#) *app)
MainWindow constructor.
- [~MainWindow](#) ()
MainWindow destructor.
- [Date](#) & [getCurrentDate](#) ()
- void [updateDate](#) ()

Methods that sets a new date.

- void [refresh](#) ()

Methods that refreshes the main window.

- void [checkJobs](#) ()

Method that checks for current jobs.

- void [resizeGrid](#) (int nRows, int nCols)

Methods that sets grid size.

- void [drawTable](#) ()

Method that overwrites the current grid with needed data.

- void **checkPermissions** ()
- void [onCurrDateButtonClicked](#) (wxCommandEvent &evt)

Event handlers.

- void **onRefreshButtonClicked** (wxCommandEvent &evt)
- void **onEditDataButtonClicked** (wxCommandEvent &evt)
- void **onDateChanged** (wxDateEvent &evt)
- void **onCellLeftClicked** (wxGridEvent &evt)
- void **onCellLeftDoubleClicked** (wxGridEvent &evt)
- void **onSizeChanged** (wxSizeEvent &evt)
- void **onShown** (wxShowEvent &evt)
- **wxDECLARE_EVENT_TABLE** ()

5.16.1 Detailed Description

[MainWindow](#) class represents the main window of the app Inherits from wxFrame class.

5.16.2 Constructor & Destructor Documentation

5.16.2.1 MainWindow()

```
MainWindow::MainWindow (
    App * app )
```

[MainWindow](#) constructor.

Parameters

<i>app</i>	- App object pointer
------------	--------------------------------------

5.16.3 Member Function Documentation

5.16.3.1 getCurrentDate()

```
Date& MainWindow::getCurrentDate ( ) [inline]
```

Returns

`_currentDate` - chosen date

5.16.3.2 `resizeGrid()`

```
void MainWindow::resizeGrid (
    int nRows,
    int nCols )
```

Methods that sets grid size.

Parameters

<i>nRows</i>	- number of rows
<i>nCols</i>	- number of columns

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

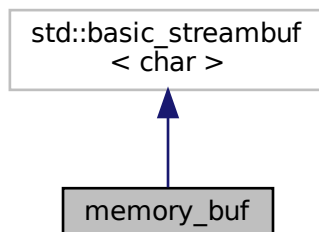
- UIPizzaClient/MainWindow.h
- UIPizzaClient/MainWindow.cpp

5.17 `memory_buf` Class Reference

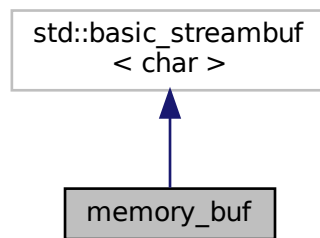
Provide in-memory alternatives to iostream streams/buffers.

```
#include <buffers.h>
```

Inheritance diagram for `memory_buf`:



Collaboration diagram for memory_buf:



Public Member Functions

- `memory_buf` (const pbyte_t buffer, size_t length)
Construct a memory buffer by taking ownership of the byte buffer.
- void `dispose` ()
Disposes the allocated buffer.
- size_t `getLength` () const
Gets the current length of the buffer.
- size_t `getCapacity` () const
Gets the length of the allocated internal buffer.
- bool `setLength` (size_t length, bool expand)
Sets length of the buffer, expanding if necessary.
- void `setData` (pbyte_t data, size_t length)
Takes ownership of the specified buffer.
- void `setData` (size_t length)
Allocates a new buffer of the specified length.
- std::streampos `getInPosition` () const
Gets the current 0-indexed position of input operations.
- std::streampos `getOutPosition` () const
Gets the current 0-indexed position of output operations.
- pbyte_t `data` () const
Return a pointer to the internal byte buffer.

Protected Member Functions

- void `updateBuffer` ()
Updates internal buffer pointers to the currently allocated buffer.
- char * `calcOffset` (char *base, char *now, char *end, std::streamoff off, std::ios_base::seekdir way)
Calculate buffer offset given a relative seek offset.
- std::streampos `seekpos` (std::streampos sp, std::ios_base::openmode which=std::ios_base::in|std::ios_base::out) override
- std::streampos `seekoff` (std::streamoff off, std::ios_base::seekdir way, std::ios_base::openmode which=std::ios_base::in|std::ios_base::out) override

5.17.1 Detailed Description

Provide in-memory alternatives to iostream streams/buffers.

In-memory byte buffer that handles both input and output operations

5.17.2 Constructor & Destructor Documentation

5.17.2.1 `memory_buf()`

```
memory_buf::memory_buf (
    const pbyte_t buffer,
    size_t length ) [inline]
```

Construct a memory buffer by taking ownership of the byte buffer.

Parameters

<i>buffer</i>	byte buffer
<i>length</i>	buffer length

5.17.3 Member Function Documentation

5.17.3.1 `setData()`

```
void memory_buf::setData (
    pbyte_t data,
    size_t length )
```

Takes ownership of the specified buffer.

Parameters

<i>data</i>	byte buffer
<i>length</i>	buffer length

5.17.3.2 `setLength()`

```
bool memory_buf::setLength (
```



```
size_t length,  
bool expand )
```

Sets length of the buffer, expanding if necessary.

Parameters

<i>length</i>	required length
<i>expand</i>	whether to allocate a larger buffer if necessary

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

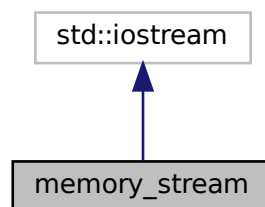
- BaseLibrary/buffers.h
- BaseLibrary/buffers.cpp

5.18 `memory_stream` Class Reference

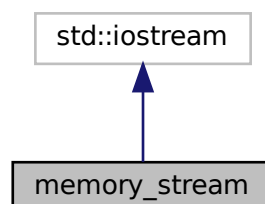
Provides an in-memory input/output stream with `std::iostream` interface, using [memory_buf](#) as the internal buffer.

```
#include <buffers.h>
```

Inheritance diagram for `memory_stream`:



Collaboration diagram for `memory_stream`:



Public Member Functions

- **memory_stream** (const pbyte_t buffer, size_t length)
- void **dispose** ()
Disposes the internal buffer.
- **memory_buf** & **buffer** ()

5.18.1 Detailed Description

Provides an in-memory input/output stream with std::iostream interface, using [memory_buf](#) as the internal buffer.

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

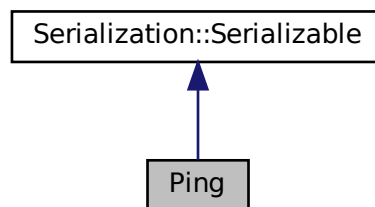
- BaseLibrary/buffers.h

5.19 Ping Class Reference

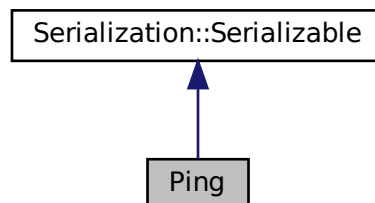
Basic [Ping](#) packet that forces the other endpoint to send a [PingReply](#).

```
#include <Ping.h>
```

Inheritance diagram for Ping:



Collaboration diagram for Ping:



Public Member Functions

- [Type](#) `getType` () const override

5.19.1 Detailed Description

Basic [Ping](#) packet that forces the other endpoint to send a [PingReply](#).

5.19.2 Member Function Documentation

5.19.2.1 `getType()`

```
Type Ping::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

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

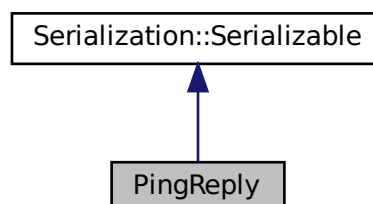
- BaseLibrary/Ping.h

5.20 PingReply Class Reference

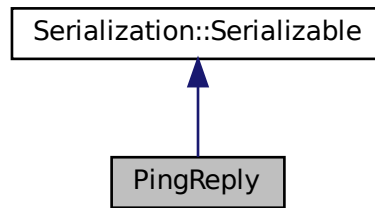
Response to a [Ping](#) request.

```
#include <PingReply.h>
```

Inheritance diagram for PingReply:



Collaboration diagram for PingReply:



Public Member Functions

- [Type](#) `getType ()` const override

5.20.1 Detailed Description

Response to a [Ping](#) request.

5.20.2 Member Function Documentation

5.20.2.1 `getType()`

```
Type PingReply::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

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

- BaseLibrary/PingReply.h

5.21 PingService Class Reference

Singleton class to broadcast [Ping](#) packets on all bound connections.

```
#include <PingService.h>
```

Collaboration diagram for PingService:



Public Member Functions

- [PingService](#) (int timeout=CLIENT_PING_INTERVAL)
Construct a new [Ping](#) Service object.
- void [start](#) ()
Starts the [PingService](#).
- void [stop](#) ()
Stops the [PingService](#).
- void [subscribe](#) ([ConnectionBase](#) *connection)
Binds the connection to this service.
- void [unsubscribe](#) ([ConnectionBase](#) *connection)
Unbinds the connection from this service.

Static Public Member Functions

- static [PingService](#) & [getInstance](#) ()
Gets the instance of this object.

Protected Member Functions

- void [worker](#) (int timeout)
Background task that broadcasts the [Ping](#) packets.

Protected Attributes

- std::atomic_bool [_run](#)
- std::set< [ConnectionBase](#) * > [_connections](#)
- std::recursive_mutex [_lock](#)
- int [_timeout](#)

Static Protected Attributes

- static [PingService](#) **_instance**

5.21.1 Detailed Description

Singleton class to broadcast [Ping](#) packets on all bound connections.

5.21.2 Constructor & Destructor Documentation

5.21.2.1 [PingService](#)()

```
PingService::PingService (
    int timeout = CLIENT_PING_INTERVAL )
```

Construct a new [Ping](#) Service object.

Parameters

<i>timeout</i>	Millisecond interval to send the Ping packets
----------------	---

5.21.3 Member Function Documentation

5.21.3.1 [getInstance\(\)](#)

```
static PingService& PingService::getInstance ( ) [inline], [static]
```

Gets the instance of this object.

Returns

[PingService](#)&

5.21.3.2 [subscribe\(\)](#)

```
void PingService::subscribe (
    ConnectionBase * connection )
```

Binds the connection to this service.

Parameters

<i>connection</i>	
-------------------	--

5.21.3.3 unsubscribe()

```
void PingService::unsubscribe (
    ConnectionBase * connection )
```

Unbinds the connection from this service.

Parameters

<i>connection</i>	
-------------------	--

5.21.3.4 worker()

```
void PingService::worker (
    int timeout ) [protected]
```

Background task that broadcasts the [Ping](#) packets.

Parameters

<i>timeout</i>	Millisecond interval
----------------	----------------------

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

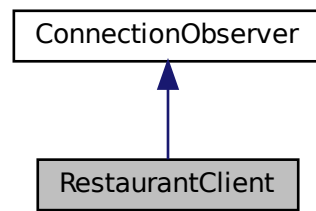
- BaseLibrary/PingService.h
- BaseLibrary/PingService.cpp

5.22 RestaurantClient Class Reference

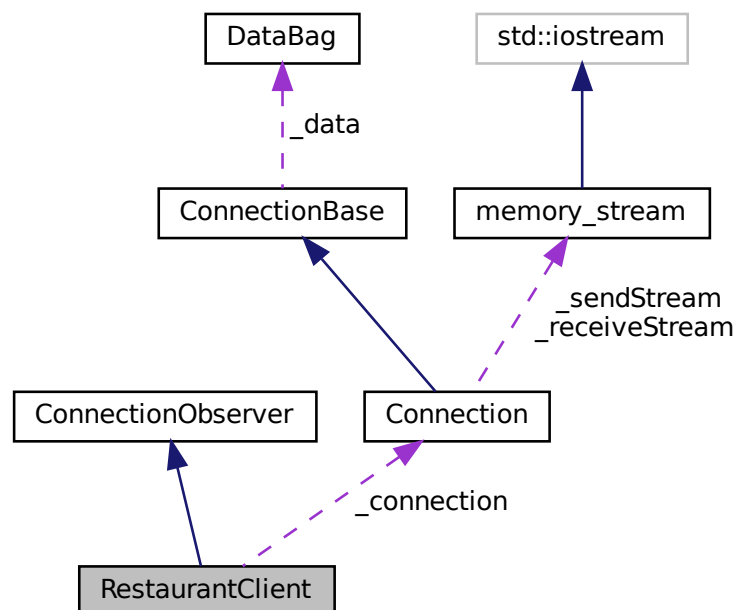
Client interface to access resources provided by the remote [RestaurantManager](#).

```
#include <RestaurantClient.h>
```

Inheritance diagram for RestaurantClient:



Collaboration diagram for RestaurantClient:



Public Member Functions

- `std::map< identity_t, ShiftWorker > & getWorkers ()`
Gets local *ShiftWorker* objects.
- `ShiftWorker & getWorker (identity_t id)`
Gets a local *ShiftWorker* object by its id.
- `void insertShift (const Shift &shift)`
Inserts a *Shift* into the local database.

- void `deleteShift` (identity_t id)
Deletes a [Shift](#) from the local database by its id.
- void `deleteWorker` (identity_t id)
Deletes a [ShiftWorker](#) from the local database by its id.
- [Shift](#) & `getShift` (identity_t id)
Gets a [Shift](#) from the local database by its id.
- std::map< identity_t, [Shift](#) > & `getShifts` ()
Gets local [Shift](#) objects.
- std::set< std::reference_wrapper< [Shift](#) > > `getShifts` (Date day)
Gets all local [Shift](#) objects by [Date](#).
- **RestaurantClient** (std::string host, int port, std::string token="")
- void `setEndpoint` (std::string host, int port)
Sets the endpoint for the remote server.
- const [Connection](#) & `getConnection` () const
Returns the internal [Connection](#) object by const reference.
- [Connection](#) & `getConnection` ()
Returns the internal [Connection](#) object by reference.
- UserPermissions `getPermissions` () const
Gets the granted [UserPermissions](#).
- void `ensureConnected` ()
Reconnects to the remote server if necessary.
- std::string `getToken` () const
Returns the current authorization token.
- void `setToken` (const std::string &token)
Sets the authorization token.
- void `connect` ()
Connects to the remote endpoint.
- void `close` ()
Closes the internal [Connection](#) object.
- bool `isAlive` () const
Returns whether the connection to remote server is active.
- int `authorize` ()
Sends an authorize request to the server with the specified token.
- int `authorize` (std::string newToken)
Sends an authorize request to the server with the specified token.
- int `queryInsertShift` (const [Shift](#) &shift, bool modify=false)
Sends a request to insert the provided [Shift](#) object.
- int `queryUpdateShift` (const [Shift](#) &shift)
Sends a request to update the provided [Shift](#) object.
- int `queryDeleteShift` (identity_t shiftId)
Sends a request to delete a [Shift](#) by its id.
- int `queryShiftsByDay` (const [Date](#) &day)
Sends a request to list all [Shift](#) objects in the specified [Date](#).
- int `queryWorkers` ()
Sends a request to list all [ShiftWorker](#) objects.
- int `queryInsertWorker` (const [ShiftWorker](#) &worker, bool modify=false)
Sends a request to insert the provided [ShiftWorker](#) object.
- int `queryUpdateWorker` (const [ShiftWorker](#) &worker)
Sends a request to update the provided [ShiftWorker](#) object.
- int `queryDeleteWorker` (identity_t workerId)
Sends a request to delete a [ShiftWorker](#) object by its id.

Protected Member Functions

- int [writeRequest](#) ([TrackablePacket](#) &payload)
Sends a request to the server while ensuring the internal [Connection](#) is alive.
- void [onAuthorize](#) ([ConnectionBase](#) *connection, const [S2C_AuthorizeReply](#) &payload, size_t size)
Event handlers.
- void [onDeleteShift](#) ([ConnectionBase](#) *connection, const [S2C_DeleteShiftReply](#) &payload, size_t size)
- void [onGetShiftsByDay](#) ([ConnectionBase](#) *connection, const [S2C_GetShiftsReply](#) &payload, size_t size)
- void [onGetWorkers](#) ([ConnectionBase](#) *connection, const [S2C_GetWorkersReply](#) &payload, size_t size)
- void [onInsertShift](#) ([ConnectionBase](#) *connection, const [S2C_InsertShiftReply](#) &payload, size_t size)
- void [onInsertWorker](#) ([ConnectionBase](#) *connection, const [S2C_InsertWorkerReply](#) &payload, size_t size)
- void [onDeleteWorker](#) ([ConnectionBase](#) *connection, const [S2C_DeleteWorkerReply](#) &payload, size_t size)
- void [onSync](#) ([ConnectionBase](#) *connection, const [S2C_ClientSync](#) &payload, size_t size)

Protected Attributes

- [Connection](#) **_connection**
- std::string **_host**
- int **_port**
- std::string **_token**
- std::map< identity_t, [ShiftWorker](#) > **_workers**
- std::map< identity_t, [Shift](#) > **_shifts**
- std::map< [Date](#), std::set< identity_t > > **_shiftsByDay**
- UserPermissions **_permissions**

5.22.1 Detailed Description

Client interface to access resources provided by the remote [RestaurantManager](#).

5.22.2 Member Function Documentation

5.22.2.1 authorize()

```
int RestaurantClient::authorize (
    std::string newToken )
```

Sends an authorize request to the server with the specified token.

Returns

unique request id

5.22.2.2 queryDeleteShift()

```
int RestaurantClient::queryDeleteShift (
    identity_t shiftId )
```

Sends a request to delete a [Shift](#) by its id.

Returns

unique request id

5.22.2.3 queryDeleteWorker()

```
int RestaurantClient::queryDeleteWorker (
    identity_t workerId )
```

Sends a request to delete a [ShiftWorker](#) object by its id.

Returns

unique request id

5.22.2.4 queryInsertShift()

```
int RestaurantClient::queryInsertShift (
    const Shift & shift,
    bool modify = false )
```

Sends a request to insert the provided [Shift](#) object.

Parameters

<i>shift</i>	Shift object
<i>modify</i>	whether to modify an existing object

Returns

unique request id

5.22.2.5 queryInsertWorker()

```
int RestaurantClient::queryInsertWorker (
    const ShiftWorker & worker,
    bool modify = false )
```

Sends a request to insert the provided [ShiftWorker](#) object.

Parameters

<i>worker</i>	ShiftWorker object
<i>modify</i>	whether to update an existing object

Returns

unique request id

5.22.2.6 queryShiftsByDay()

```
int RestaurantClient::queryShiftsByDay (
    const Date & day )
```

Sends a request to list all [Shift](#) objects in the specified [Date](#).

Returns

unique request id

5.22.2.7 queryUpdateShift()

```
int RestaurantClient::queryUpdateShift (
    const Shift & shift ) [inline]
```

Sends a request to update the provided [Shift](#) object.

Returns

unique request id

5.22.2.8 queryUpdateWorker()

```
int RestaurantClient::queryUpdateWorker (
    const ShiftWorker & worker ) [inline]
```

Sends a request to update the provided [ShiftWorker](#) object.

Parameters

<i>worker</i>	ShiftWorker object
---------------	------------------------------------

Returns

unique request id

5.22.2.9 queryWorkers()

```
int RestaurantClient::queryWorkers ( )
```

Sends a request to list all [ShiftWorker](#) objects.

Returns

unique request id

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

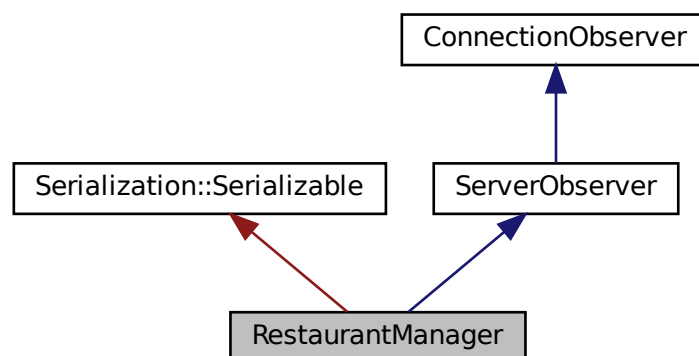
- UIPizzaClient/RestaurantClient.h
- UIPizzaClient/RestaurantClient.cpp

5.23 RestaurantManager Class Reference

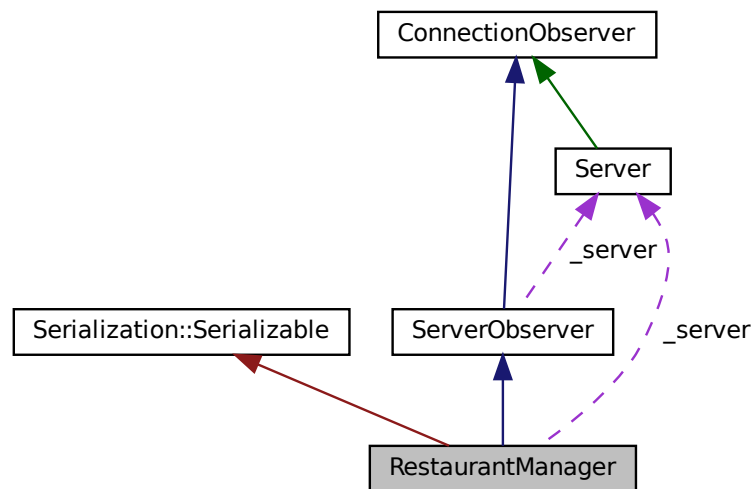
Class that binds to a [Server](#) and provides the [RestaurantManager](#) service.

```
#include <RestaurantManager.h>
```

Inheritance diagram for RestaurantManager:



Collaboration diagram for RestaurantManager:



Public Member Functions

- [Type](#) `getType ()` const override
- **RestaurantManager** ([Server](#) *server)
- void **handlePing** ([ConnectionBase](#) *connection, const [Ping](#) &payload, size_t size)
- void **handleAuthorize** ([ConnectionBase](#) *connection, const [C2S_Authorize](#) &payload, size_t size)
- void **handleGetShiftsByDay** ([ConnectionBase](#) *connection, const [C2S_GetShiftsByDay](#) &payload, size_t size)
- void **handleInsertShift** ([ConnectionBase](#) *connection, const [C2S_InsertShift](#) &payload, size_t size)
- void **handleDeleteShift** ([ConnectionBase](#) *connection, const [C2S_DeleteShift](#) &payload, size_t size)
- void **handleGetWorkers** ([ConnectionBase](#) *connection, const [C2S_GetWorkers](#) &payload, size_t size)
- void **handleInsertWorker** ([ConnectionBase](#) *connection, const [C2S_InsertWorker](#) &payload, size_t size)
- void **handleDeleteWorker** ([ConnectionBase](#) *connection, const [C2S_DeleteWorker](#) &payload, size_t size)
- bool **verifyPermission** ([ConnectionBase](#) *connection, UserPermissions required)
 - Checks the permissions of the connected [Connection](#).*
- virtual std::map< identity_t, [Shift](#) > & **getShifts** ()
 - Gets the [Shift](#) objects by reference.*
- virtual std::map< [Date](#), std::set< identity_t > > & **getShiftsByDay** ()
 - Gets the [Shift](#) objects by day by reference.*
- virtual std::map< identity_t, [ShiftWorker](#) > & **getWorkers** ()
 - Gets the [ShiftWorker](#) objects by reference.*
- virtual std::map< std::string, UserPermissions > & **getAccessTokens** ()
 - Gets the access tokens by reference.*
- virtual [Shift](#) & **insertShift** ([Shift](#) shift, bool modify=false)
 - Insert a [Shift](#) into the database.*
- virtual bool **deleteShift** (identity_t shiftId)
 - Deletes a [Shift](#) from the database.*
- virtual bool **verifyShift** (const [Shift](#) &shift)
 - Checks if the [Shift](#) does not collide with others.*

- virtual [ShiftWorker](#) & [insertWorker](#) ([ShiftWorker](#) worker, bool modify=false)
Insert a [ShiftWorker](#) into the database.
- virtual bool [deleteWorker](#) (identity_t workerId)
Deletes a [ShiftWorker](#) from the database.
- std::ostream & [serialize](#) (std::ostream &dst) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &src) override
Deserializes this object from a binary input stream.

Static Public Member Functions

- template<typename T >
static identity_t [newIdentityId](#) (const std::map< identity_t, T > &map)
Generates a new identity id for the given collection.

Protected Member Functions

- void [onPayloadReceived](#) ([ConnectionBase](#) *connection, const [Serializable](#) &payload, size_t size) override
Event handler for payload received.

Protected Attributes

- [Server](#) * **_server**
- std::recursive_mutex **_lock**
- std::map< identity_t, [Shift](#) > **_shifts**
- std::map< [Date](#), std::set< identity_t > > **_shiftsByDay**
- std::map< identity_t, [ShiftWorker](#) > **_workers**
- std::map< std::string, UserPermissions > **_accessTokens**

5.23.1 Detailed Description

Class that binds to a [Server](#) and provides the [RestaurantManager](#) service.

5.23.2 Member Function Documentation

5.23.2.1 deleteShift()

```
bool RestaurantManager::deleteShift (
    identity_t shiftId ) [virtual]
```

Deletes a [Shift](#) from the database.

Parameters

<i>shift↔ Id</i>	id of the Shift
----------------------	---------------------------------

Returns

whether an object was removed

5.23.2.2 deleteWorker()

```
bool RestaurantManager::deleteWorker (
    identity_t workerId ) [virtual]
```

Deletes a [ShiftWorker](#) from the database.

Parameters

<i>worker↔ Id</i>	id of the ShiftWorker
-----------------------	---------------------------------------

Returns

whether an object was removed

5.23.2.3 deserialize()

```
std::istream & RestaurantManager::deserialize (
    std::istream & source ) [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.23.2.4 getAccessTokens()

```
virtual std::map<std::string, UserPermissions>& RestaurantManager::getAccessTokens ( ) [inline],  
[virtual]
```

Gets the access tokens by reference.

Returns

`std::map<std::string, UserPermissions>&`

5.23.2.5 getShifts()

```
virtual std::map<identity_t, Shift>& RestaurantManager::getShifts ( ) [inline], [virtual]
```

Gets the [Shift](#) objects by reference.

Returns

`std::map<identity_t, Shift>&`

5.23.2.6 getShiftsByDay()

```
virtual std::map<Date, std::set<identity_t> >& RestaurantManager::getShiftsByDay ( ) [inline],  
[virtual]
```

Gets the [Shift](#) objects by day by reference.

Returns

`std::map<Date, std::set<identity_t>>&`

5.23.2.7 getType()

```
Type RestaurantManager::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.23.2.8 getWorkers()

```
virtual std::map<identity_t, ShiftWorker>& RestaurantManager::getWorkers ( ) [inline], [virtual]
```

Gets the [ShiftWorker](#) objects by reference.

Returns

`std::map<identity_t, ShiftWorker>&`

5.23.2.9 insertShift()

```
Shift & RestaurantManager::insertShift (
    Shift shift,
    bool modify = false ) [virtual]
```

Insert a [Shift](#) into the database.

Parameters

<i>shift</i>	object to be inserted
<i>modify</i>	whether to modify an existing record

Returns

[Shift](#)& reference to the inserted object

5.23.2.10 insertWorker()

```
ShiftWorker & RestaurantManager::insertWorker (
    ShiftWorker worker,
    bool modify = false ) [virtual]
```

Insert a [ShiftWorker](#) into the database.

Parameters

<i>worker</i>	object to be inserted
<i>modify</i>	whether to modify an existing record

Returns

[ShiftWorker](#)& reference to the inserted object

5.23.2.11 newIdentityId()

```
template<typename T >
static identity_t RestaurantManager::newIdentityId (
    const std::map< identity_t, T > & map ) [inline], [static]
```

Generates a new identity id for the given collection.

Template Parameters

<i>T</i>	collection type
----------	-----------------

Parameters

<i>map</i>	collection
------------	------------

Returns

identity_t new unique identity id

5.23.2.12 onPayloadReceived()

```
void RestaurantManager::onPayloadReceived (
    ConnectionBase * connection,
    const Serializable & payload,
    size_t size ) [inline], [override], [protected], [virtual]
```

Event handler for payload received.

Parameters

<i>connection</i>	event source
<i>payload</i>	received payload
<i>size</i>	size (in bytes) of received payload

Reimplemented from [ConnectionObserver](#).

5.23.2.13 serialize()

```
std::ostream & RestaurantManager::serialize (
    std::ostream & destination ) const [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.23.2.14 verifyPermission()

```
bool RestaurantManager::verifyPermission (
    ConnectionBase * connection,
    UserPermissions required )
```

Checks the permissions of the connected [Connection](#).

Parameters

<i>connection</i>	client
<i>required</i>	required permissions

Returns

whether the client has all the required permissions

5.23.2.15 verifyShift()

```
bool RestaurantManager::verifyShift (
    const Shift & shift ) [virtual]
```

Checks if the [Shift](#) does not collide with others.

Parameters

<i>shift</i>	object to be queried
--------------	----------------------

Returns

whether the shift can be inserted

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

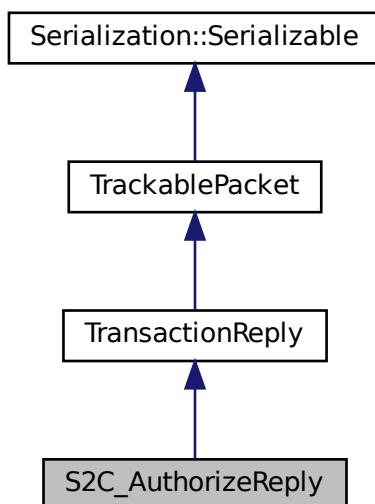
- BaseLibrary/RestaurantManager.h
- BaseLibrary/RestaurantManager.cpp

5.24 S2C_AuthorizeReply Class Reference

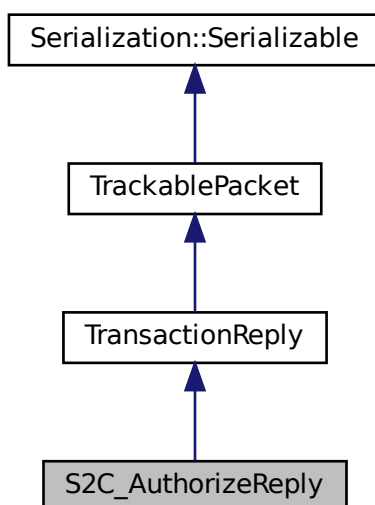
Server-to-client reply to a token-based authorization request.

```
#include <S2C_AuthorizeReply.h>
```

Inheritance diagram for S2C_AuthorizeReply:



Collaboration diagram for S2C_AuthorizeReply:



Public Member Functions

- [Type](#) [getType](#) () const override
- [S2C_AuthorizeReply](#) (int requestId)
Construct a new authorize response.
- [S2C_AuthorizeReply](#) (int requestId, bool success, UserPermissions permissions, const std::string &msg="")
Construct a new authorize response.
- UserPermissions [getPermissions](#) () const
Gets the permissions granted.
- void [setPermissions](#) (UserPermissions permissions)
Sets the permissions granted.
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- UserPermissions [_permissions](#)

Friends

- bool **operator==** (const [S2C_AuthorizeReply](#) &lhs, const [S2C_AuthorizeReply](#) &rhs)
- bool **operator!=** (const [S2C_AuthorizeReply](#) &lhs, const [S2C_AuthorizeReply](#) &rhs)

5.24.1 Detailed Description

Server-to-client reply to a token-based authorization request.

5.24.2 Constructor & Destructor Documentation

5.24.2.1 [S2C_AuthorizeReply](#)() [1/2]

```
S2C_AuthorizeReply::S2C_AuthorizeReply (
    int requestId ) [inline]
```

Construct a new authorize response.

Parameters

<i>requestId</i>	request id
------------------	------------

5.24.2.2 S2C_AuthorizeReply() [2/2]

```
S2C_AuthorizeReply::S2C_AuthorizeReply (
    int requestId,
    bool success,
    UserPermissions permissions,
    const std::string & msg = "" ) [inline]
```

Construct a new authorize response.

Parameters

<i>requestId</i>	request id
<i>success</i>	whether authorization succeeded
<i>permissions</i>	permissions granted
<i>msg</i>	error message

5.24.3 Member Function Documentation

5.24.3.1 deserialize()

```
std::istream& S2C_AuthorizeReply::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.24.3.2 getPermissions()

```
UserPermissions S2C_AuthorizeReply::getPermissions ( ) const [inline]
```

Gets the permissions granted.

Returns

UserPermissions permissions

5.24.3.3 `getType()`

```
Type S2C_AuthorizeReply::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.24.3.4 `serialize()`

```
std::ostream& S2C_AuthorizeReply::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.24.3.5 `setPermissions()`

```
void S2C_AuthorizeReply::setPermissions (
    UserPermissions permissions ) [inline]
```

Sets the permissions granted.

Parameters

<i>permissions</i>	permissions
--------------------	-------------

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

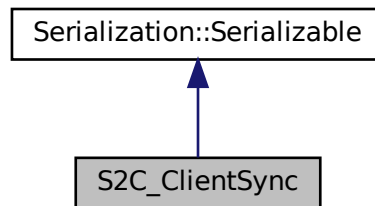
- BaseLibrary/S2C_AuthorizeReply.h

5.25 S2C_ClientSync Class Reference

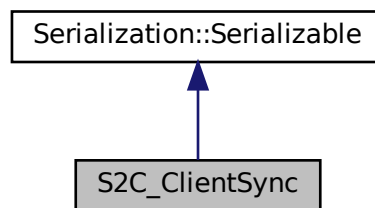
Server-to-client broadcast packet that updates the client state.

```
#include <S2C_ClientSync.h>
```

Inheritance diagram for S2C_ClientSync:



Collaboration diagram for S2C_ClientSync:



Public Member Functions

- [Type](#) [getType](#) () const override
- std::set< identity_t > [getRemovedWorkers](#) () const
Gets the removed worker ids.
- std::set< identity_t > [getRemovedShifts](#) () const
Gets the removed shift ids.
- std::set< identity_t > & [getRemovedWorkers](#) ()
Gets the removed worker ids by reference.
- std::set< identity_t > & [getRemovedShifts](#) ()
Gets the removed shift ids by reference.
- std::set< [ShiftWorker](#) > [getChangedWorkers](#) () const
Gets the set of changed [ShiftWorker](#) objects.
- std::set< [Shift](#) > [getChangedShifts](#) () const

Gets the set of changed [Shift](#) objects.

- `std::set< ShiftWorker > & getChangedWorkers ()`

Gets the set of changed [ShiftWorker](#) objects by reference.

- `std::set< Shift > & getChangedShifts ()`

Gets the set of changed [Shift](#) objects by reference.

- `std::ostream & serialize (std::ostream &dst) const` override

Serializes this object into a binary output stream.

- `std::istream & deserialize (std::istream &src) override`

Deserializes this object from a binary input stream.

Protected Attributes

- `std::set< identity_t > _removedWorkers`
- `std::set< identity_t > _removedShifts`
- `std::set< ShiftWorker > _changedWorkers`
- `std::set< Shift > _changedShifts`

Friends

- `bool operator== (const S2C_ClientSync &lhs, const S2C_ClientSync &rhs)`
- `bool operator!= (const S2C_ClientSync &lhs, const S2C_ClientSync &rhs)`

5.25.1 Detailed Description

Server-to-client broadcast packet that updates the client state.

5.25.2 Member Function Documentation

5.25.2.1 `deserialize()`

```
std::istream& S2C_ClientSync::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.25.2.2 getChangedShifts() [1/2]

```
std::set<Shift> S2C_ClientSync::getChangedShifts ( ) const [inline]
```

Gets the set of changed [Shift](#) objects.

Returns

std::set<Shift>

5.25.2.3 getChangedShifts() [2/2]

```
std::set<Shift>& S2C_ClientSync::getChangedShifts ( ) [inline]
```

Gets the set of changed [Shift](#) objects by reference.

Returns

std::set<Shift>&

5.25.2.4 getChangedWorkers() [1/2]

```
std::set<ShiftWorker> S2C_ClientSync::getChangedWorkers ( ) const [inline]
```

Gets the set of changed [ShiftWorker](#) objects.

Returns

std::set<ShiftWorker>

5.25.2.5 getChangedWorkers() [2/2]

```
std::set<ShiftWorker>& S2C_ClientSync::getChangedWorkers ( ) [inline]
```

Gets the set of changed [ShiftWorker](#) objects by reference.

Returns

std::set<ShiftWorker>&

5.25.2.6 getRemovedShifts() [1/2]

```
std::set<identity_t> S2C_ClientSync::getRemovedShifts ( ) const [inline]
```

Gets the removed shift ids.

Returns

`std::set<identity_t>`

5.25.2.7 getRemovedShifts() [2/2]

```
std::set<identity_t>& S2C_ClientSync::getRemovedShifts ( ) [inline]
```

Gets the removed shift ids by reference.

Returns

`std::set<identity_t>&`

5.25.2.8 getRemovedWorkers() [1/2]

```
std::set<identity_t> S2C_ClientSync::getRemovedWorkers ( ) const [inline]
```

Gets the removed worker ids.

Returns

`std::set<identity_t>`

5.25.2.9 getRemovedWorkers() [2/2]

```
std::set<identity_t>& S2C_ClientSync::getRemovedWorkers ( ) [inline]
```

Gets the removed worker ids by reference.

Returns

`std::set<identity_t>&`

5.25.2.10 getType()

```
Type S2C_ClientSync::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.25.2.11 serialize()

```
std::ostream& S2C_ClientSync::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

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

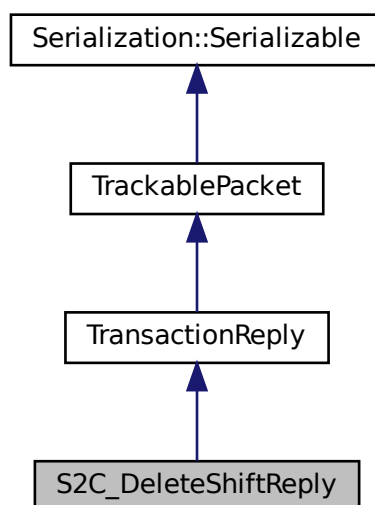
- BaseLibrary/S2C_ClientSync.h

5.26 S2C_DeleteShiftReply Class Reference

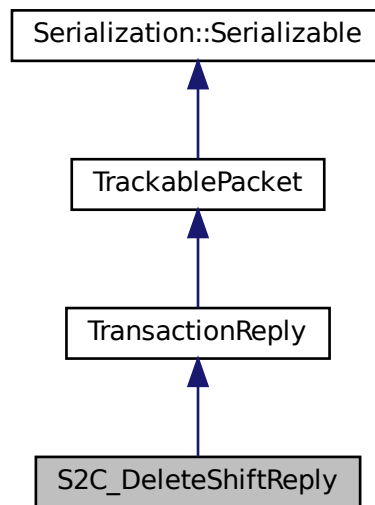
Server-to-client response to deleting a [Shift](#) object.

```
#include <S2C_DeleteShiftReply.h>
```

Inheritance diagram for S2C_DeleteShiftReply:



Collaboration diagram for S2C_DeleteShiftReply:



Public Member Functions

- [Type](#) `getType ()` const override
- virtual `identity_t getId ()` const
Gets the [Shift](#) id.
- virtual void `setId (identity_t id)`
Sets the [Shift](#) id.
- [S2C_DeleteShiftReply](#) (int requestId, identity_t shiftId)
Construct a new response to deleting a [Shift](#) object.
- `std::ostream & serialize (std::ostream &destination)` const override
Serializes this object into a binary output stream.
- `std::istream & deserialize (std::istream &source)` override
Deserializes this object from a binary input stream.

Protected Attributes

- `identity_t _id`

Friends

- `bool operator== (const S2C_DeleteShiftReply &lhs, const S2C_DeleteShiftReply &rhs)`
- `bool operator!= (const S2C_DeleteShiftReply &lhs, const S2C_DeleteShiftReply &rhs)`

5.26.1 Detailed Description

Server-to-client response to deleting a [Shift](#) object.

5.26.2 Constructor & Destructor Documentation

5.26.2.1 S2C_DeleteShiftReply()

```
S2C_DeleteShiftReply::S2C_DeleteShiftReply (
    int requestId,
    identity_t shiftId ) [inline]
```

Construct a new response to deleting a [Shift](#) object.

Parameters

<i>requestId</i>	request id
<i>shiftId</i>	shift id

5.26.3 Member Function Documentation

5.26.3.1 deserialize()

```
std::istream& S2C_DeleteShiftReply::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.26.3.2 getId()

```
virtual identity_t S2C_DeleteShiftReply::getId ( ) const [inline], [virtual]
```

Gets the [Shift](#) id.

Returns

identity_t

5.26.3.3 `getType()`

```
Type S2C_DeleteShiftReply::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.26.3.4 `serialize()`

```
std::ostream& S2C_DeleteShiftReply::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.26.3.5 `setId()`

```
virtual void S2C_DeleteShiftReply::setId (
    identity_t id ) [inline], [virtual]
```

Sets the [Shift](#) id.

Parameters

<i>id</i>	
-----------	--

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

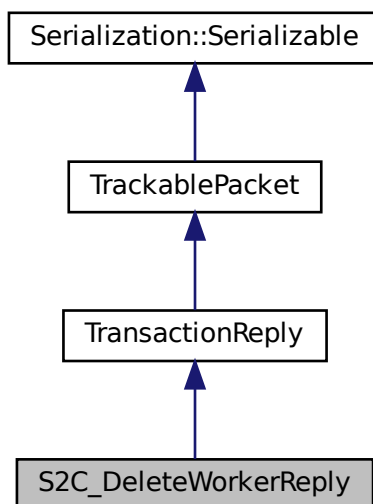
- BaseLibrary/S2C_DeleteShiftReply.h

5.27 S2C_DeleteWorkerReply Class Reference

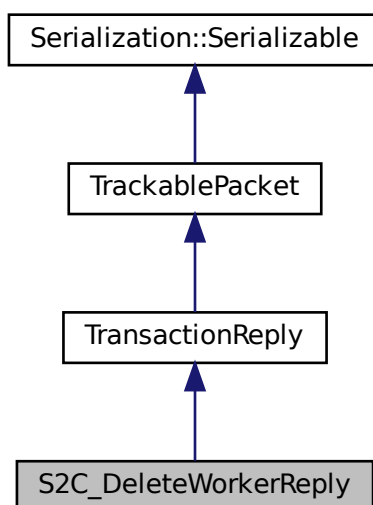
Server-to-client response to deleting a [ShiftWorker](#) object.

```
#include <S2C_DeleteWorkerReply.h>
```

Inheritance diagram for S2C_DeleteWorkerReply:



Collaboration diagram for S2C_DeleteWorkerReply:



Public Member Functions

- [Type](#) [getType](#) () const override
- virtual [identity_t](#) [getId](#) () const
Gets the [ShiftWorker](#) id.
- virtual void [setId](#) ([identity_t](#) id)
Sets the [ShiftWorker](#) id.
- [S2C_DeleteWorkerReply](#) (int requestId, [identity_t](#) id)
Construct a new response to [ShiftWorker](#) deletion.
- [std::ostream](#) & [serialize](#) ([std::ostream](#) &destination) const override
Serializes this object into a binary output stream.
- [std::istream](#) & [deserialize](#) ([std::istream](#) &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- [identity_t](#) [_id](#)

Friends

- bool **operator==** (const [S2C_DeleteWorkerReply](#) &lhs, const [S2C_DeleteWorkerReply](#) &rhs)
- bool **operator!=** (const [S2C_DeleteWorkerReply](#) &lhs, const [S2C_DeleteWorkerReply](#) &rhs)

5.27.1 Detailed Description

Server-to-client response to deleting a [ShiftWorker](#) object.

5.27.2 Constructor & Destructor Documentation

5.27.2.1 S2C_DeleteWorkerReply()

```
S2C_DeleteWorkerReply::S2C_DeleteWorkerReply (
    int requestId,
    identity_t id ) [inline]
```

Construct a new response to [ShiftWorker](#) deletion.

Parameters

<i>requestId</i>	request id
<i>id</i>	ShiftWorker id

5.27.3 Member Function Documentation

5.27.3.1 deserialize()

```
std::istream& S2C_DeleteWorkerReply::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.27.3.2 getId()

```
virtual identity_t S2C_DeleteWorkerReply::getId ( ) const [inline], [virtual]
```

Gets the [ShiftWorker](#) id.

Returns

identity_t

5.27.3.3 getType()

```
Type S2C_DeleteWorkerReply::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.27.3.4 serialize()

```
std::ostream& S2C_DeleteWorkerReply::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.27.3.5 setId()

```
virtual void S2C_DeleteWorkerReply::setId (  
    identity_t id ) [inline], [virtual]
```

Sets the [ShiftWorker](#) id.

Parameters

<i>id</i>	
-----------	--

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

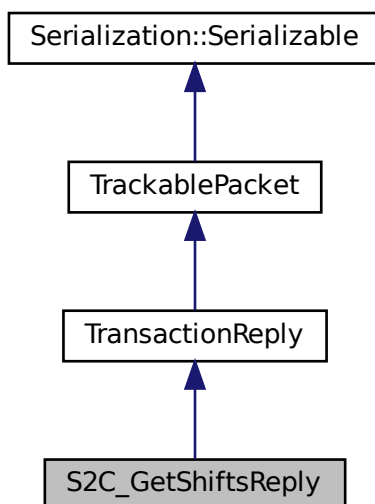
- BaseLibrary/S2C_DeleteWorkerReply.h

5.28 S2C_GetShiftsReply Class Reference

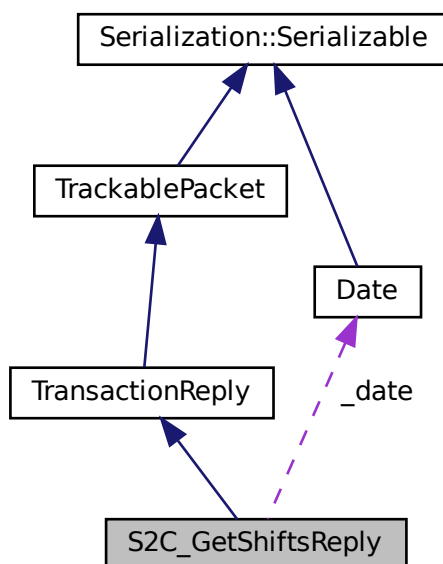
Server-to-client response that lists all [Shift](#) objects in the given day.

```
#include <S2C_GetShiftsReply.h>
```

Inheritance diagram for S2C_GetShiftsReply:



Collaboration diagram for S2C_GetShiftsReply:



Public Member Functions

- `Type` `getType` () const override

- [S2C_GetShiftsReply](#) (int requestId, [Date](#) date)
Construct a new response that lists all Shifts in the given day.
- std::set< [Shift](#) > [getShifts](#) () const
Gets the Shifts in the queried day.
- std::set< [Shift](#) > & [getShifts](#) ()
Gets the Shifts in the queried day by reference.
- void [setShifts](#) (const std::set< [Shift](#) > &shifts)
Sets the Shifts.
- [Date](#) [getDate](#) () const
Gets the queried date.
- [Date](#) & [getDate](#) ()
Gets the queried date by reference.
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- std::set< [Shift](#) > **_shifts**
- [Date](#) **_date**

Friends

- bool **operator==** (const [S2C_GetShiftsReply](#) &lhs, const [S2C_GetShiftsReply](#) &rhs)
- bool **operator!=** (const [S2C_GetShiftsReply](#) &lhs, const [S2C_GetShiftsReply](#) &rhs)

5.28.1 Detailed Description

Server-to-client response that lists all [Shift](#) objects in the given day.

5.28.2 Constructor & Destructor Documentation

5.28.2.1 S2C_GetShiftsReply()

```
S2C_GetShiftsReply::S2C_GetShiftsReply (
    int requestId,
    Date date ) [inline]
```

Construct a new response that lists all Shifts in the given day.

Parameters

<i>requestId</i>	request id
<i>date</i>	queried day

5.28.3 Member Function Documentation

5.28.3.1 deserialize()

```
std::istream& S2C_GetShiftsReply::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.28.3.2 getDate() [1/2]

```
Date S2C_GetShiftsReply::getDate ( ) const [inline]
```

Gets the queried date.

Returns

[Date](#)

5.28.3.3 getDate() [2/2]

```
Date& S2C_GetShiftsReply::getDate ( ) [inline]
```

Gets the queried date by reference.

Returns

[Date](#)&

5.28.3.4 getShifts() [1/2]

```
std::set<Shift> S2C_GetShiftsReply::getShifts ( ) const [inline]
```

Gets the Shifts in the queried day.

Returns

std::set<Shift>

5.28.3.5 getShifts() [2/2]

```
std::set<Shift>& S2C_GetShiftsReply::getShifts ( ) [inline]
```

Gets the Shifts in the queried day by reference.

Returns

std::set<Shift>&

5.28.3.6 getType()

```
Type S2C_GetShiftsReply::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.28.3.7 serialize()

```
std::ostream& S2C_GetShiftsReply::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.28.3.8 setShifts()

```
void S2C_GetShiftsReply::setShifts (
    const std::set< Shift > & shifts ) [inline]
```

Sets the Shifts.

Parameters

<i>shifts</i>	
---------------	--

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

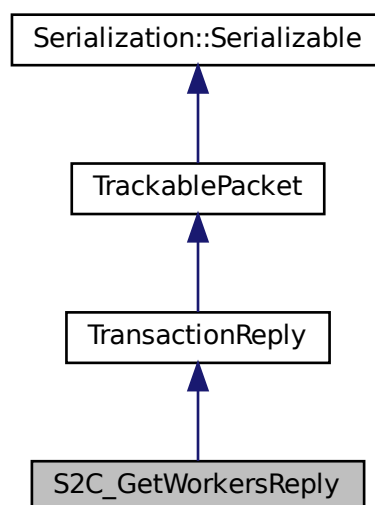
- BaseLibrary/S2C_GetShiftsReply.h

5.29 S2C_GetWorkersReply Class Reference

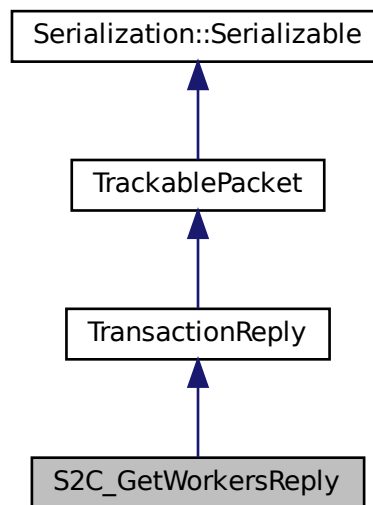
Server-to-client response that lists all [ShiftWorker](#) objects in the database.

```
#include <S2C_GetWorkersReply.h>
```

Inheritance diagram for S2C_GetWorkersReply:



Collaboration diagram for S2C_GetWorkersReply:



Public Member Functions

- [Type](#) [getType](#) () const override
- [S2C_GetWorkersReply](#) (int requestId)
Construct a new response that lists all ShiftWorkers.
- std::map< identity_t, [ShiftWorker](#) > [getWorkers](#) () const
Gets a copy of all ShiftWorkers.
- std::map< identity_t, [ShiftWorker](#) > & [getWorkers](#) ()
Gets all ShiftWorkers by reference.
- void [setWorkers](#) (std::map< identity_t, [ShiftWorker](#) > workers)
Sets the set of ShiftWorkers.
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- std::map< identity_t, [ShiftWorker](#) > [_workers](#)

Friends

- bool [operator==](#) (const [S2C_GetWorkersReply](#) &lhs, const [S2C_GetWorkersReply](#) &rhs)
- bool [operator!=](#) (const [S2C_GetWorkersReply](#) &lhs, const [S2C_GetWorkersReply](#) &rhs)

5.29.1 Detailed Description

Server-to-client response that lists all [ShiftWorker](#) objects in the database.

5.29.2 Constructor & Destructor Documentation

5.29.2.1 S2C_GetWorkersReply()

```
S2C_GetWorkersReply::S2C_GetWorkersReply (
    int requestId ) [inline]
```

Construct a new response that lists all ShiftWorkers.

Parameters

<i>requestId</i>	request id
------------------	------------

5.29.3 Member Function Documentation

5.29.3.1 deserialize()

```
std::istream& S2C_GetWorkersReply::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.29.3.2 getType()

```
Type S2C_GetWorkersReply::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.29.3.3 getWorkers() [1/2]

```
std::map<identity_t, ShiftWorker> S2C_GetWorkersReply::getWorkers ( ) const [inline]
```

Gets a copy of all ShiftWorkers.

Returns

std::map<identity_t, ShiftWorker> copy of all ShiftWorkers

5.29.3.4 getWorkers() [2/2]

```
std::map<identity_t, ShiftWorker>& S2C_GetWorkersReply::getWorkers ( ) [inline]
```

Gets all ShiftWorkers by reference.

Returns

std::map<identity_t, ShiftWorker>& reference to all ShiftWorkers

5.29.3.5 serialize()

```
std::ostream& S2C_GetWorkersReply::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.29.3.6 setWorkers()

```
void S2C_GetWorkersReply::setWorkers (
    std::map< identity_t, ShiftWorker > workers ) [inline]
```

Sets the set of ShiftWorkers.

Parameters

<i>workers</i>	set of ShiftWorkers
----------------	---------------------

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

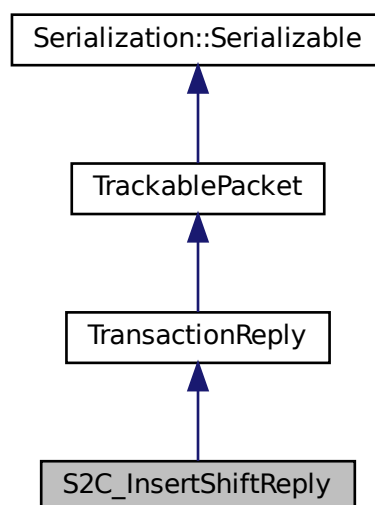
- BaseLibrary/S2C_GetWorkersReply.h

5.30 S2C_InsertShiftReply Class Reference

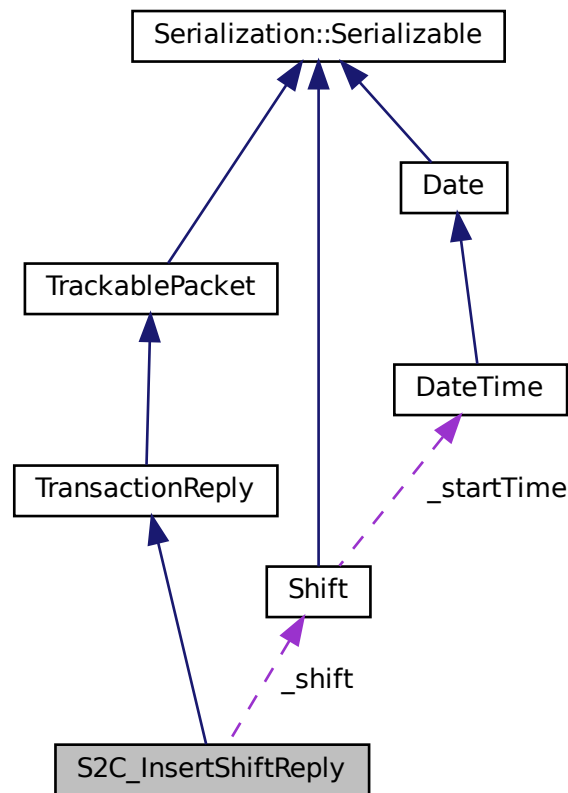
Server-to-client response to insert a [Shift](#) object.

```
#include <S2C_InsertShiftReply.h>
```

Inheritance diagram for S2C_InsertShiftReply:



Collaboration diagram for S2C_InsertShiftReply:



Public Member Functions

- [Type](#) [getType](#) () const override
- virtual [Shift](#) & [getShift](#) ()
Gets the queried [Shift](#) object by reference.
- virtual [Shift](#) [getShift](#) () const
Gets a copy of the queried [Shift](#) object.
- virtual void [setShift](#) (const [Shift](#) &shift)
Sets the queried [Shift](#) object.
- [S2C_InsertShiftReply](#) (int requestId, const [Shift](#) &shift)
Construct a new response to insert a [Shift](#) object.
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- [Shift](#) [_shift](#)

Friends

- bool **operator==** (const [S2C_InsertShiftReply](#) &lhs, const [S2C_InsertShiftReply](#) &rhs)
- bool **operator!=** (const [S2C_InsertShiftReply](#) &lhs, const [S2C_InsertShiftReply](#) &rhs)

5.30.1 Detailed Description

Server-to-client response to insert a [Shift](#) object.

5.30.2 Constructor & Destructor Documentation

5.30.2.1 S2C_InsertShiftReply()

```
S2C_InsertShiftReply::S2C_InsertShiftReply (
    int requestId,
    const Shift & shift ) [inline]
```

Construct a new response to insert a [Shift](#) object.

Parameters

<i>requestId</i>	request id
<i>shift</i>	queried object

5.30.3 Member Function Documentation

5.30.3.1 deserialize()

```
std::istream& S2C_InsertShiftReply::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.30.3.2 getShift() [1/2]

```
virtual Shift& S2C_InsertShiftReply::getShift ( ) [inline], [virtual]
```

Gets the queried [Shift](#) object by reference.

Returns

[Shift](#)&

5.30.3.3 getShift() [2/2]

```
virtual Shift S2C_InsertShiftReply::getShift ( ) const [inline], [virtual]
```

Gets a copy of the queried [Shift](#) object.

Returns

[Shift](#)

5.30.3.4 getType()

```
Type S2C_InsertShiftReply::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.30.3.5 serialize()

```
std::ostream& S2C_InsertShiftReply::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.30.3.6 setShift()

```
virtual void S2C_InsertShiftReply::setShift (
    const Shift & shift ) [inline], [virtual]
```

Sets the queried [Shift](#) object.

Parameters

<i>shift</i>	
--------------	--

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

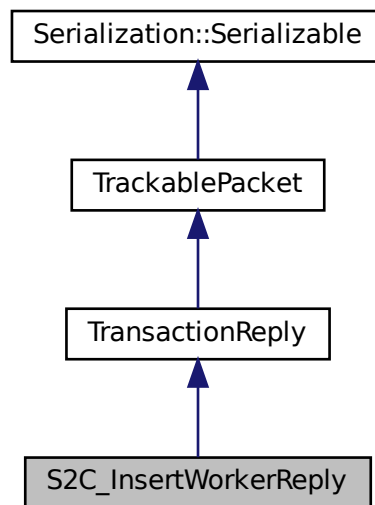
- BaseLibrary/S2C_InsertShiftReply.h

5.31 S2C_InsertWorkerReply Class Reference

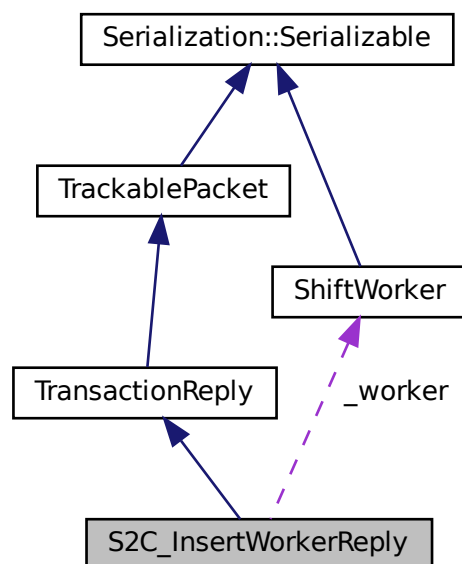
Server-to-client response to insert a [ShiftWorker](#) object.

```
#include <S2C_InsertWorkerReply.h>
```

Inheritance diagram for S2C_InsertWorkerReply:



Collaboration diagram for S2C_InsertWorkerReply:



Public Member Functions

- [Type](#) `getType` () const override

- virtual [ShiftWorker](#) & [getWorker](#) ()
Gets the queried [ShiftWorker](#) by reference.
- virtual [ShiftWorker](#) [getWorker](#) () const
Gets a copy of the queried [ShiftWorker](#).
- virtual void [setWorker](#) (const [ShiftWorker](#) &worker)
Sets the queried [ShiftWorker](#).
- [S2C_InsertWorkerReply](#) (int requestId, [ShiftWorker](#) worker)
Construct a new response to insert a [ShiftWorker](#).
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- [ShiftWorker](#) _worker

Friends

- bool **operator==** (const [S2C_InsertWorkerReply](#) &lhs, const [S2C_InsertWorkerReply](#) &rhs)
- bool **operator!=** (const [S2C_InsertWorkerReply](#) &lhs, const [S2C_InsertWorkerReply](#) &rhs)

5.31.1 Detailed Description

Server-to-client response to insert a [ShiftWorker](#) object.

5.31.2 Constructor & Destructor Documentation

5.31.2.1 [S2C_InsertWorkerReply](#)()

```
S2C_InsertWorkerReply::S2C_InsertWorkerReply (
    int requestId,
    ShiftWorker worker ) [inline]
```

Construct a new response to insert a [ShiftWorker](#).

Parameters

<i>requestId</i>	request id
<i>worker</i>	queried ShiftWorker

5.31.3 Member Function Documentation

5.31.3.1 deserialize()

```
std::istream& S2C_InsertWorkerReply::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.31.3.2 getType()

```
Type S2C_InsertWorkerReply::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.31.3.3 getWorker() [1/2]

```
virtual ShiftWorker& S2C_InsertWorkerReply::getWorker ( ) [inline], [virtual]
```

Gets the queried [ShiftWorker](#) by reference.

Returns

[ShiftWorker](#)&

5.31.3.4 `getWorker()` [2/2]

```
virtual ShiftWorker S2C_InsertWorkerReply::getWorker ( ) const [inline], [virtual]
```

Gets a copy of the queried [ShiftWorker](#).

Returns

[ShiftWorker](#)

5.31.3.5 `serialize()`

```
std::ostream& S2C_InsertWorkerReply::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.31.3.6 `setWorker()`

```
virtual void S2C_InsertWorkerReply::setWorker (
    const ShiftWorker & worker ) [inline], [virtual]
```

Sets the queried [ShiftWorker](#).

Parameters

<i>worker</i>	
---------------	--

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

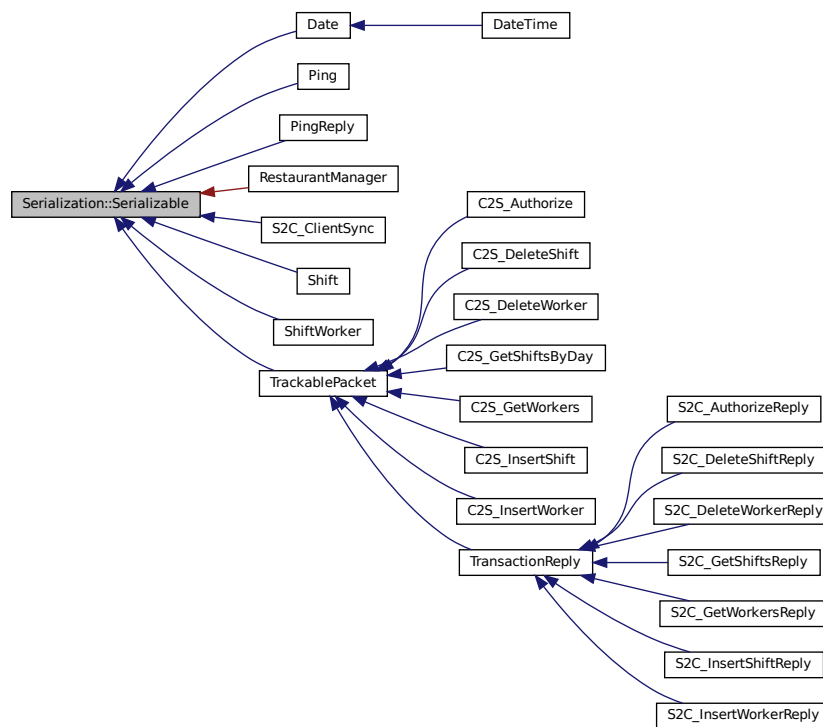
- BaseLibrary/S2C_InsertWorkerReply.h

5.32 Serialization::Serializable Class Reference

Base-class for all serializable objects.

```
#include <Serializable.h>
```

Inheritance diagram for Serialization::Serializable:



Public Member Functions

- virtual [Type](#) [getType](#) () const =0
- virtual std::ostream & [serialize](#) (std::ostream &destination) const
Serializes this object into a binary output stream.
- virtual std::istream & [deserialize](#) (std::istream &source)
Deserializes this object from a binary input stream.

Friends

- bool **operator==** (const [Serializable](#) &lhs, const [Serializable](#) &rhs)
- bool **operator!=** (const [Serializable](#) &lhs, const [Serializable](#) &rhs)

5.32.1 Detailed Description

Base-class for all serializable objects.

5.32.2 Member Function Documentation

5.32.2.1 deserialize()

```
virtual std::istream& Serialization::Serializable::deserialize (
    std::istream & source ) [inline], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented in [RestaurantManager](#), [Shift](#), [DateTime](#), [ShiftWorker](#), [Date](#), [C2S_InsertWorker](#), [TransactionReply](#), [C2S_InsertShift](#), [S2C_ClientSync](#), [S2C_GetShiftsReply](#), [S2C_InsertWorkerReply](#), [S2C_GetWorkersReply](#), [S2C_↵_InsertShiftReply](#), [S2C_AuthorizeReply](#), [C2S_GetShiftsByDay](#), [S2C_DeleteShiftReply](#), [S2C_DeleteWorkerReply](#), [C2S_DeleteShift](#), [C2S_Authorize](#), [C2S_DeleteWorker](#), and [TrackablePacket](#).

5.32.2.2 getType()

```
virtual Type Serialization::Serializable::getType ( ) const [pure virtual]
```

Returns

Type of this serializable object

Implemented in [ShiftWorker](#), [RestaurantManager](#), [Shift](#), [S2C_GetShiftsReply](#), [S2C_ClientSync](#), [Date](#), [DateTime](#), [C2S_InsertWorker](#), [S2C_InsertWorkerReply](#), [S2C_GetWorkersReply](#), [C2S_InsertShift](#), [S2C_AuthorizeReply](#), [C2↵S_DeleteShift](#), [C2S_GetShiftsByDay](#), [C2S_Authorize](#), [S2C_InsertShiftReply](#), [C2S_DeleteWorker](#), [S2C_Delete↵ShiftReply](#), [S2C_DeleteWorkerReply](#), [C2S_GetWorkers](#), [Ping](#), and [PingReply](#).

5.32.2.3 serialize()

```
virtual std::ostream& Serialization::Serializable::serialize (
    std::ostream & destination ) const [inline], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

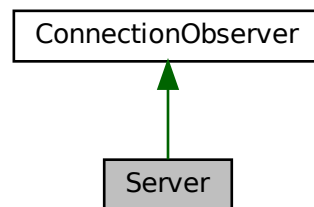
Reimplemented in [RestaurantManager](#), [Shift](#), [DateTime](#), [ShiftWorker](#), [Date](#), [C2S_InsertWorker](#), [TransactionReply](#), [C2S_InsertShift](#), [S2C_GetShiftsReply](#), [S2C_ClientSync](#), [S2C_InsertWorkerReply](#), [S2C_AuthorizeReply](#), [S2C_InsertShiftReply](#), [C2S_GetShiftsByDay](#), [S2C_GetWorkersReply](#), [S2C_DeleteShiftReply](#), [S2C_DeleteWorkerReply](#), [C2S_DeleteShift](#), [C2S_Authorize](#), [C2S_DeleteWorker](#), and [TrackablePacket](#).

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

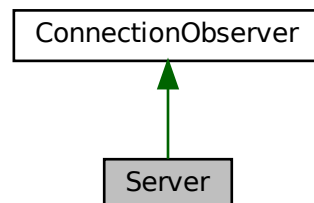
- BaseLibrary/Serializable.h

5.33 Server Class Reference

Inheritance diagram for Server:



Collaboration diagram for Server:



Public Member Functions

- virtual std::map< int, [ConnectionBase](#) * > [clients](#) () const
Gets all connected clients.
- virtual bool [isRunning](#) () const
- virtual void [start](#) (const std::string &host, int port)
Starts listening for incoming connections on the specified endpoint.
- virtual void [stop](#) ()
Stops listening for incoming connections and drops all clients.
- virtual void [writeToAll](#) (const [Serializable](#) &msg)
Sends a packet to all connected clients.
- virtual void [join](#) ()
Joins the listener thread.
- virtual void [subscribe](#) ([ServerObserver](#) *observer)
Subscribes to the events of this server object.
- virtual void [unsubscribe](#) ([ServerObserver](#) *observer)
Unsubscribes from the events of this server object.

Protected Member Functions

- virtual bool [shouldAcceptSocket](#) (PSocket *socket)
Predicate that should decide whether to accept the socket connection.
- virtual void [prepareClient](#) (PSocket *socket)
Internal function to setup the [Connection](#) object for the accepted socket.
- virtual void [listenWorker](#) ()
Background task to accept incoming connections.
- virtual void [deleteConnectionAsync](#) ([ConnectionBase](#) *connection)
Schedules a connection object to be deleted.
- void [onConnected](#) ([ConnectionBase](#) *connection) override
Event handler for connection established.
- void [onDisconnected](#) ([ConnectionBase](#) *connection, std::exception exception) override
Event handler for connection closed.
- void [onPayloadSent](#) ([ConnectionBase](#) *connection, const [Serializable](#) &payload, size_t size) override
Event handler for payload sent.
- void [onPayloadReceived](#) ([ConnectionBase](#) *connection, const [Serializable](#) &payload, size_t size) override
Event handler for payload received.

Protected Attributes

- std::set< [ServerObserver](#) * > [_observers](#)
- std::map< int, [ConnectionBase](#) * > [_clients](#)
- std::set< [ConnectionBase](#) * > [_graveyard](#)
- std::atomic< bool > [_running](#)
- std::thread [_listenThread](#)
- PSocket * [_listenSocket](#)

5.33.1 Member Function Documentation

5.33.1.1 clients()

```
virtual std::map<int, ConnectionBase> Server::clients ( ) const [inline], [virtual]
```

Gets all connected clients.

Returns

`std::map<int, ConnectionBase>` set of all clients

5.33.1.2 deleteConnectionAsync()

```
void Server::deleteConnectionAsync (
    ConnectionBase * connection ) [protected], [virtual]
```

Schedules a connection object to be deleted.

Parameters

<i>connection</i>	object to be deleted
-------------------	----------------------

5.33.1.3 isRunning()

```
virtual bool Server::isRunning ( ) const [inline], [virtual]
```

Returns

true Whether the server is currently accepting incoming connections

5.33.1.4 join()

```
void Server::join ( ) [virtual]
```

Joins the listener thread.

5.33.1.5 listenWorker()

```
void Server::listenWorker ( ) [protected], [virtual]
```

Background task to accept incoming connections.

5.33.1.6 onConnected()

```
void Server::onConnected (
    ConnectionBase * connection ) [override], [protected], [virtual]
```

Event handler for connection established.

Parameters

<i>connection</i>	event source
-------------------	--------------

Reimplemented from [ConnectionObserver](#).

5.33.1.7 onDisconnected()

```
void Server::onDisconnected (
    ConnectionBase * connection,
    std::exception exception ) [override], [protected], [virtual]
```

Event handler for connection closed.

Parameters

<i>connection</i>	event source
<i>exception</i>	connection exception

Reimplemented from [ConnectionObserver](#).

5.33.1.8 onPayloadReceived()

```
void Server::onPayloadReceived (
    ConnectionBase * connection,
    const Serializable & payload,
    size_t size ) [override], [protected], [virtual]
```

Event handler for payload received.

Parameters

<i>connection</i>	event source
<i>payload</i>	received payload
<i>size</i>	size (in bytes) of received payload

Reimplemented from [ConnectionObserver](#).

5.33.1.9 onPayloadSent()

```
void Server::onPayloadSent (
    ConnectionBase * connection,
    const Serializable & payload,
    size_t size ) [override], [protected], [virtual]
```

Event handler for payload sent.

Parameters

<i>connection</i>	event source
<i>payload</i>	sent payload
<i>size</i>	size (in bytes) of sent payload

Reimplemented from [ConnectionObserver](#).

5.33.1.10 prepareClient()

```
void Server::prepareClient (
    PSocket * socket ) [protected], [virtual]
```

Internal function to setup the [Connection](#) object for the accepted socket.

Parameters

<i>socket</i>	
---------------	--

5.33.1.11 shouldAcceptSocket()

```
bool Server::shouldAcceptSocket (
    PSocket * socket ) [protected], [virtual]
```

Predicate that should decide whether to accept the socket connection.

Parameters

<i>socket</i>	accepted socket
---------------	-----------------

Returns

whether to accept the socket

5.33.1.12 start()

```
void Server::start (
    const std::string & host,
    int port ) [virtual]
```

Starts listening for incoming connections on the specified endpoint.

Parameters

<i>host</i>	IP address
<i>port</i>	port number

5.33.1.13 stop()

```
void Server::stop ( ) [virtual]
```

Stops listening for incoming connections and drops all clients.

5.33.1.14 subscribe()

```
void Server::subscribe (
    ServerObserver * observer ) [virtual]
```

Subscribes to the events of this server object.

Parameters

<i>observer</i>	
-----------------	--

5.33.1.15 unsubscribe()

```
void Server::unsubscribe (
    ServerObserver * observer ) [virtual]
```

Unsubscribes from the events of this server object.

Parameters

<i>observer</i>	
-----------------	--

5.33.1.16 writeToAll()

```
void Server::writeToAll (
    const Serializable & msg ) [virtual]
```

Sends a packet to all connected clients.

Parameters

<i>msg</i>	payload
------------	---------

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

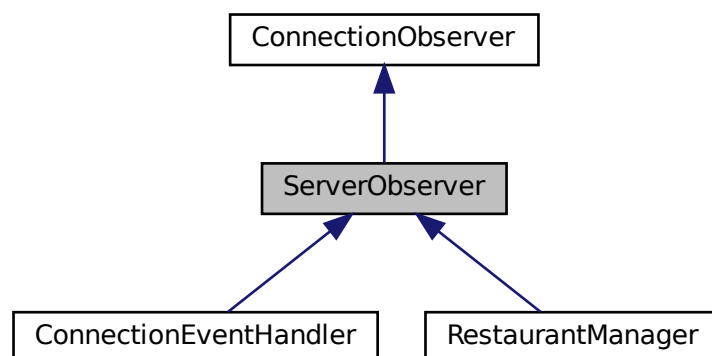
- BaseLibrary/Server.h
- BaseLibrary/Server.cpp

5.34 ServerObserver Class Reference

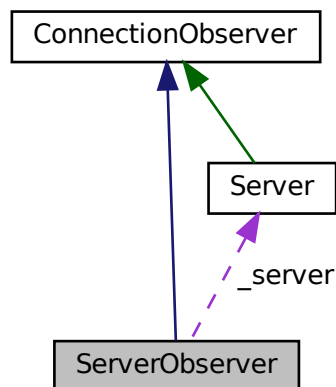
Observer that allows listening to events of a [Server](#) object.

```
#include <Server.h>
```

Inheritance diagram for ServerObserver:



Collaboration diagram for ServerObserver:



Public Member Functions

- `Server * getServer () const`
Gets the bound `Server` object.
- `void setServer (Server *server)`
Sets the bound `Server` object.

Protected Attributes

- `Server * _server`

5.34.1 Detailed Description

Observer that allows listening to events of a `Server` object.

5.34.2 Member Function Documentation

5.34.2.1 `getServer()`

```
Server* ServerObserver::getServer ( ) const [inline]
```

Gets the bound `Server` object.

Returns

`Server*`

5.34.2.2 `setServer()`

```
void ServerObserver::setServer (
    Server * server ) [inline]
```

Sets the bound [Server](#) object.

Parameters

<i>server</i>	
---------------	--

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

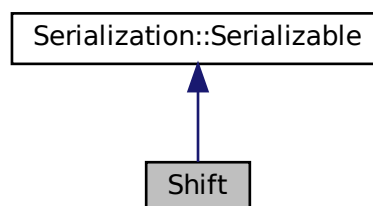
- BaseLibrary/Server.h

5.35 Shift Class Reference

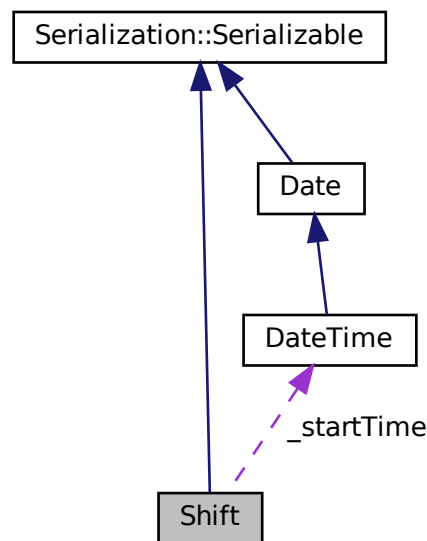
Class that represents a [Shift](#) in the schedule.

```
#include <Shift.h>
```

Inheritance diagram for Shift:



Collaboration diagram for Shift:



Public Member Functions

- `Type` `getType ()` const override
- `Shift` (`DateTime` startTime, int workHours, std::wstring jobName, identity_t workerId=0, identity_t id=0)
Construct a new *Shift* object.
- `DateTime` & `getStartTime ()`
Gets the *Shift*'s start time by reference.
- const `DateTime` & `getStartTime ()` const
Gets the *Shift*'s start time by const reference.
- `DateTime` `getEndTime ()` const
Gets the *Shift*'s end time.
- void `setStartTime` (`DateTime` startTime)
Sets the start time.
- uint8_t `getWorkHours ()` const
Gets the duration (in hours)
- void `setWorkHours` (const uint8_t workHours)
Sets the duration (in hours)
- identity_t `getWorkerId ()` const
Gets the associated worker id.
- void `setWorkerId` (const identity_t workerId)
Sets the associated worker id.
- identity_t `getId ()` const
Gets the id.
- void `setId` (const identity_t id)
Sets the id.
- virtual std::wstring `getJobName ()` const

Gets the name of this job.

- virtual void [setJobName](#) (std::wstring jobName)

Sets the name of this job.

- std::ostream & [serialize](#) (std::ostream &destination) const override

Serializes this object into a binary output stream.

- std::istream & [deserialize](#) (std::istream &source) override

Deserializes this object from a binary input stream.

Protected Attributes

- identity_t _id {}
- [DateTime](#) _startTime
- uint8_t _workHours {}
- identity_t _workerId {}
- std::wstring _jobName

Friends

- bool **operator==** (const [Shift](#) &lhs, const [Shift](#) &rhs)
- bool **operator!=** (const [Shift](#) &lhs, const [Shift](#) &rhs)
- bool **operator<** (const [Shift](#) &lhs, const [Shift](#) &rhs)
- bool **operator<=** (const [Shift](#) &lhs, const [Shift](#) &rhs)
- bool **operator>** (const [Shift](#) &lhs, const [Shift](#) &rhs)
- bool **operator>=** (const [Shift](#) &lhs, const [Shift](#) &rhs)

5.35.1 Detailed Description

Class that represents a [Shift](#) in the schedule.

5.35.2 Constructor & Destructor Documentation

5.35.2.1 Shift()

```
Shift::Shift (
    DateTime startTime,
    int workHours,
    std::wstring jobName,
    identity_t workerId = 0,
    identity_t id = 0 ) [inline]
```

Construct a new [Shift](#) object.

Parameters

<i>startTime</i>	starting time
<i>workHours</i>	duration
<i>jobName</i>	name of the job
<i>workerId</i>	id of the worker
<i>id</i>	id of the shift

5.35.3 Member Function Documentation

5.35.3.1 deserialize()

```
std::istream& Shift::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.35.3.2 getEndTime()

```
DateTime Shift::getEndTime ( ) const [inline]
```

Gets the [Shift](#)'s end time.

Returns

[DateTime](#)

5.35.3.3 getId()

```
identity_t Shift::getId ( ) const [inline]
```

Gets the id.

Returns

identity_t

5.35.3.4 `getJobName()`

```
virtual std::wstring Shift::getJobName ( ) const [inline], [virtual]
```

Gets the name of this job.

Returns

`std::wstring`

5.35.3.5 `getStartTime()` [1/2]

```
DateTime& Shift::getStartTime ( ) [inline]
```

Gets the [Shift](#)'s start time by reference.

Returns

[DateTime&](#)

5.35.3.6 `getStartTime()` [2/2]

```
const DateTime& Shift::getStartTime ( ) const [inline]
```

Gets the [Shift](#)'s start time by const reference.

Returns

`const DateTime&`

5.35.3.7 `getType()`

```
Type Shift::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.35.3.8 getWorkerId()

```
identity_t Shift::getWorkerId ( ) const [inline]
```

Gets the associated worker id.

Returns

identity_t

5.35.3.9 getWorkHours()

```
uint8_t Shift::getWorkHours ( ) const [inline]
```

Gets the duration (in hours)

Returns

uint8_t hours

5.35.3.10 serialize()

```
std::ostream& Shift::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.35.3.11 setId()

```
void Shift::setId (
    const identity_t id ) [inline]
```

Sets the id.

Parameters

<i>id</i>	
-----------	--

5.35.3.12 setJobName()

```
virtual void Shift::setJobName (  
    std::wstring jobName ) [inline], [virtual]
```

Sets the name of this job.

Parameters

<i>jobName</i>	
----------------	--

5.35.3.13 setStartTime()

```
void Shift::setStartTime (  
    DateTime startTime ) [inline]
```

Sets the start time.

Parameters

<i>startTime</i>	
------------------	--

5.35.3.14 setWorkerId()

```
void Shift::setWorkerId (  
    const identity_t workerId ) [inline]
```

Sets the associated worker id.

Parameters

<i>workerId</i>	
-----------------	--

5.35.3.15 setWorkHours()

```
void Shift::setWorkHours (
    const uint8_t workHours ) [inline]
```

Sets the duration (in hours)

Parameters

<i>workHours</i>	
------------------	--

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

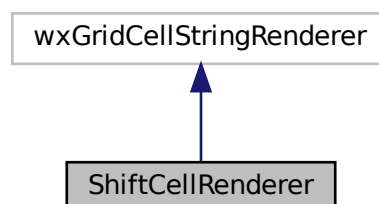
- BaseLibrary/Shift.h

5.36 ShiftCellRenderer Class Reference

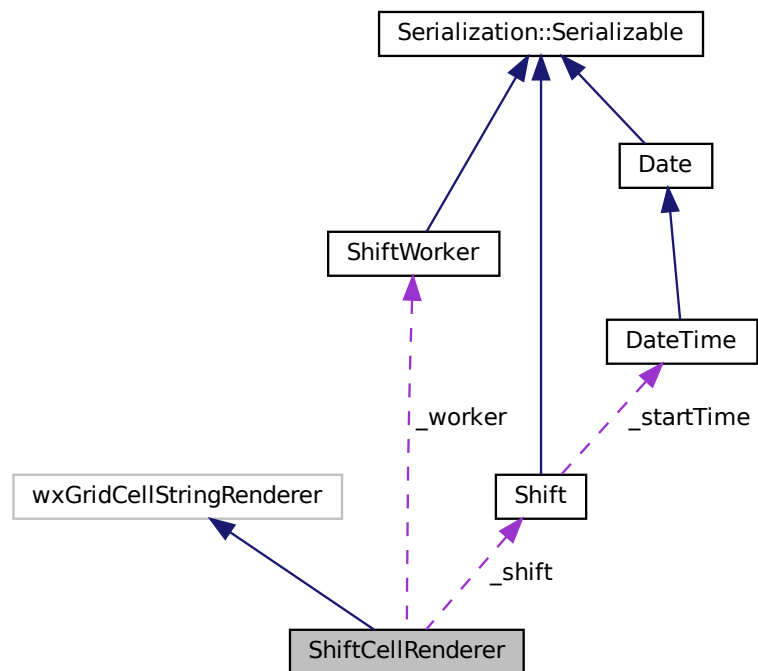
[ShiftCellRenderer](#) class represents set of settings for beautiful UI layout Inherits from wxGridCellStringRenderer.

```
#include <ShiftCellRenderer.h>
```

Inheritance diagram for ShiftCellRenderer:



Collaboration diagram for ShiftCellRenderer:



Public Member Functions

- [ShiftCellRenderer](#) ()
ShiftCellRenderer constructor.
- void [Draw](#) (wxGrid &grid, wxGridCellAttr &attr, wxDC &dc, const wxRect &rect, int row, int col, bool isSelected) override
Method that does some magic on the grid cell to make it more appealing.
- wxSize [GetBestSize](#) (wxGrid &grid, wxGridCellAttr &attr, wxDC &dc, int row, int col) override
Method that tries to get the best size for the grid to fit the currecn window size.
- virtual [Shift](#) & [getShift](#) ()
- virtual void [setShift](#) (Shift shift)
- virtual [ShiftWorker](#) & [getWorker](#) ()
- virtual void [setWorker](#) (ShiftWorker worker)

Protected Attributes

- [Shift _shift](#)
current shift obj
- [ShiftWorker _worker](#)
current shiftWorker obj

5.36.1 Detailed Description

[ShiftCellRenderer](#) class represents set of settings for beautiful UI layout Inherits from wxGridCellStringRenderer.

5.36.2 Member Function Documentation

5.36.2.1 Draw()

```
void ShiftCellRenderer::Draw (
    wxGrid & grid,
    wxGridCellAttr & attr,
    wxDC & dc,
    const wxRect & rect,
    int row,
    int col,
    bool isSelected ) [inline], [override]
```

Method that does some magic on the grid cell to make it more appealing.

Parameters

<i>grid</i>	wxGrid obj
<i>attr</i>	wxGridCellAttr obj
<i>dc</i>	wxDC obj
<i>rect</i>	wxRect obj
<i>row</i>	row index
<i>col</i>	col index
<i>isSelected</i>	bool if sth is selected xD

5.36.2.2 GetBestSize()

```
wxSize ShiftCellRenderer::GetBestSize (
    wxGrid & grid,
    wxGridCellAttr & attr,
    wxDC & dc,
    int row,
    int col ) [inline], [override]
```

Method that tries to get the best size for the grid to fit the currecn window size.

Parameters

<i>grid</i>	wxGrid obj
<i>attr</i>	wxGridCellAttr obj
<i>dc</i>	wxDC obj
<i>row</i>	row index
<i>col</i>	col index

5.36.2.3 getShift()

```
virtual Shift& ShiftCellRenderer::getShift ( ) [inline], [virtual]
```

Returns

[Shift](#) obj

5.36.2.4 getWorker()

```
virtual ShiftWorker& ShiftCellRenderer::getWorker ( ) [inline], [virtual]
```

Returns

[ShiftWorker](#) obj

5.36.2.5 setShift()

```
virtual void ShiftCellRenderer::setShift (
    Shift shift ) [inline], [virtual]
```

Parameters

<i>shift</i>	- Shift obj
--------------	-----------------------------

5.36.2.6 setWorker()

```
virtual void ShiftCellRenderer::setWorker (
    ShiftWorker worker ) [inline], [virtual]
```

Parameters

<i>worker</i>	- ShiftWorker obj
---------------	-----------------------------------

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

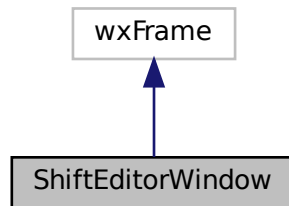
- UIPizzaClient/ShiftCellRenderer.h

5.37 ShiftEditorWindow Class Reference

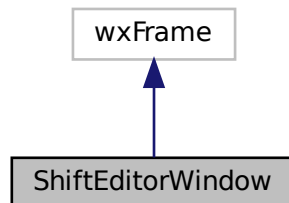
[ShiftEditorWindow](#) class represents a [Shift](#) Editor window Inherits from wxFrame class.

```
#include <ShiftEditorWindow.h>
```

Inheritance diagram for ShiftEditorWindow:



Collaboration diagram for ShiftEditorWindow:



Public Member Functions

- [ShiftEditorWindow](#) ([App](#) *app, identity_t shiftId=0)
[ShiftEditorWindow](#) constructor.
- [~ShiftEditorWindow](#) ()
[ShiftEditorWindow](#) destructor.
- void [refresh](#) ()
Methods that refreshes every window in the current context.
- [ShiftWorker](#) [getWorkerData](#) (std::wstring input)
Methods that converts worker wstring into [ShiftWorker](#) object.
- void [onShiftInserted](#) (const [S2C_InsertShiftReply](#) &reply)
- void [onShiftDeleted](#) (const [S2C_DeleteShiftReply](#) &reply)
- [wxDECLARE_EVENT_TABLE](#) ()

5.37.1 Detailed Description

[ShiftEditorWindow](#) class represents a [Shift](#) Editor window Inherits from wxFrame class.

5.37.2 Constructor & Destructor Documentation

5.37.2.1 ShiftEditorWindow()

```
ShiftEditorWindow::ShiftEditorWindow (
    App * app,
    identity_t shiftId = 0 )
```

[ShiftEditorWindow](#) constructor.

Parameters

<i>app</i>	- App object pointer
<i>shiftId</i>	- shift ID (default 0)

5.37.3 Member Function Documentation

5.37.3.1 getWorkerData()

```
ShiftWorker ShiftEditorWindow::getWorkerData (
    std::wstring input )
```

Methods that converts worker wstring into [ShiftWorker](#) object.

Parameters

<i>input</i>	- wstring with worker data
--------------	----------------------------

Returns

[ShiftWorker](#) object containing worker's data

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

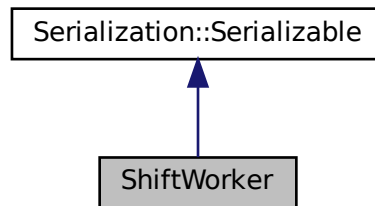
- UIPizzaClient/ShiftEditorWindow.h
- UIPizzaClient/ShiftEditorWindow.cpp

5.38 ShiftWorker Class Reference

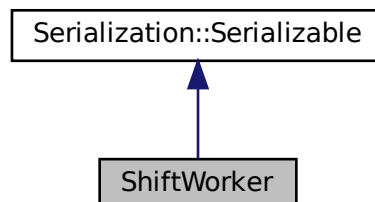
Class representing a worker associated with a [Shift](#).

```
#include <ShiftWorker.h>
```

Inheritance diagram for ShiftWorker:



Collaboration diagram for ShiftWorker:



Public Member Functions

- virtual identity_t [getId](#) () const
Gets the id of this worker.
- virtual void [setId](#) (const identity_t workerId)
Sets the id of this worker.
- virtual std::wstring [getFirstName](#) () const
Gets the first name.
- virtual void [setFirstName](#) (const std::wstring &firstName)
Sets the first name.
- virtual std::wstring [getLastName](#) () const
Gets the last name.
- virtual void [setLastName](#) (const std::wstring &lastName)

- *Sets the last name.*
- virtual std::wstring [getTitle](#) () const
- *Gets the title of this worker.*
- virtual void [setTitle](#) (const std::wstring &title)
- *Sets the title of this worker.*
- [Type](#) [getType](#) () const override
- virtual std::wstring [toString](#) () const
- *Returns a string representation of this object.*
- **ShiftWorker** (std::wstring firstName, std::wstring lastName, std::wstring title, identity_t id=0)
- std::ostream & [serialize](#) (std::ostream &destination) const override
- *Serializes this object into a binary output stream.*
- std::istream & [deserialize](#) (std::istream &source) override
- *Deserializes this object from a binary input stream.*
- **ShiftWorker** (const [ShiftWorker](#) &other)
- **ShiftWorker** ([ShiftWorker](#) &&other) noexcept
- [ShiftWorker](#) & **operator=** (const [ShiftWorker](#) &other)
- [ShiftWorker](#) & **operator=** ([ShiftWorker](#) &&other) noexcept

Protected Attributes

- identity_t **_id**
- std::wstring **_firstName**
- std::wstring **_lastName**
- std::wstring **_title**

Friends

- bool **operator==** (const [ShiftWorker](#) &lhs, const [ShiftWorker](#) &rhs)
- bool **operator!=** (const [ShiftWorker](#) &lhs, const [ShiftWorker](#) &rhs)
- bool **operator<** (const [ShiftWorker](#) &lhs, const [ShiftWorker](#) &rhs)
- bool **operator<=** (const [ShiftWorker](#) &lhs, const [ShiftWorker](#) &rhs)
- bool **operator>** (const [ShiftWorker](#) &lhs, const [ShiftWorker](#) &rhs)
- bool **operator>=** (const [ShiftWorker](#) &lhs, const [ShiftWorker](#) &rhs)

5.38.1 Detailed Description

Class representing a worker associated with a [Shift](#).

5.38.2 Member Function Documentation

5.38.2.1 [deserialize\(\)](#)

```
std::istream& ShiftWorker::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

5.38.2.2 getFirstName()

```
virtual std::wstring ShiftWorker::getFirstName ( ) const [inline], [virtual]
```

Gets the first name.

Returns

std::wstring

5.38.2.3 getId()

```
virtual identity_t ShiftWorker::getId ( ) const [inline], [virtual]
```

Gets the id of this worker.

Returns

identity_t

5.38.2.4 getLastName()

```
virtual std::wstring ShiftWorker::getLastName ( ) const [inline], [virtual]
```

Gets the last name.

Returns

std::wstring

5.38.2.5 getTitle()

```
virtual std::wstring ShiftWorker::getTitle ( ) const [inline], [virtual]
```

Gets the title of this worker.

Returns

std::wstring

5.38.2.6 getType()

```
Type ShiftWorker::getType ( ) const [inline], [override], [virtual]
```

Returns

Type of this serializable object

Implements [Serialization::Serializable](#).

5.38.2.7 serialize()

```
std::ostream& ShiftWorker::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

5.38.2.8 setFirstName()

```
virtual void ShiftWorker::setFirstName (
    const std::wstring & firstName ) [inline], [virtual]
```

Sets the first name.

Parameters

<i>firstName</i>	
------------------	--

5.38.2.9 setId()

```
virtual void ShiftWorker::setId (
    const identity_t workerId ) [inline], [virtual]
```

Sets the id of this worker.

Parameters

<i>workerId</i>	
-----------------	--

5.38.2.10 setLastName()

```
virtual void ShiftWorker::setLastName (
    const std::wstring & lastName ) [inline], [virtual]
```

Sets the last name.

Parameters

<i>lastName</i>	
-----------------	--

5.38.2.11 setTitle()

```
virtual void ShiftWorker::setTitle (
    const std::wstring & title ) [inline], [virtual]
```

Sets the title of this worker.

Parameters

<i>title</i>	
--------------	--

5.38.2.12 toString()

```
virtual std::wstring ShiftWorker::toString ( ) const [inline], [virtual]
```

Returns a string representation of this object.

Returns

std::wstring

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

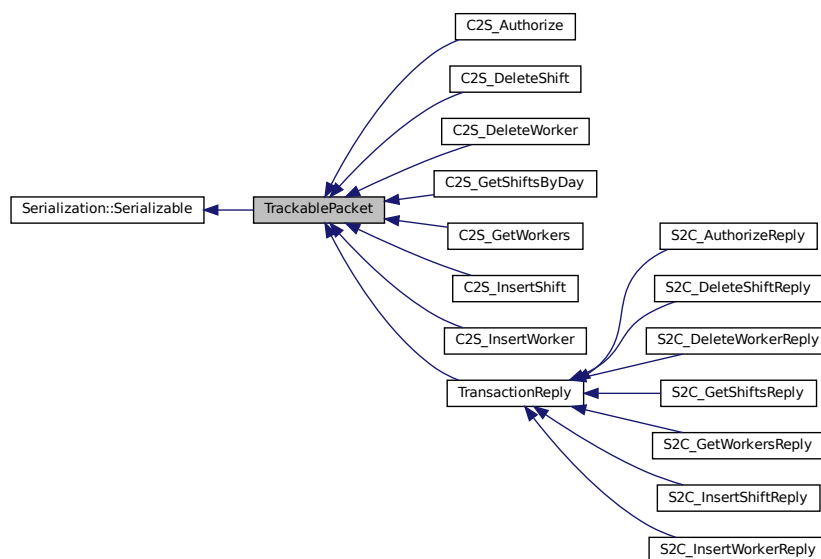
- BaseLibrary/ShiftWorker.h

5.39 TrackablePacket Class Reference

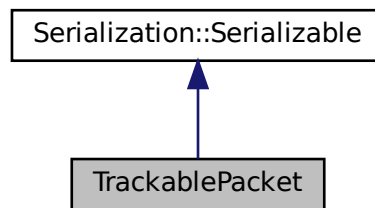
Base-class for tracking request-response packets by id.

```
#include <TrackablePacket.h>
```

Inheritance diagram for TrackablePacket:



Collaboration diagram for TrackablePacket:



Public Member Functions

- `TrackablePacket` (int requestId)
Construct a new Trackable Packet.
- void `setRequestId` (int requestId)
Sets the Request Id.
- int `getRequestId` () const
Gets the object's Request Id.
- std::ostream & `serialize` (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & `deserialize` (std::istream &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- int `_requestId`

Friends

- bool `operator==` (const `TrackablePacket` &lhs, const `TrackablePacket` &rhs)
- bool `operator!=` (const `TrackablePacket` &lhs, const `TrackablePacket` &rhs)

5.39.1 Detailed Description

Base-class for tracking request-response packets by id.

5.39.2 Constructor & Destructor Documentation

5.39.2.1 TrackablePacket()

```
TrackablePacket::TrackablePacket (
    int requestId ) [inline]
```

Construct a new Trackable Packet.

Parameters

<i>request↔ Id</i>	request id
------------------------	------------

5.39.3 Member Function Documentation

5.39.3.1 deserialize()

```
std::istream& TrackablePacket::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [Serialization::Serializable](#).

Reimplemented in [TransactionReply](#).

5.39.3.2 getRequestId()

```
int TrackablePacket::getRequestId ( ) const [inline]
```

Gets the object's Request Id.

Returns

int Request Id

5.39.3.3 serialize()

```
std::ostream& TrackablePacket::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [Serialization::Serializable](#).

Reimplemented in [TransactionReply](#).

5.39.3.4 setRequestId()

```
void TrackablePacket::setRequestId (
    int requestId ) [inline]
```

Sets the Request Id.

Parameters

<i>requestId</i>	
------------------	--

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

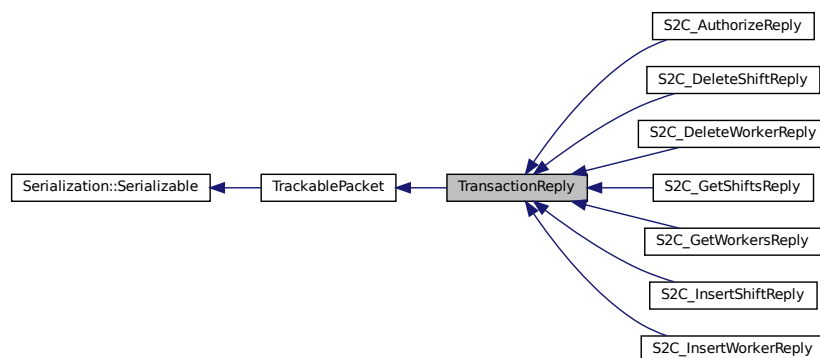
- BaseLibrary/TrackablePacket.h

5.40 TransactionReply Class Reference

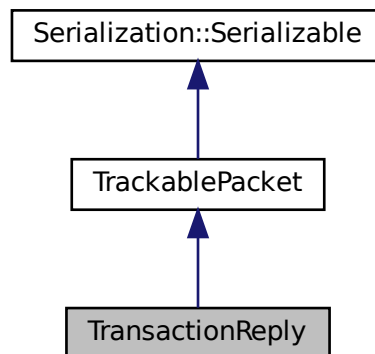
Base-class for trackable transaction replies with success flags and error message.

```
#include <TransactionReply.h>
```

Inheritance diagram for TransactionReply:



Collaboration diagram for TransactionReply:



Public Member Functions

- [TransactionReply](#) (int requestId)
Construct a new successful [TransactionReply](#) object.
- [TransactionReply](#) (int requestId, std::string msg)
Construct a new failure [TransactionReply](#) object and sets the error message.
- [TransactionReply](#) (int requestId, bool success, std::string msg="")
Construct a new [Transaction Reply](#) object.
- virtual bool [isSuccess](#) () const
- virtual void [setSuccess](#) (const bool success)
Sets the success flag.
- virtual std::string [getErrorMsg](#) () const
Gets the error message.
- virtual void [setErrorMsg](#) (const std::string &msg)
Sets the error message and modifies the success flag to false.
- std::ostream & [serialize](#) (std::ostream &destination) const override
Serializes this object into a binary output stream.
- std::istream & [deserialize](#) (std::istream &source) override
Deserializes this object from a binary input stream.

Protected Attributes

- bool **_success**
- std::string **_errorMsg**

Friends

- bool **operator==** (const [TransactionReply](#) &lhs, const [TransactionReply](#) &rhs)
- bool **operator!=** (const [TransactionReply](#) &lhs, const [TransactionReply](#) &rhs)

5.40.1 Detailed Description

Base-class for trackable transaction replies with success flags and error message.

5.40.2 Constructor & Destructor Documentation

5.40.2.1 TransactionReply() [1/3]

```
TransactionReply::TransactionReply (
    int requestId ) [inline]
```

Construct a new successful [TransactionReply](#) object.

Parameters

<i>requestId</i>	request id
------------------	------------

5.40.2.2 TransactionReply() [2/3]

```
TransactionReply::TransactionReply (
    int requestId,
    std::string msg ) [inline]
```

Construct a new failure [TransactionReply](#) object and sets the error message.

Parameters

<i>requestId</i>	request id
<i>msg</i>	error message

5.40.2.3 TransactionReply() [3/3]

```
TransactionReply::TransactionReply (
    int requestId,
    bool success,
    std::string msg = "" ) [inline]
```

Construct a new Transaction Reply object.

Parameters

<i>requestId</i>	request id
<i>success</i>	whether the transaction succeeded
<i>msg</i>	error message

5.40.3 Member Function Documentation

5.40.3.1 deserialize()

```
std::istream& TransactionReply::deserialize (
    std::istream & source ) [inline], [override], [virtual]
```

Deserializes this object from a binary input stream.

Parameters

<i>source</i>	binary input stream
---------------	---------------------

Returns

input stream

Reimplemented from [TrackablePacket](#).

5.40.3.2 getErrorMsg()

```
virtual std::string TransactionReply::getErrorMsg ( ) const [inline], [virtual]
```

Gets the error message.

Returns

std::string error message

5.40.3.3 isSuccess()

```
virtual bool TransactionReply::isSuccess ( ) const [inline], [virtual]
```

Returns

Whether the transaction succeeded

5.40.3.4 serialize()

```
std::ostream& TransactionReply::serialize (
    std::ostream & destination ) const [inline], [override], [virtual]
```

Serializes this object into a binary output stream.

Parameters

<i>destination</i>	binary output stream
--------------------	----------------------

Returns

output stream

Reimplemented from [TrackablePacket](#).

5.40.3.5 setErrorMsg()

```
virtual void TransactionReply::setErrorMsg (
    const std::string & msg ) [inline], [virtual]
```

Sets the error message and modifies the success flag to false.

Parameters

<i>msg</i>	
------------	--

5.40.3.6 setSuccess()

```
virtual void TransactionReply::setSuccess (
    const bool success ) [inline], [virtual]
```

Sets the success flag.

Parameters

<i>success</i>	Whether the transaction succeeded
----------------	-----------------------------------

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

- BaseLibrary/TransactionReply.h

5.41 Serialization::TypeInfo Struct Reference

Struct to hold the object graph of serializable types.

```
#include <serialization.h>
```

Public Attributes

- bool **registered** = false
- std::function< std::shared_ptr< [Serializable](#) >> **constructor** = nullptr
Function that constructs a serializable type and returns its shared pointer.
- [Type parent](#) = Type::Unknown
Parent-type of this type.
- const char * **name** = nullptr
Compiler-generated name for this type.

Static Public Attributes

- static std::map< std::type_index, [Type](#) > **NATIVE_TYPES**
- static std::map< [Type](#), [TypeInfo](#) > **TYPES**

5.41.1 Detailed Description

Struct to hold the object graph of serializable types.

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

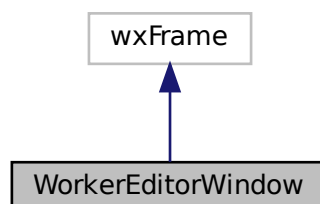
- BaseLibrary/serialization.h
- BaseLibrary/serialization.cpp

5.42 WorkerEditorWindow Class Reference

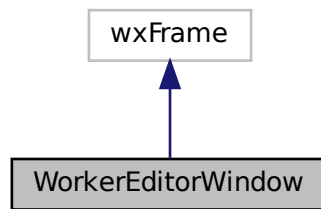
[WorkerEditorWindow](#) class represents the Worker Editor window Inherits from the wxFrame class.

```
#include <WorkerEditorWindow.h>
```

Inheritance diagram for WorkerEditorWindow:



Collaboration diagram for WorkerEditorWindow:



Public Member Functions

- [WorkerEditorWindow](#) ([App](#) *app, identity_t workerId=0)
WorkerEditorWindow constructor.
- [~WorkerEditorWindow](#) ()
WorkerEditorWindow destructor.
- void **onSubmitButtonClicked** (wxCommandEvent &evt)
- void **onDeleteButtonClicked** (wxCommandEvent &evt)
- void **onWorkerInserted** (const [S2C_InsertWorkerReply](#) &reply)
- void **onWorkerDeleted** (const [S2C_DeleteWorkerReply](#) &reply)
- **wxDECLARE_EVENT_TABLE** ()

5.42.1 Detailed Description

[WorkerEditorWindow](#) class represents the Worker Editor window Inherits from the `wxFrame` class.

5.42.2 Constructor & Destructor Documentation

5.42.2.1 WorkerEditorWindow()

```

WorkerEditorWindow::WorkerEditorWindow (
    App * app,
    identity_t workerId = 0 )
  
```

[WorkerEditorWindow](#) constructor.

Parameters

<i>app</i>	- App object pointer
<i>workerId</i>	- ID of the chosen worker

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

- UIPizzaClient/WorkerEditorWindow.h
- UIPizzaClient/WorkerEditorWindow.cpp

Index

addHandler
 ConnectionObserver, [57](#), [58](#)

App, [9](#)
 getClient, [10](#)
 getMainWindow, [10](#)
 getShiftWindow, [11](#)
 getWorkerWindow, [11](#)
 setMainWindow, [11](#)
 setShiftWindow, [11](#)
 setWorkerWindow, [12](#)

assertConnected
 ConnectionBase, [46](#)

authorize
 RestaurantClient, [88](#)

C2S_Authorize, [12](#)
 C2S_Authorize, [14](#)
 deserialize, [14](#)
 getToken, [14](#)
 getType, [14](#)
 serialize, [15](#)
 setToken, [15](#)

C2S_DeleteShift, [16](#)
 C2S_DeleteShift, [17](#)
 deserialize, [17](#)
 getShiftId, [18](#)
 getType, [18](#)
 serialize, [18](#)
 setShiftId, [19](#)

C2S_DeleteWorker, [19](#)
 C2S_DeleteWorker, [21](#)
 deserialize, [21](#)
 getType, [22](#)
 getWorkerId, [22](#)
 serialize, [22](#)
 setWorkerId, [22](#)

C2S_GetShiftsByDay, [23](#)
 C2S_GetShiftsByDay, [25](#)
 deserialize, [25](#)
 getDate, [25](#)
 getType, [26](#)
 serialize, [26](#)
 setDate, [26](#)

C2S_GetWorkers, [27](#)
 getType, [28](#)

C2S_InsertShift, [29](#)
 C2S_InsertShift, [31](#)
 deserialize, [31](#)
 getShift, [32](#)
 getType, [32](#)

isModifyExisting, [32](#)
 serialize, [32](#)
 setModifyExisting, [33](#)
 setShift, [33](#)

C2S_InsertWorker, [33](#)
 C2S_InsertWorker, [35](#)
 deserialize, [35](#)
 getType, [36](#)
 getWorker, [36](#)
 isModifyExisting, [36](#)
 serialize, [37](#)
 setModifyExisting, [37](#)
 setWorker, [37](#)

cleanup
 Connection, [40](#)
 ConnectionBase, [46](#)

clear
 DataBag, [61](#)

clients
 Server, [136](#)

close
 ConnectionBase, [46](#)

closeError
 Connection, [40](#)
 ConnectionBase, [47](#)

connect
 Connection, [40](#), [41](#)
 ConnectionBase, [47](#)

Connection, [38](#)
 cleanup, [40](#)
 closeError, [40](#)
 connect, [40](#), [41](#)
 Connection, [39](#)
 isAlive, [41](#)
 readAsync, [41](#)
 readSync, [41](#)
 readSyncInternal, [42](#)
 receiveFromSocket, [42](#)
 sendBuffer, [42](#)
 setReadingAsync, [43](#)
 writeSync, [43](#)

ConnectionBase, [43](#)
 assertConnected, [46](#)
 cleanup, [46](#)
 close, [46](#)
 closeError, [47](#)
 connect, [47](#)
 ConnectionBase, [46](#)
 getData, [48](#)

- getId, [48](#)
- isAlive, [48](#)
- isReadingAsync, [49](#)
- newRequestId, [49](#)
- onConnected, [49](#)
- onDisconnected, [49](#)
- onPayloadReceived, [50](#)
- onPayloadSent, [50](#)
- readAsync, [50](#)
- readSync, [50](#)
- setReadingAsync, [51](#)
- subscribe, [51](#)
- unsubscribe, [51](#)
- writeReplySync, [52](#)
- writeRequestSync, [52](#)
- writeSync, [53](#)
- ConnectionEventHandler, [53](#)
 - onConnected, [54](#)
 - onDisconnected, [56](#)
 - onPayloadSent, [56](#)
- ConnectionObserver, [56](#)
 - addHandler, [57](#), [58](#)
 - onConnected, [58](#)
 - onDisconnected, [59](#)
 - onPayloadReceived, [59](#)
 - onPayloadSent, [59](#)
 - removeHandlers, [60](#)
- DataBag, [60](#)
 - clear, [61](#)
 - get, [61](#)
 - put, [62](#)
 - remove, [62](#)
- Date, [63](#)
 - Date, [64](#), [65](#)
 - deserialize, [65](#)
 - getDay, [65](#)
 - getMonth, [66](#)
 - getType, [66](#)
 - getYear, [66](#)
 - serialize, [66](#)
 - setDay, [67](#)
 - setMonth, [67](#)
 - setYear, [67](#)
- DateTime, [68](#)
 - DateTime, [70](#)
 - deserialize, [71](#)
 - getDate, [71](#)
 - getHour, [72](#)
 - getMinute, [72](#)
 - getSecond, [72](#)
 - getType, [72](#)
 - serialize, [72](#)
 - setHour, [73](#)
 - setMinute, [73](#)
 - setSecond, [73](#)
- deleteConnectionAsync
 - Server, [137](#)
- deleteShift
 - RestaurantManager, [93](#)
- deleteWorker
 - RestaurantManager, [94](#)
- deserialize
 - C2S_Authorize, [14](#)
 - C2S_DeleteShift, [17](#)
 - C2S_DeleteWorker, [21](#)
 - C2S_GetShiftsByDay, [25](#)
 - C2S_InsertShift, [31](#)
 - C2S_InsertWorker, [35](#)
 - Date, [65](#)
 - DateTime, [71](#)
 - RestaurantManager, [94](#)
 - S2C_AuthorizeReply, [101](#)
 - S2C_ClientSync, [104](#)
 - S2C_DeleteShiftReply, [110](#)
 - S2C_DeleteWorkerReply, [114](#)
 - S2C_GetShiftsReply, [118](#)
 - S2C_GetWorkersReply, [122](#)
 - S2C_InsertShiftReply, [126](#)
 - S2C_InsertWorkerReply, [131](#)
 - Serialization::Serializable, [134](#)
 - Shift, [146](#)
 - ShiftWorker, [157](#)
 - TrackablePacket, [163](#)
 - TransactionReply, [167](#)
- Draw
 - ShiftCellRenderer, [152](#)
- get
 - DataBag, [61](#)
- getAccessTokens
 - RestaurantManager, [94](#)
- GetBestSize
 - ShiftCellRenderer, [152](#)
- getChangedShifts
 - S2C_ClientSync, [104](#), [105](#)
- getChangedWorkers
 - S2C_ClientSync, [105](#)
- getClient
 - App, [10](#)
- getCurrentDate
 - MainWindow, [75](#)
- getData
 - ConnectionBase, [48](#)
- getDate
 - C2S_GetShiftsByDay, [25](#)
 - DateTime, [71](#)
 - S2C_GetShiftsReply, [118](#)
- getDay
 - Date, [65](#)
- getEndTime
 - Shift, [146](#)
- getErrorMsg
 - TransactionReply, [167](#)
- getFirstName
 - ShiftWorker, [158](#)
- getHour
 - DateTime, [72](#)

- getId
 - ConnectionBase, 48
 - S2C_DeleteShiftReply, 110
 - S2C_DeleteWorkerReply, 114
 - Shift, 146
 - ShiftWorker, 158
- getInstance
 - PingService, 84
- getJobName
 - Shift, 146
- getLastName
 - ShiftWorker, 158
- getMainWindow
 - App, 10
- getMinute
 - DateTime, 72
- getMonth
 - Date, 66
- getPermissions
 - S2C_AuthorizeReply, 101
- getRemovedShifts
 - S2C_ClientSync, 105, 106
- getRemovedWorkers
 - S2C_ClientSync, 106
- getRequestId
 - TrackablePacket, 163
- getSecond
 - DateTime, 72
- getServer
 - ServerObserver, 142
- getShift
 - C2S_InsertShift, 32
 - S2C_InsertShiftReply, 127
 - ShiftCellRenderer, 153
- getShiftId
 - C2S_DeleteShift, 18
- getShiftWindow
 - App, 11
- getShifts
 - RestaurantManager, 95
 - S2C_GetShiftsReply, 118, 119
- getShiftsByDay
 - RestaurantManager, 95
- getStartTime
 - Shift, 147
- getTitle
 - ShiftWorker, 158
- getToken
 - C2S_Authorize, 14
- getType
 - C2S_Authorize, 14
 - C2S_DeleteShift, 18
 - C2S_DeleteWorker, 22
 - C2S_GetShiftsByDay, 26
 - C2S_GetWorkers, 28
 - C2S_InsertShift, 32
 - C2S_InsertWorker, 36
 - Date, 66
 - DateTime, 72
 - Ping, 81
 - PingReply, 82
 - RestaurantManager, 95
 - S2C_AuthorizeReply, 101
 - S2C_ClientSync, 106
 - S2C_DeleteShiftReply, 110
 - S2C_DeleteWorkerReply, 114
 - S2C_GetShiftsReply, 119
 - S2C_GetWorkersReply, 122
 - S2C_InsertShiftReply, 127
 - S2C_InsertWorkerReply, 131
 - Serialization::Serializable, 134
 - Shift, 147
 - ShiftWorker, 159
- getWorkHours
 - Shift, 148
- getWorker
 - C2S_InsertWorker, 36
 - S2C_InsertWorkerReply, 131
 - ShiftCellRenderer, 153
- getWorkerData
 - ShiftEditorWindow, 155
- getWorkerId
 - C2S_DeleteWorker, 22
 - Shift, 147
- getWorkerWindow
 - App, 11
- getWorkers
 - RestaurantManager, 95
 - S2C_GetWorkersReply, 123
- getYear
 - Date, 66
- insertShift
 - RestaurantManager, 96
- insertWorker
 - RestaurantManager, 96
- isAlive
 - Connection, 41
 - ConnectionBase, 48
- isModifyExisting
 - C2S_InsertShift, 32
 - C2S_InsertWorker, 36
- isReadingAsync
 - ConnectionBase, 49
- isRunning
 - Server, 137
- isSuccess
 - TransactionReply, 167
- join
 - Server, 137
- listenWorker
 - Server, 137
- MainWindow, 74
 - getCurrentDate, 75

- MainWindow, 75
 - resizeGrid, 76
- memory_buf, 76
 - memory_buf, 78
 - setData, 78
 - setLength, 78
- memory_stream, 79
- newIdentityId
 - RestaurantManager, 96
- newRequestId
 - ConnectionBase, 49
- onConnected
 - ConnectionBase, 49
 - ConnectionEventHandler, 54
 - ConnectionObserver, 58
 - Server, 137
- onDisconnected
 - ConnectionBase, 49
 - ConnectionEventHandler, 56
 - ConnectionObserver, 59
 - Server, 138
- onPayloadReceived
 - ConnectionBase, 50
 - ConnectionObserver, 59
 - RestaurantManager, 97
 - Server, 138
- onPayloadSent
 - ConnectionBase, 50
 - ConnectionEventHandler, 56
 - ConnectionObserver, 59
 - Server, 138
- Ping, 80
 - getType, 81
- PingReply, 81
 - getType, 82
- PingService, 83
 - getInstance, 84
 - PingService, 84
 - subscribe, 84
 - unsubscribe, 85
 - worker, 85
- prepareClient
 - Server, 139
- put
 - DataBag, 62
- queryDeleteShift
 - RestaurantClient, 88
- queryDeleteWorker
 - RestaurantClient, 89
- queryInsertShift
 - RestaurantClient, 89
- queryInsertWorker
 - RestaurantClient, 89
- queryShiftsByDay
 - RestaurantClient, 90
- queryUpdateShift
 - RestaurantClient, 90
- queryUpdateWorker
 - RestaurantClient, 90
- queryWorkers
 - RestaurantClient, 91
- readAsync
 - Connection, 41
 - ConnectionBase, 50
- readSync
 - Connection, 41
 - ConnectionBase, 50
- readSyncInternal
 - Connection, 42
- receiveFromSocket
 - Connection, 42
- register_models
 - Serialization, 8
- remove
 - DataBag, 62
- removeHandlers
 - ConnectionObserver, 60
- resizeGrid
 - MainWindow, 76
- RestaurantClient, 85
 - authorize, 88
 - queryDeleteShift, 88
 - queryDeleteWorker, 89
 - queryInsertShift, 89
 - queryInsertWorker, 89
 - queryShiftsByDay, 90
 - queryUpdateShift, 90
 - queryUpdateWorker, 90
 - queryWorkers, 91
- RestaurantManager, 91
 - deleteShift, 93
 - deleteWorker, 94
 - deserialize, 94
 - getAccessTokens, 94
 - getShifts, 95
 - getShiftsByDay, 95
 - getType, 95
 - getWorkers, 95
 - insertShift, 96
 - insertWorker, 96
 - newIdentityId, 96
 - onPayloadReceived, 97
 - serialize, 97
 - verifyPermission, 98
 - verifyShift, 98
- S2C_AuthorizeReply, 99
 - deserialize, 101
 - getPermissions, 101
 - getType, 101
 - S2C_AuthorizeReply, 100
 - serialize, 102
 - setPermissions, 102

- S2C_ClientSync, 103
 - deserialize, 104
 - getChangedShifts, 104, 105
 - getChangedWorkers, 105
 - getRemovedShifts, 105, 106
 - getRemovedWorkers, 106
 - getType, 106
 - serialize, 107
- S2C_DeleteShiftReply, 108
 - deserialize, 110
 - getId, 110
 - getType, 110
 - S2C_DeleteShiftReply, 110
 - serialize, 111
 - setId, 111
- S2C_DeleteWorkerReply, 112
 - deserialize, 114
 - getId, 114
 - getType, 114
 - S2C_DeleteWorkerReply, 113
 - serialize, 114
 - setId, 115
- S2C_GetShiftsReply, 115
 - deserialize, 118
 - getDate, 118
 - getShifts, 118, 119
 - getType, 119
 - S2C_GetShiftsReply, 117
 - serialize, 119
 - setShifts, 120
- S2C_GetWorkersReply, 120
 - deserialize, 122
 - getType, 122
 - getWorkers, 123
 - S2C_GetWorkersReply, 122
 - serialize, 123
 - setWorkers, 123
- S2C_InsertShiftReply, 124
 - deserialize, 126
 - getShift, 127
 - getType, 127
 - S2C_InsertShiftReply, 126
 - serialize, 127
 - setShift, 128
- S2C_InsertWorkerReply, 128
 - deserialize, 131
 - getType, 131
 - getWorker, 131
 - S2C_InsertWorkerReply, 130
 - serialize, 132
 - setWorker, 132
- sendBuffer
 - Connection, 42
- Serialization, 7
 - register_models, 8
 - Type, 8
- Serialization::Serializable, 133
 - deserialize, 134
 - getType, 134
 - serialize, 134
- Serialization::TypeInfo, 169
- serialize
 - C2S_Authorize, 15
 - C2S_DeleteShift, 18
 - C2S_DeleteWorker, 22
 - C2S_GetShiftsByDay, 26
 - C2S_InsertShift, 32
 - C2S_InsertWorker, 37
 - Date, 66
 - DateTime, 72
 - RestaurantManager, 97
 - S2C_AuthorizeReply, 102
 - S2C_ClientSync, 107
 - S2C_DeleteShiftReply, 111
 - S2C_DeleteWorkerReply, 114
 - S2C_GetShiftsReply, 119
 - S2C_GetWorkersReply, 123
 - S2C_InsertShiftReply, 127
 - S2C_InsertWorkerReply, 132
 - Serialization::Serializable, 134
 - Shift, 148
 - ShiftWorker, 159
 - TrackablePacket, 163
 - TransactionReply, 167
- Server, 135
 - clients, 136
 - deleteConnectionAsync, 137
 - isRunning, 137
 - join, 137
 - listenWorker, 137
 - onConnected, 137
 - onDisconnected, 138
 - onPayloadReceived, 138
 - onPayloadSent, 138
 - prepareClient, 139
 - shouldAcceptSocket, 139
 - start, 139
 - stop, 140
 - subscribe, 140
 - unsubscribe, 140
 - writeToAll, 140
- ServerObserver, 141
 - getServer, 142
 - setServer, 142
- setData
 - memory_buf, 78
- setDate
 - C2S_GetShiftsByDay, 26
- setDay
 - Date, 67
- setErrorMsg
 - TransactionReply, 168
- setFirstName
 - ShiftWorker, 159
- setHour
 - DateTime, 73

- setId
 - S2C_DeleteShiftReply, 111
 - S2C_DeleteWorkerReply, 115
 - Shift, 148
 - ShiftWorker, 160
- setJobName
 - Shift, 149
- setLastName
 - ShiftWorker, 160
- setLength
 - memory_buf, 78
- setMainWindow
 - App, 11
- setMinute
 - DateTime, 73
- setModifyExisting
 - C2S_InsertShift, 33
 - C2S_InsertWorker, 37
- setMonth
 - Date, 67
- setPermissions
 - S2C_AuthorizeReply, 102
- setReadingAsync
 - Connection, 43
 - ConnectionBase, 51
- setRequestId
 - TrackablePacket, 164
- setSecond
 - DateTime, 73
- setServer
 - ServerObserver, 142
- setShift
 - C2S_InsertShift, 33
 - S2C_InsertShiftReply, 128
 - ShiftCellRenderer, 153
- setShiftId
 - C2S_DeleteShift, 19
- setShiftWindow
 - App, 11
- setShifts
 - S2C_GetShiftsReply, 120
- setStartTime
 - Shift, 149
- setSuccess
 - TransactionReply, 168
- setTitle
 - ShiftWorker, 160
- setToken
 - C2S_Authorize, 15
- setWorkHours
 - Shift, 149
- setWorker
 - C2S_InsertWorker, 37
 - S2C_InsertWorkerReply, 132
 - ShiftCellRenderer, 153
- setWorkerId
 - C2S_DeleteWorker, 22
 - Shift, 149
- setWorkerWindow
 - App, 12
- setWorkers
 - S2C_GetWorkersReply, 123
- setYear
 - Date, 67
- Shift, 143
 - deserialize, 146
 - getEndTime, 146
 - getId, 146
 - getJobName, 146
 - getStartTime, 147
 - getType, 147
 - getWorkHours, 148
 - getWorkerId, 147
 - serialize, 148
 - setId, 148
 - setJobName, 149
 - setStartTime, 149
 - setWorkHours, 149
 - setWorkerId, 149
 - Shift, 145
- ShiftCellRenderer, 150
 - Draw, 152
 - GetBestSize, 152
 - getShift, 153
 - getWorker, 153
 - setShift, 153
 - setWorker, 153
- ShiftEditorWindow, 154
 - getWorkerData, 155
 - ShiftEditorWindow, 155
- ShiftWorker, 156
 - deserialize, 157
 - getFirstName, 158
 - getId, 158
 - getLastName, 158
 - getTitle, 158
 - getType, 159
 - serialize, 159
 - setFirstName, 159
 - setId, 160
 - setLastName, 160
 - setTitle, 160
 - toString, 160
- shouldAcceptSocket
 - Server, 139
- start
 - Server, 139
- stop
 - Server, 140
- subscribe
 - ConnectionBase, 51
 - PingService, 84
 - Server, 140
- toString
 - ShiftWorker, 160
- TrackablePacket, 161

- deserialize, [163](#)
- getRequestId, [163](#)
- serialize, [163](#)
- setRequestId, [164](#)
- TrackablePacket, [162](#)
- TransactionReply, [164](#)
 - deserialize, [167](#)
 - getErrorMsg, [167](#)
 - isSuccess, [167](#)
 - serialize, [167](#)
 - setErrorMsg, [168](#)
 - setSuccess, [168](#)
 - TransactionReply, [166](#)
- Type
 - Serialization, [8](#)
- unsubscribe
 - ConnectionBase, [51](#)
 - PingService, [85](#)
 - Server, [140](#)
- verifyPermission
 - RestaurantManager, [98](#)
- verifyShift
 - RestaurantManager, [98](#)
- worker
 - PingService, [85](#)
- WorkerEditorWindow, [169](#)
 - WorkerEditorWindow, [170](#)
- writeReplySync
 - ConnectionBase, [52](#)
- writeRequestSync
 - ConnectionBase, [52](#)
- writeSync
 - Connection, [43](#)
 - ConnectionBase, [53](#)
- writeToAll
 - Server, [140](#)