

QRodSystems

0.0.7

Generated by Doxygen 1.9.1



<b>1 Hierarchical Index</b>	<b>1</b>
1.1 Class Hierarchy	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 File Index</b>	<b>5</b>
3.1 File List	5
<b>4 Class Documentation</b>	<b>9</b>
4.1 QRS::Core::AbstractDataObject Class Reference	9
4.1.1 Detailed Description	10
4.1.2 Member Function Documentation	10
4.1.2.1 deserialize()	10
4.1.2.2 getAvailableItemKey()	11
4.2 QRS::HierarchyModels::AbstractHierarchyItem Class Reference	11
4.2.1 Detailed Description	12
4.3 QRS::HierarchyModels::AbstractHierarchyModel Class Reference	12
4.3.1 Detailed Description	13
4.4 QRS::Core::Array< T > Class Template Reference	13
4.4.1 Detailed Description	14
4.5 QRS::TableModels::BaseTableModel Class Reference	14
4.5.1 Detailed Description	15
4.6 QRS::HierarchyModels::DataObjectsHierarchyItem Class Reference	15
4.6.1 Detailed Description	16
4.7 QRS::HierarchyModels::DataObjectsHierarchyModel Class Reference	16
4.7.1 Detailed Description	17
4.8 QRS::Managers::DataObjectsManager Class Reference	18
4.8.1 Detailed Description	19
4.9 QRS::Managers::DoubleSpinBoxItemDelegate Class Reference	20
4.9.1 Detailed Description	20
4.10 QRS::Core::HierarchyNode Class Reference	20
4.10.1 Detailed Description	21
4.11 QRS::Core::HierarchyTree Class Reference	22
4.11.1 Detailed Description	23
4.12 QRS::App::LogWidget Class Reference	23
4.12.1 Detailed Description	23
4.13 QRS::App::MainWindow Class Reference	24
4.13.1 Detailed Description	25
4.14 QRS::App::ManagersTab Class Reference	26
4.14.1 Detailed Description	26
4.15 QRS::Core::MatrixDataObject Class Reference	26
4.15.1 Detailed Description	27

4.16 QRS::TableModels::MatrixTableModel Class Reference . . . . .	27
4.16.1 Detailed Description . . . . .	28
4.17 QRS::Core::Project Class Reference . . . . .	28
4.17.1 Detailed Description . . . . .	30
4.18 QRS::HierarchyModels::ProjectHierarchyModel Class Reference . . . . .	30
4.18.1 Detailed Description . . . . .	31
4.19 QRS::Core::Array< T >::Row< U > Struct Template Reference . . . . .	31
4.19.1 Detailed Description . . . . .	31
4.20 QRS::Core::ScalarDataObject Class Reference . . . . .	32
4.20.1 Detailed Description . . . . .	32
4.21 QRS::Core::SurfaceDataObject Class Reference . . . . .	33
4.21.1 Detailed Description . . . . .	34
4.22 QRS::TableModels::SurfaceTableModel Class Reference . . . . .	34
4.22.1 Detailed Description . . . . .	35
4.23 QRS::TableModels::TableModelInterface Class Reference . . . . .	35
4.23.1 Detailed Description . . . . .	36
4.24 QRS::Core::VectorDataObject Class Reference . . . . .	36
4.24.1 Detailed Description . . . . .	37
4.25 QRS::Graph::View3D Class Reference . . . . .	37
4.25.1 Detailed Description . . . . .	37
<b>5 File Documentation . . . . .</b>	<b>39</b>
5.1 /home/qinterfly/Library/Projects/QRodSystems/src/central/controltabs.cpp File Reference . . . . .	39
5.1.1 Detailed Description . . . . .	39
5.2 /home/qinterfly/Library/Projects/QRodSystems/src/central/controltabs.h File Reference . . . . .	39
5.2.1 Detailed Description . . . . .	40
5.3 /home/qinterfly/Library/Projects/QRodSystems/src/central/logwidget.cpp File Reference . . . . .	40
5.3.1 Detailed Description . . . . .	40
5.4 /home/qinterfly/Library/Projects/QRodSystems/src/central/logwidget.h File Reference . . . . .	41
5.4.1 Detailed Description . . . . .	41
5.5 /home/qinterfly/Library/Projects/QRodSystems/src/central/mainwindow.cpp File Reference . . . . .	41
5.5.1 Detailed Description . . . . .	42
5.6 /home/qinterfly/Library/Projects/QRodSystems/src/central/mainwindow.h File Reference . . . . .	42
5.6.1 Detailed Description . . . . .	42
5.7 /home/qinterfly/Library/Projects/QRodSystems/src/central/projecthierarchymodel.cpp File Reference . . . . .	43
5.7.1 Detailed Description . . . . .	43
5.8 /home/qinterfly/Library/Projects/QRodSystems/src/central/projecthierarchymodel.h File Reference . . . . .	43
5.8.1 Detailed Description . . . . .	43
5.9 /home/qinterfly/Library/Projects/QRodSystems/src/central/uiconstants.h File Reference . . . . .	44
5.9.1 Detailed Description . . . . .	44
5.10 /home/qinterfly/Library/Projects/QRodSystems/src/core/abstractdataobject.cpp File Reference . . . . .	44
5.10.1 Detailed Description . . . . .	44

5.11 /home/qinterfly/Library/Projects/QRodSystems/src/core/abstractdataobject.h File Reference . . . . .	45
5.11.1 Detailed Description . . . . .	45
5.12 /home/qinterfly/Library/Projects/QRodSystems/src/core/array.cpp File Reference . . . . .	45
5.12.1 Detailed Description . . . . .	46
5.13 /home/qinterfly/Library/Projects/QRodSystems/src/core/array.h File Reference . . . . .	46
5.13.1 Detailed Description . . . . .	47
5.14 /home/qinterfly/Library/Projects/QRodSystems/src/core/datatypes.h File Reference . . . . .	47
5.14.1 Detailed Description . . . . .	47
5.15 /home/qinterfly/Library/Projects/QRodSystems/src/core/hierarchyndnode.cpp File Reference . . . . .	47
5.15.1 Detailed Description . . . . .	48
5.16 /home/qinterfly/Library/Projects/QRodSystems/src/core/hierarchyndnode.h File Reference . . . . .	48
5.16.1 Detailed Description . . . . .	48
5.17 /home/qinterfly/Library/Projects/QRodSystems/src/core/hierarchytree.cpp File Reference . . . . .	48
5.17.1 Detailed Description . . . . .	49
5.18 /home/qinterfly/Library/Projects/QRodSystems/src/core/hierarchytree.h File Reference . . . . .	49
5.18.1 Detailed Description . . . . .	49
5.19 /home/qinterfly/Library/Projects/QRodSystems/src/core/matrixdataobject.cpp File Reference . . . . .	50
5.19.1 Detailed Description . . . . .	50
5.20 /home/qinterfly/Library/Projects/QRodSystems/src/core/matrixdataobject.h File Reference . . . . .	50
5.20.1 Detailed Description . . . . .	50
5.21 /home/qinterfly/Library/Projects/QRodSystems/src/core/project.cpp File Reference . . . . .	51
5.21.1 Detailed Description . . . . .	51
5.22 /home/qinterfly/Library/Projects/QRodSystems/src/core/project.h File Reference . . . . .	51
5.22.1 Detailed Description . . . . .	52
5.23 /home/qinterfly/Library/Projects/QRodSystems/src/core/scalardataobject.cpp File Reference . . . . .	52
5.23.1 Detailed Description . . . . .	52
5.24 /home/qinterfly/Library/Projects/QRodSystems/src/core/scalardataobject.h File Reference . . . . .	52
5.24.1 Detailed Description . . . . .	53
5.25 /home/qinterfly/Library/Projects/QRodSystems/src/core/surfacedataobject.cpp File Reference . . . . .	53
5.25.1 Detailed Description . . . . .	53
5.26 /home/qinterfly/Library/Projects/QRodSystems/src/core/surfacedataobject.h File Reference . . . . .	53
5.26.1 Detailed Description . . . . .	54
5.27 /home/qinterfly/Library/Projects/QRodSystems/src/core/utilities.cpp File Reference . . . . .	54
5.27.1 Detailed Description . . . . .	54
5.28 /home/qinterfly/Library/Projects/QRodSystems/src/core/utilities.h File Reference . . . . .	54
5.28.1 Detailed Description . . . . .	55
5.29 /home/qinterfly/Library/Projects/QRodSystems/src/core/vectordataobject.cpp File Reference . . . . .	55
5.29.1 Detailed Description . . . . .	55
5.30 /home/qinterfly/Library/Projects/QRodSystems/src/core/vectordataobject.h File Reference . . . . .	56
5.30.1 Detailed Description . . . . .	56
5.31 /home/qinterfly/Library/Projects/QRodSystems/src/main/main.cpp File Reference . . . . .	56
5.31.1 Detailed Description . . . . .	56

5.32	/home/qinterfly/Library/Projects/QRodSystems/src/managers/basetablemodel.cpp File Reference	57
5.32.1	Detailed Description	57
5.33	/home/qinterfly/Library/Projects/QRodSystems/src/managers/basetablemodel.h File Reference	57
5.33.1	Detailed Description	57
5.34	/home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectshierarchyitem.cpp File Reference	58
5.34.1	Detailed Description	58
5.35	/home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectshierarchyitem.h File Reference	58
5.35.1	Detailed Description	59
5.36	/home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectshierarchyitemmodel.cpp File Reference	59
5.36.1	Detailed Description	59
5.37	/home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectshierarchyitemmodel.h File Reference	59
5.37.1	Detailed Description	60
5.38	/home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectsmanager.cpp File Reference	60
5.38.1	Detailed Description	61
5.39	/home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectsmanager.h File Reference	61
5.39.1	Detailed Description	62
5.40	/home/qinterfly/Library/Projects/QRodSystems/src/managers/doublespinboxitemdelegate.cpp File Reference	62
5.40.1	Detailed Description	62
5.41	/home/qinterfly/Library/Projects/QRodSystems/src/managers/doublespinboxitemdelegate.h File Reference	62
5.41.1	Detailed Description	63
5.42	/home/qinterfly/Library/Projects/QRodSystems/src/managers/matrixtablemodel.cpp File Reference	63
5.42.1	Detailed Description	63
5.43	/home/qinterfly/Library/Projects/QRodSystems/src/managers/matrixtablemodel.h File Reference	63
5.43.1	Detailed Description	64
5.44	/home/qinterfly/Library/Projects/QRodSystems/src/managers/surfacetablemodel.cpp File Reference	64
5.44.1	Detailed Description	64
5.45	/home/qinterfly/Library/Projects/QRodSystems/src/managers/surfacetablemodel.h File Reference	64
5.45.1	Detailed Description	65
5.46	/home/qinterfly/Library/Projects/QRodSystems/src/managers/tablemodelinterface.cpp File Reference	65
5.46.1	Detailed Description	65
5.47	/home/qinterfly/Library/Projects/QRodSystems/src/managers/tablemodelinterface.h File Reference	65
5.47.1	Detailed Description	66
5.48	/home/qinterfly/Library/Projects/QRodSystems/src/models/abstrachierarchyitem.cpp File Reference	66
5.48.1	Detailed Description	66
5.49	/home/qinterfly/Library/Projects/QRodSystems/src/models/abstrachierarchyitem.h File Reference	66
5.49.1	Detailed Description	67
5.50	/home/qinterfly/Library/Projects/QRodSystems/src/models/abstrachierarchyitemmodel.cpp File Reference	67
5.50.1	Detailed Description	67

---

5.51 /home/qinterfly/Library/Projects/QRodSystems/src/models/abstracthierarchymodel.h File Reference	67
5.51.1 Detailed Description	68
5.52 /home/qinterfly/Library/Projects/QRodSystems/src/render/view3d.cpp File Reference	68
5.52.1 Detailed Description	68
5.53 /home/qinterfly/Library/Projects/QRodSystems/src/render/view3d.h File Reference	68
5.53.1 Detailed Description	69





# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

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

QRS::Core::AbstractDataObject . . . . .	9
QRS::Core::MatrixDataObject . . . . .	26
QRS::Core::ScalarDataObject . . . . .	32
QRS::Core::SurfaceDataObject . . . . .	33
QRS::Core::VectorDataObject . . . . .	36
QRS::Core::Array< T > . . . . .	13
QRS::Core::HierarchyNode . . . . .	20
QRS::Core::HierarchyTree . . . . .	22
QDialog	
QRS::Managers::DataObjectsManager . . . . .	18
QMainWindow	
QRS::App::MainWindow . . . . .	24
QObject	
QRS::Core::Project . . . . .	28
QOpenGLFunctions	
QRS::Graph::View3D . . . . .	37
QOpenGLWidget	
QRS::Graph::View3D . . . . .	37
QStandardItem	
QRS::HierarchyModels::AbstractHierarchyItem . . . . .	11
QRS::HierarchyModels::DataObjectsHierarchyItem . . . . .	15
QStandardItemModel	
QRS::HierarchyModels::AbstractHierarchyModel . . . . .	12
QRS::HierarchyModels::DataObjectsHierarchyModel . . . . .	16
QRS::HierarchyModels::ProjectHierarchyModel . . . . .	30
QRS::TableModels::BaseTableModel . . . . .	14
QRS::TableModels::MatrixTableModel . . . . .	27
QRS::TableModels::SurfaceTableModel . . . . .	34
QStyledItemDelegate	
QRS::Managers::DoubleSpinBoxItemDelegate . . . . .	20
QTableWidget	
QRS::App::LogWidget . . . . .	23
QWidget	
QRS::App::ManagersTab . . . . .	26
QRS::Core::Array< T >::Row< U > . . . . .	31

QRS::TableModels::TableModelInterface . . . . .	35
QRS::TableModels::BaseTableModel . . . . .	14
QRS::TableModels::MatrixTableModel . . . . .	27
QRS::TableModels::SurfaceTableModel . . . . .	34

## Chapter 2

# Class Index

### 2.1 Class List

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

<a href="#">QRS::Core::AbstractDataObject</a>	9
Data object which is designed in the way to be represented in a table easily . . . . .	
<a href="#">QRS::HierarchyModels::AbstractHierarchyItem</a>	11
Item to represent a hierarchy of elements of the same type . . . . .	
<a href="#">QRS::HierarchyModels::AbstractHierarchyModel</a>	12
Hierarchy model which enables one to drag and drop elements of the same type . . . . .	
<a href="#">QRS::Core::Array&lt; T &gt;</a>	13
Numerical array class . . . . .	
<a href="#">QRS::TableModels::BaseTableModel</a>	14
Table model to represent either a scalar or vector data object . . . . .	
<a href="#">QRS::HierarchyModels::DataObjectsHierarchyItem</a>	15
Item to represent a hierarchy of data objects . . . . .	
<a href="#">QRS::HierarchyModels::DataObjectsHierarchyModel</a>	16
Tree model to represent and modify a hierarchy of data objects . . . . .	
<a href="#">QRS::Managers::DataObjectsManager</a>	18
Manager to create objects of different types: scalars, vectors, matrices and surfaces . . . . .	
<a href="#">QRS::Managers::DoubleSpinBoxItemDelegate</a>	20
Class to specify how table values can be edited . . . . .	
<a href="#">QRS::Core::HierarchyNode</a>	20
Hierarchy representative . . . . .	
<a href="#">QRS::Core::HierarchyTree</a>	22
Hierarchy of data objects (n-ary tree) . . . . .	
<a href="#">QRS::App::LogWidget</a>	23
Log all the messages sent . . . . .	
<a href="#">QRS::App::MainWindow</a>	24
The main window of the program . . . . .	
<a href="#">QRS::App::ManagersTab</a>	26
A toolbar consisted of object designers . . . . .	
<a href="#">QRS::Core::MatrixDataObject</a>	26
Matrix data object . . . . .	
<a href="#">QRS::TableModels::MatrixTableModel</a>	27
Table model to represent a matrix data object . . . . .	
<a href="#">QRS::Core::Project</a>	28
<a href="#">Project</a> class to interact with a created system of rods . . . . .	
<a href="#">QRS::HierarchyModels::ProjectHierarchyModel</a>	30
Project hierarchy representative . . . . .	

<a href="#">QRS::Core::Array&lt; T &gt;::Row&lt; U &gt;</a>	
Proxy class to acquire a row by index . . . . .	31
<a href="#">QRS::Core::ScalarDataObject</a>	
Scalar data object . . . . .	32
<a href="#">QRS::Core::SurfaceDataObject</a>	
Surface data object . . . . .	33
<a href="#">QRS::TableModels::SurfaceTableModel</a>	
Table model to represent a surface data object . . . . .	34
<a href="#">QRS::TableModels::TableModelInterface</a>	
User interface to add and remove items . . . . .	35
<a href="#">QRS::Core::VectorDataObject</a>	
Vector data object . . . . .	36
<a href="#">QRS::Graph::View3D</a>	
A widget to represent the resulted rod system . . . . .	37

## Chapter 3

# File Index

### 3.1 File List

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

/home/qinterfly/Library/Projects/QRodSystems/src/central/ <a href="#">controltabs.cpp</a>	
Implementation of the ControlTabs class . . . . .	39
/home/qinterfly/Library/Projects/QRodSystems/src/central/ <a href="#">controltabs.h</a>	
Declaration of the ControlTabs class . . . . .	39
/home/qinterfly/Library/Projects/QRodSystems/src/central/ <a href="#">logwidget.cpp</a>	
Implementation of the LogWidget class . . . . .	40
/home/qinterfly/Library/Projects/QRodSystems/src/central/ <a href="#">logwidget.h</a>	
Declaration of the LogWidget class . . . . .	41
/home/qinterfly/Library/Projects/QRodSystems/src/central/ <a href="#">mainwindow.cpp</a>	
Implementation of the MainWindow class . . . . .	41
/home/qinterfly/Library/Projects/QRodSystems/src/central/ <a href="#">mainwindow.h</a>	
Declaration of the MainWindow class . . . . .	42
/home/qinterfly/Library/Projects/QRodSystems/src/central/ <a href="#">projecthierarchymodel.cpp</a>	
Definition of the ProjectHierarchyModel class . . . . .	43
/home/qinterfly/Library/Projects/QRodSystems/src/central/ <a href="#">projecthierarchymodel.h</a>	
Declaration of the ProjectHierarchyModel class . . . . .	43
/home/qinterfly/Library/Projects/QRodSystems/src/central/ <a href="#">uiconstants.h</a>	
Common graphical constants shared between several windows . . . . .	44
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">abstractdataobject.cpp</a>	
Implementation of the AbstractDataObject class . . . . .	44
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">abstractdataobject.h</a>	
Declaration of the AbstractDataObject class . . . . .	45
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">array.cpp</a>	
Implementation of the Array class . . . . .	45
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">array.h</a>	
Declaration of the Array class . . . . .	46
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">datatypes.h</a>	
Specification of data types used in a project . . . . .	47
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">hierarchynode.cpp</a>	
Implementation of the HierarchyNode class . . . . .	47
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">hierarchynode.h</a>	
Declaration of the HierarchyNode class . . . . .	48
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">hierarchytree.cpp</a>	
Implementation of the HierarchyTree class . . . . .	48
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">hierarchytree.h</a>	
Declaration of the HierarchyTree class . . . . .	49

/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">matrixdataobject.cpp</a>	
Implementation of the MatrixDataObject class . . . . .	50
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">matrixdataobject.h</a>	
Declaration of the MatrixDataObject class . . . . .	50
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">project.cpp</a>	
Implementation of the Project class . . . . .	51
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">project.h</a>	
Declaration of the Project class . . . . .	51
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">scalardataobject.cpp</a>	
Implementation of the ScalarDataObject class . . . . .	52
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">scalardataobject.h</a>	
Declaration of the ScalarDataObject class . . . . .	52
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">surfacedataobject.cpp</a>	
Implementation of the SurfaceDataObject class . . . . .	53
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">surfacedataobject.h</a>	
Declaration of the SurfaceDataObject class . . . . .	53
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">utilities.cpp</a>	
Implementation of utilities . . . . .	54
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">utilities.h</a>	
Declaration of utilities . . . . .	54
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">vectordataobject.cpp</a>	
Implementation of the VectorDataObject class . . . . .	55
/home/qinterfly/Library/Projects/QRodSystems/src/core/ <a href="#">vectordataobject.h</a>	
Declaration of the VectorDataObject class . . . . .	56
/home/qinterfly/Library/Projects/QRodSystems/src/main/ <a href="#">main.cpp</a>	
The startup function . . . . .	56
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">basetablemodel.cpp</a>	
Implementation of the BaseTableModel class . . . . .	57
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">basetablemodel.h</a>	
Declaration of the BaseTableModel class . . . . .	57
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">dataobjectshierarchyitem.cpp</a>	
Definition of the DataObjectsHierarchyItem class . . . . .	58
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">dataobjectshierarchyitem.h</a>	
Declaration of the DataObjectsHierarchyItem class . . . . .	58
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">dataobjectshierarchymodel.cpp</a>	
Definition of the DataObjectsHierarchyModel class . . . . .	59
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">dataobjectshierarchymodel.h</a>	
Declaration of the DataObjectsHierarchyModel class . . . . .	59
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">dataobjectsmanager.cpp</a>	
Implementation of the DataObjectsManager class . . . . .	60
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">dataobjectsmanager.h</a>	
Declaration of the DataObjectsManager class . . . . .	61
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">doublespinboxitemdelegate.cpp</a>	
Implementation of the DoubleSpinBoxItemDelegate class . . . . .	62
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">doublespinboxitemdelegate.h</a>	
Declaration of the DoubleSpinBoxItemDelegate class . . . . .	62
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">matrixtablemodel.cpp</a>	
Implementation of the MatrixTableModel class . . . . .	63
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">matrixtablemodel.h</a>	
Declaration of the MatrixTableModel class . . . . .	63
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">surfactablemodel.cpp</a>	
Implementation of the SurfaceTableModel class . . . . .	64
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">surfactablemodel.h</a>	
Declaration of the SurfaceTableModel class . . . . .	64
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">tablemodelinterface.cpp</a>	
Implementation of static functions of TableModelInterface . . . . .	65
/home/qinterfly/Library/Projects/QRodSystems/src/managers/ <a href="#">tablemodelinterface.h</a>	
Declaration of the TableModelInterface . . . . .	65

/home/qinterfly/Library/Projects/QRodSystems/src/models/ <a href="#">abstracthierarchyitem.cpp</a>	
Definition of the AbstractHierarchyItem class . . . . .	66
/home/qinterfly/Library/Projects/QRodSystems/src/models/ <a href="#">abstracthierarchyitem.h</a>	
Declaration of the AbstractHierarchyItem class . . . . .	66
/home/qinterfly/Library/Projects/QRodSystems/src/models/ <a href="#">abstracthierarchymodel.cpp</a>	
Definition of the AbstractHierarchyModel class . . . . .	67
/home/qinterfly/Library/Projects/QRodSystems/src/models/ <a href="#">abstracthierarchymodel.h</a>	
Declaration of the AbstractHierarchyModel class . . . . .	67
/home/qinterfly/Library/Projects/QRodSystems/src/render/ <a href="#">view3d.cpp</a>	
Implementation of the View3D class . . . . .	68
/home/qinterfly/Library/Projects/QRodSystems/src/render/ <a href="#">view3d.h</a>	
Declaration of the View3D class . . . . .	68





## Chapter 4

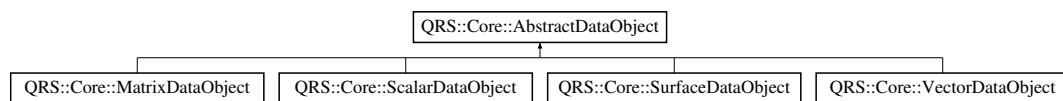
# Class Documentation

### 4.1 QRS::Core::AbstractDataObject Class Reference

Data object which is designed in the way to be represented in a table easily.

```
#include <abstractdataobject.h>
```

Inheritance diagram for QRS::Core::AbstractDataObject:



#### Public Member Functions

- [AbstractDataObject](#) (DataObjectType type, QString const &name)  
*Base constructor.*
- virtual [AbstractDataObject](#) \* **clone** () const =0
- virtual [DataItemType](#) & **addItem** (DataKeyType key)=0
- void [removeItem](#) (DataValueType key)  
*Remove an entity with the specified key.*
- bool [changeItemKey](#) (DataKeyType oldKey, DataKeyType newKey, DataHolder \*items=nullptr)  
*Modify a key existed.*
- DataValueType [getAvailableItemKey](#) (DataValueType key, DataHolder const \*items=nullptr) const
- bool [setArrayValue](#) (DataKeyType key, DataValueType newValue, uint iRow=0, uint iColumn=0)  
*Set an array value with the specified indices.*
- DataHolder & **getItems** ()
- [DataItemType](#) & **getItem** (DataValueType keyParameter)
- DataIDType **id** () const
- DataObjectType **type** () const
- QString const & **name** () const
- void **setName** (QString const &name)
- virtual void [serialize](#) (QDataStream &stream) const  
*Serialize an abstract data object.*
- virtual void [deserialize](#) (QDataStream &stream)  
*Partly deserialize an abstract data object.*
- virtual void **import** (QTextStream &stream)=0

## Static Public Member Functions

- static uint **numberOfObjects** ()
- static void **setNumberOfObjects** (uint numObjects)

## Protected Attributes

- const DataObjectType **mType**  
*Object type.*
- QString **mName**  
*Name of an object.*
- DataIDType **mID**  
*Unique object identificator.*
- DataHolder **mItems**  
*Map contains all created entities.*

## Static Private Attributes

- static uint **smNumObjects** = 0  
*Number of all objects created.*

## Friends

- QDataStream & **operator<<** (QDataStream &stream, [AbstractDataObject](#) const &obj)  
*Print a data object to a stream.*
- QDataStream & **operator>>** (QDataStream &stream, [AbstractDataObject](#) &obj)  
*Read a data object from a stream.*

### 4.1.1 Detailed Description

Data object which is designed in the way to be represented in a table easily.

### 4.1.2 Member Function Documentation

#### 4.1.2.1 deserialize()

```
void AbstractDataObject::deserialize (
    QDataStream & stream ) [virtual]
```

Partly deserialize an abstract data object.

It is assumed that a type and name have already been assigned. So, only an identifier and items need to be set.

Reimplemented in [QRS::Core::SurfaceDataObject](#).

## 4.1.2.2 getAvailableItemKey()

```
DataValueType AbstractDataObject::getAvailableItemKey (
    DataValueType key,
    DataHolder const * items = nullptr ) const
```

Check if a given key is unique

## Returns

Returns the input value of the key if it is unique, otherwise – a first available key

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

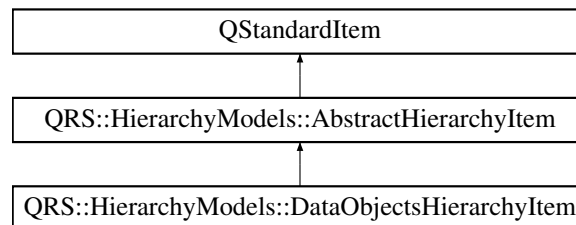
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[abstractdataobject.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[abstractdataobject.cpp](#)

## 4.2 QRS::HierarchyModels::AbstractHierarchyItem Class Reference

Item to represent a hierarchy of elements of the same type.

```
#include <abstrachierarchyitem.h>
```

Inheritance diagram for QRS::HierarchyModels::AbstractHierarchyItem:



## Public Member Functions

- **AbstractHierarchyItem** (const QIcon &icon, const QString &text, [Core::HierarchyNode](#) \*pNode)
- void [writePointer](#) (QDataStream &out) const  
*Write the pointer to the current item to a stream.*
- virtual int **type** () const =0

## Static Public Member Functions

- static [AbstractHierarchyItem](#) \* [readPointer](#) (QDataStream &in)  
*Retrieve a pointer to an item from a stream.*

## Protected Attributes

- [Core::HierarchyNode](#) \* **mpNode** = nullptr

## Friends

- class **AbstractHierarchyModel**

### 4.2.1 Detailed Description

Item to represent a hierarchy of elements of the same type.

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

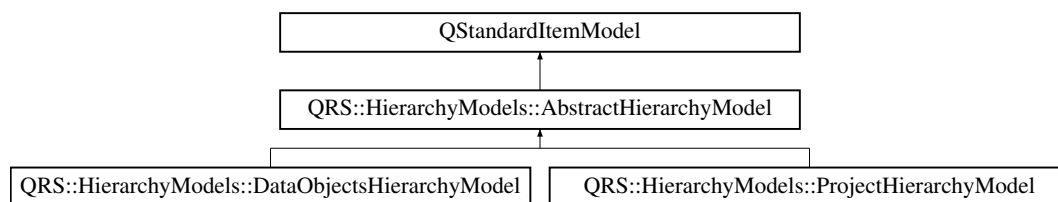
- /home/qinterfly/Library/Projects/QRodSystems/src/models/[abstrachierarchyitem.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/models/[abstrachierarchyitem.cpp](#)

## 4.3 QRS::HierarchyModels::AbstractHierarchyModel Class Reference

Hierarchy model which enables one to drag and drop elements of the same type.

```
#include <abstrachierarchymodel.h>
```

Inheritance diagram for QRS::HierarchyModels::AbstractHierarchyModel:



## Signals

- void **dataModified** (bool flag)

## Public Member Functions

- **AbstractHierarchyModel** (QString const &mimeType, QTreeView \*pView=nullptr)
- virtual void **updateContent** ()=0
- virtual void **clearContent** ()=0
- Qt::DropActions **supportedDragActions** () const override  
*Specify allowed drag actions.*
- Qt::DropActions **supportedDropActions** () const override  
*Specify allowed drop actions.*
- QStringList **mimeTypes** () const override  
*Retrieve the mime types.*
- QMimeData \* **mimeTypeData** (const QModelIndexList &indices) const override  
*Encode each item according to a given list of indices.*
- bool **dropMimeType** (QMimeData const \*pMimeData, Qt::DropAction action, int row, int column, const QModelIndex &parent) override  
*Process the drop action.*

## Protected Attributes

- QString const **kMimeType**

## Private Member Functions

- bool [processDropOnItem](#) (QDataStream &stream, int &numItems, QModelIndex const &indexParent)  
*Merge several items into one entity.*
- bool [processDropBetweenItems](#) (QDataStream &stream, int &numItems, QModelIndex const &indexParent, int row)  
*Change the order of items.*

### 4.3.1 Detailed Description

Hierarchy model which enables one to drag and drop elements of the same type.

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

- /home/qinterfly/Library/Projects/QRodSystems/src/models/[abstracthierarchymodel.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/models/[abstracthierarchymodel.cpp](#)

## 4.4 QRS::Core::Array< T > Class Template Reference

Numerical array class.

```
#include <array.h>
```

## Classes

- struct [Row](#)  
*Proxy class to acquire a row by index.*

## Public Member Functions

- **Array** (IndexType numRows=0, IndexType numCols=0)
- [Array](#) ([Array](#)< T > const &another)  
*Copy constructor.*
- [Array](#) ([Array](#)< T > &&another)  
*Move constructor.*
- T \* **data** ()
- void [resize](#) (IndexType numRows, IndexType numCols)  
*Resize and copy previous values if possible.*
- void [removeColumn](#) (IndexType iRemoveColumn)  
*Remove a column by index.*
- void [swapColumns](#) (IndexType iFirstColumn, IndexType iSecondColumn)  
*Swap two columns.*
- IndexType **rows** () const
- IndexType **cols** () const
- IndexType **size** () const
- [Row](#)< T > **operator[]** (IndexType iRow)

## Private Attributes

- IndexType `mNumRows`  
*Number of rows.*
- IndexType `mNumCols`  
*Number of columns.*
- T \* `mpData` = nullptr  
*Pointer to the data stored.*

## Friends

- template<typename K >  
QDebug `operator<<` (QDebug stream, `Array< K >` &array)  
*Print all array values using the matrix format.*
- template<typename K >  
QDataStream & `operator<<` (QDataStream &stream, `Array< K >` const &array)  
*Write an array to a stream.*
- template<typename K >  
QDataStream & `operator>>` (QDataStream &stream, `Array< K >` &array)  
*Read an array from a stream.*

### 4.4.1 Detailed Description

```
template<typename T>
class QRS::Core::Array< T >
```

Numerical array class.

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

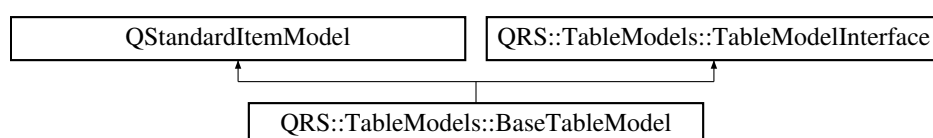
- /home/qinterfly/Library/Projects/QRodSystems/src/core/array.h
- /home/qinterfly/Library/Projects/QRodSystems/src/core/array.cpp

## 4.5 QRS::TableModels::BaseTableModel Class Reference

Table model to represent either a scalar or vector data object.

```
#include <basetablemodel.h>
```

Inheritance diagram for QRS::TableModels::BaseTableModel:



## Public Member Functions

- **BaseTableModel** (QWidget \*parent=nullptr)
- void **setDataObject** (Core::AbstractDataObject \*pDataObject)  
*Set a data object to represent.*
- bool **setData** (const QModelIndex &indexEdit, const QVariant &value, int role=Qt::EditRole) override  
*Set the data acquired from a delegate.*
- void **insertItemAfterSelected** (QItemSelectionModel \*pSelectionModel) override  
*Insert a new item after selected one.*
- void **insertLeadingItemAfterSelected** (QItemSelectionModel \*) override
- void **removeSelectedItem** (QItemSelectionModel \*pSelectionModel) override  
*Remove an array under selection.*
- void **removeSelectedLeadingItem** (QItemSelectionModel \*) override

## Private Member Functions

- void **updateContent** ()  
*Represent all items which a data object contains.*
- void **clearContent** ()  
*Clear previously created items.*

## Private Attributes

- Core::AbstractDataObject \* **mpDataObject** = nullptr

## Additional Inherited Members

### 4.5.1 Detailed Description

Table model to represent either a scalar or vector data object.

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

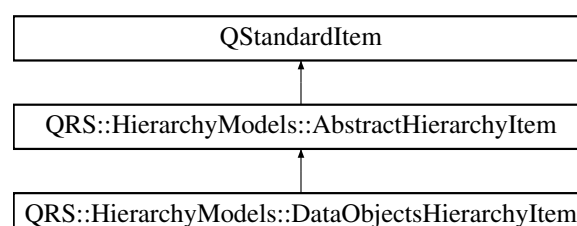
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/basetablemodel.h
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/basetablemodel.cpp

## 4.6 QRS::HierarchyModels::DataObjectsHierarchyItem Class Reference

Item to represent a hierarchy of data objects.

```
#include <dataobjectshierarchyitem.h>
```

Inheritance diagram for QRS::HierarchyModels::DataObjectsHierarchyItem:



## Public Member Functions

- [DataObjectsHierarchyItem](#) (mapDataObjects &dataObjects, [Core::HierarchyTree](#) &hierarchyDataObjects, QString const &name="Root")  
*Create the representer of the structure of data objects.*
- [DataObjectsHierarchyItem](#) ([Core::HierarchyNode](#) \*pNode, [Core::AbstractDataObject](#) \*pDataObject)  
*Construct an item to represent a data object.*
- [DataObjectsHierarchyItem](#) ([Core::HierarchyNode](#) \*pNode)  
*Construct an item to represent a directory.*
- int **type** () const override

## Private Member Functions

- void **appendItems** (mapDataObjects &dataObjects, [Core::HierarchyNode](#) \*pNode)

## Private Attributes

- [Core::AbstractDataObject](#) \* **mpDataObject** = nullptr

## Friends

- class **DataObjectsHierarchyModel**

## Additional Inherited Members

### 4.6.1 Detailed Description

Item to represent a hierarchy of data objects.

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

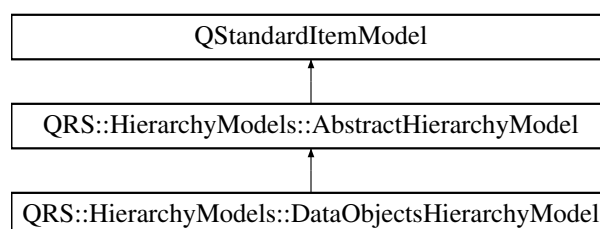
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/[dataobjectshierarchyitem.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/[dataobjectshierarchyitem.cpp](#)

## 4.7 QRS::HierarchyModels::DataObjectsHierarchyModel Class Reference

Tree model to represent and modify a hierarchy of data objects.

```
#include <dataobjectshierarchymodel.h>
```

Inheritance diagram for QRS::HierarchyModels::DataObjectsHierarchyModel:





## Public Slots

- void [retrieveSelectedDataObject](#) ()  
*Retrieve a selected data object.*
- void [removeSelectedItems](#) ()  
*Remove data objects under selection.*

## Signals

- void **dataObjectSelected** (Core::DataIDType id)
- void **selectionCleared** ()

## Public Member Functions

- **DataObjectsHierarchyModel** (mapDataObjects &dataObjects, [Core::HierarchyTree](#) &hierarchyDataObjects, QTreeView \*pView=nullptr)
- void [updateContent](#) () override  
*Update all the content.*
- void [clearContent](#) () override  
*Clear all the items.*
- bool [isEmpty](#) () const  
*Check if there are data objects to represent.*
- void [selectItem](#) (int iRow)  
*Select an item by row index.*

## Private Slots

- void [renameDataObject](#) (QStandardItem \*pStandardItem)  
*Rename a data object after editing.*

## Private Attributes

- mapDataObjects & **mDataObjects**
- [Core::HierarchyTree](#) & **mHierarchyDataObjects**

## Additional Inherited Members

### 4.7.1 Detailed Description

Tree model to represent and modify a hierarchy of data objects.

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

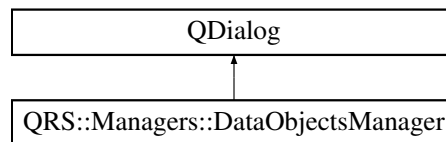
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/[dataobjectshierarchymodel.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/[dataobjectshierarchymodel.cpp](#)

## 4.8 QRS::Managers::DataObjectsManager Class Reference

Manager to create objects of different types: scalars, vectors, matroces and surfaces.

```
#include <dataobjectsmanager.h>
```

Inheritance diagram for QRS::Managers::DataObjectsManager:



### Public Slots

- void [apply](#) ()  
*Apply all the changes made by user.*
- Core::DataIDType [addScalar](#) ()  
*Add a scalar object.*
- Core::DataIDType [addVector](#) ()  
*Add a vector object.*
- Core::DataIDType [addMatrix](#) ()  
*Add a matrix object.*
- Core::DataIDType [addSurface](#) ()  
*Add a surface object.*
- void [insertItemAfterSelected](#) ()  
*Insert a new array into the data object.*
- void [insertLeadingItemAfterSelected](#) ()  
*Insert a new leading item into the data object.*
- void [removeSelectedItem](#) ()  
*Remove a selected item.*
- void [removeSelectedLeadingItem](#) ()  
*Remove a selected leading item.*
- void [importDataObjects](#) ()  
*Import data objects from a file.*
- void [representDataObject](#) (Core::DataIDType id)  
*Represent a selected data object according to its type.*
- void [clearDataObjectRepresentation](#) ()  
*Clear a visual data of a data object.*

### Signals

- void [closed](#) ()

### Public Member Functions

- **DataObjectsManager** ([Core::Project](#) &project, QSettings &settings, QString &lastPath, QWidget \*parent=nullptr)
- void [selectDataObject](#) (int iRow)  
*Select a data object by row index.*
- mapDataObjects const & [getDataObjects](#) ()

## Private Member Functions

- void [closeEvent](#) (QCloseEvent \*pEvent) override  
*Save settings and delete handling widgets before closing the window.*
- void [createContent](#) ()  
*Create all the widgets.*
- ads::CDockWidget \* [createDataTableWidget](#) ()  
*Create a tabbed widget to interact with data tables.*
- ads::CDockWidget \* [createDataObjectsWidget](#) ()  
*Create an object to present all data objects.*
- ads::CDockWidget \* [createCodeWidget](#) ()  
*Create a widget enables to code data objects.*
- QLayout \* [createDialogControls](#) ()  
*Create dialog controls.*
- void [retrieveDataObjects](#) ()  
*Make a copy of existed data objects.*
- void [restoreSettings](#) ()  
*Restore settings from a file.*
- void [saveSettings](#) ()  
*Save settings to a file.*
- void [emplaceDataObject](#) (Core::AbstractDataObject \*pDataObject)  
*Helper function to insert data objects into the manager.*
- void [addListDataObjects](#) (Core::AbstractDataObject \*pDataObject)
- bool [isDataTableModifiable](#) ()  
*Helper function to check if it is possible to interact with data object content.*
- void [importDataObject](#) (QString const &path, QString const &fileName)  
*Import a data object from a file.*

## Private Attributes

- Ui::DataObjectsManager \* **mpUi**
- ads::CDockManager \* **mpDockManager**
- QTreeView \* **mpTreeDataObjects**
- QTreeView \* **mpDataTable**
- [DoubleSpinBoxItemDelegate](#) \* **mplItemDelegate** = nullptr
- [Core::Project](#) & **mProject**
- QSettings & **mSettings**
- mapDataObjects **mDataObjects**
- [Core::HierarchyTree](#) **mHierarchyDataObjects**
- QString & **mLastPath**
- [TableModels::TableModelInterface](#) \* **mpTableModelInterface** = nullptr
- [TableModels::BaseTableModel](#) \* **mpBaseTableModel**
- [TableModels::MatrixTableModel](#) \* **mpMatrixTableModel**
- [TableModels::SurfaceTableModel](#) \* **mpSurfaceTableModel**
- [HierarchyModels::DataObjectsHierarchyModel](#) \* **mpTreeDataObjectsModel**

### 4.8.1 Detailed Description

Manager to create objects of different types: scalars, vectors, matroces and surfaces.

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

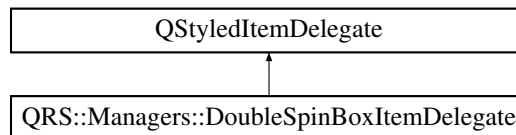
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/[dataobjectsmanager.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/[dataobjectsmanager.cpp](#)

## 4.9 QRS::Managers::DoubleSpinBoxItemDelegate Class Reference

Class to specify how table values can be edited.

```
#include <doublespinboxitemdelegate.h>
```

Inheritance diagram for QRS::Managers::DoubleSpinBoxItemDelegate:



### Public Member Functions

- **DoubleSpinBoxItemDelegate** (QObject \*parent=nullptr)
- QWidget \* **createEditor** (QWidget \*parent, const QStyleOptionViewItem &option, const QModelIndex &index) const override  
*Create a double value editor.*
- void **setEditorData** (QWidget \*pEditor, const QModelIndex &index) const override  
*Specify data to show.*
- void **setModelData** (QWidget \*pEditor, QAbstractItemModel \*pModel, const QModelIndex &index) const override  
*Set data to a model.*
- void **updateEditorGeometry** (QWidget \*pEditor, const QStyleOptionViewItem &option, const QModelIndex &index) const override  
*Set a geometry to render.*

### 4.9.1 Detailed Description

Class to specify how table values can be edited.

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

- /home/qinterfly/Library/Projects/QRodSystems/src/managers/[doublespinboxitemdelegate.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/[doublespinboxitemdelegate.cpp](#)

## 4.10 QRS::Core::HierarchyNode Class Reference

Hierarchy representative.

```
#include <hierarchynode.h>
```

### Public Types

- enum **NodeType** { **kObject** , **kDirectory** }

## Public Member Functions

- [HierarchyNode](#) (NodeType type, QVariant value)  
*Node constructor.*
- void [appendChild](#) ([HierarchyNode](#) \*node)  
*Add a child node.*
- bool [hasParent](#) () const
- bool [hasChild](#) () const
- bool [hasNextSibling](#) () const
- [HierarchyNode](#) \* [parent](#) ()
- [HierarchyNode](#) \* [firstChild](#) ()
- [HierarchyNode](#) \* [nextSibling](#) ()
- NodeType [type](#) () const
- QVariant & [value](#) ()
- [HierarchyNode](#) \* [groupNodes](#) ([HierarchyNode](#) \*pChildNode)  
*Merge two nodes into one entity.*
- bool [setBefore](#) ([HierarchyNode](#) \*pSetNode)  
*Set a given node before the current one.*
- bool [setAfter](#) ([HierarchyNode](#) \*pSetNode)  
*Set a given node after the current one.*

## Private Member Functions

- void [excludeNodeFromHierarchy](#) ()  
*Remove all links to the node.*
- bool [isSetAllowed](#) ([HierarchyNode](#) const \*pNode) const  
*Check whether it is possible to place a given item before or after the current one.*
- bool [isParentOf](#) ([HierarchyNode](#) const \*pNode) const  
*Check whether the current item contains a given node as a child.*

## Private Attributes

- [HierarchyNode](#) \* [mpParent](#) = nullptr
- [HierarchyNode](#) \* [mpFirstChild](#) = nullptr
- [HierarchyNode](#) \* [mpNextSibling](#) = nullptr
- [HierarchyNode](#) \* [mpPreviousSibling](#) = nullptr
- NodeType [mType](#)
- QVariant [mValue](#)

## Friends

- class [HierarchyTree](#)

### 4.10.1 Detailed Description

Hierarchy representative.

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

- /home/qinterfly/Library/Projects/QRodSystems/src/core/[hierarchynode.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[hierarchynode.cpp](#)

## 4.11 QRS::Core::HierarchyTree Class Reference

Hierarchy of data objects (n-array tree)

```
#include <hierarchytree.h>
```

### Public Member Functions

- [HierarchyTree](#) ()  
*Base tree constructor.*
- [HierarchyTree](#) ([HierarchyNode](#) \*pRootNode)  
*Take the user defined node as the root.*
- [HierarchyTree](#) (QDataStream &stream, int numNodes)  
*Read a tree from a stream.*
- [HierarchyTree](#) & [operator=](#) ([HierarchyTree](#) const &another)  
*Copy assignment operator.*
- [HierarchyTree](#) & [operator=](#) ([HierarchyTree](#) &&another)  
*Move assignment operator.*
- [~HierarchyTree](#) ()  
*Tree destructor.*
- void [clear](#) ()  
*Delete all nodes except the root node.*
- void [appendNode](#) ([HierarchyNode](#) \*pNode)  
*Append a node to the root node.*
- bool [removeNode](#) ([HierarchyNode::NodeType](#) type, QVariant const &value)  
*Remove a node by type and value.*
- void [removeNode](#) ([HierarchyNode](#) \*pNode)  
*Remove a node and all its subnodes.*
- void [changeNodeValue](#) ([HierarchyNode::NodeType](#) type, QVariant const &oldValue, QVariant const &newValue)  
*Change the value of a node.*
- [HierarchyNode](#) \* [root](#) ()
- [HierarchyTree](#) [clone](#) () const  
*Clone a tree.*
- [HierarchyNode](#) \* [findNode](#) ([HierarchyNode](#) \*pBaseNode, [HierarchyNode::NodeType](#) type, QVariant const &value) const  
*Find a node by type and value.*
- int [size](#) () const  
*Get a number of nodes.*

### Private Member Functions

- [HierarchyNode](#) \* [copyNode](#) ([HierarchyNode](#) \*pBaseNode, uint relativeLevel) const  
*Copy a node.*
- void [removeNodeSiblings](#) ([HierarchyNode](#) \*pNode)  
*Remove all subnodes.*
- void [printNode](#) (uint level, [HierarchyNode](#) \*pNode, QDebug stream) const  
*Print a current node and all its subnodes.*
- void [writeNode](#) ([HierarchyNode](#) \*pNode, QDataStream &stream) const  
*Print a current node and all its subnodes.*
- int [countNodes](#) ([HierarchyNode](#) \*pNode, int &numNodes) const  
*Count all nodes.*

## Private Attributes

- [HierarchyNode](#) \* **mpRootNode** = nullptr

## Friends

- QDebug [operator<<](#) (QDebug stream, [HierarchyTree](#) &tree)  
*Print a tree structure.*
- QDataStream & [operator<<](#) (QDataStream &stream, [HierarchyTree](#) const &tree)  
*Write a tree structure to a stream.*

### 4.11.1 Detailed Description

Hierarchy of data objects (n-array tree)

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

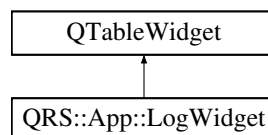
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[hierarchytree.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[hierarchytree.cpp](#)

## 4.12 QRS::App::LogWidget Class Reference

Log all the messages sent.

```
#include <logwidget.h>
```

Inheritance diagram for QRS::App::LogWidget:



## Public Member Functions

- **LogWidget** (QWidget \*parent=nullptr)
- void [log](#) (QtMsgType messageType, const QString &message)  
*Represent a message sent.*

### 4.12.1 Detailed Description

Log all the messages sent.

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

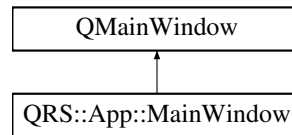
- /home/qinterfly/Library/Projects/QRodSystems/src/central/[logwidget.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/central/[logwidget.cpp](#)

## 4.13 QRS::App::MainWindow Class Reference

The main window of the program.

```
#include <mainwindow.h>
```

Inheritance diagram for QRS::App::MainWindow:



### Public Member Functions

- **MainWindow** (QWidget \*parent=nullptr)
- void [openProject](#) (QString const &filePath)  
*Open the specific project.*
- bool [saveProject](#) ()  
*Save the current project.*

### Static Public Attributes

- static [LogWidget](#) \* **pLogger** = nullptr

### Private Slots

- void [createProject](#) ()  
*Create a project and substitute the current one with it.*
- void [openProjectDialog](#) ()  
*Open a project by using a dialog.*
- void [openRecentProject](#) ()  
*Open the project which was selected from the Recent Projects menu.*
- bool [saveAsProject](#) ()  
*Save the current project under a new name.*
- void [projectModified](#) ()  
*Whenever a project has been modified.*
- void [saveSettings](#) ()  
*Save the current window settings.*
- void [restoreSettings](#) ()  
*Restore window settings from a file.*
- void [createDataObjectsManager](#) ()  
*Show a manager for designing data objects.*
- void [createRodPropertiesManager](#) ()  
*Show a manager to set rod properties based on the created data objects.*
- void [createRodConstructorManager](#) ()  
*Show a manager to create a rod with assigned data properties.*
- void [deleteDataObjectsManager](#) ()  
*Delete a manager of data objects after being used.*
- void [aboutProgram](#) ()  
*Show information about a program.*



## Private Member Functions

- void [initializeWindow](#) ()  
*Set a state and geometry of [MainWindow](#).*
- void [createContent](#) ()  
*Create all the widgets and corresponding actions.*
- void [closeEvent](#) (QCloseEvent \*pEvent) override  
*Save project and settings before exit.*
- ads::CDockWidget \* [createProjectHierarchyWidget](#) ()  
*Create a widget to represent a project hierarchy.*
- ads::CDockWidget \* [createGLWidget](#) ()  
*Create an OpenGL widget to render rods.*
- ads::CDockWidget \* [createLogWidget](#) ()  
*Create a window for logging.*
- ads::CDockWidget \* [createPropertiesWidget](#) ()  
*Create a window to modify properties of selected objects.*
- void [setProjectTitle](#) ()  
*Show information a name of a project.*
- void [retrieveRecentProjects](#) ()  
*Retrieve recent projects from the settings file.*
- void [addToRecentProjects](#) ()  
*Add the current project to the recent ones.*
- void [specifyMenuConnections](#) ()  
*Set signals and slots for menu actions.*
- bool [saveProjectChangesDialog](#) ()  
*Save project changes.*
- bool [saveProjectHelper](#) (QString const &filePath)  
*Helper method to perform saving of the current project.*

## Private Attributes

- Ui::MainWindow \* **mpUi**
- ads::CDockManager \* **mpDockManager**
- QLabel \* **mpStatusLabel**
- [Managers::DataObjectsManager](#) \* **mpDataObjectsManager** = nullptr
- [Core::Project](#) \* **mpProject**
- QSharedPointer< QSettings > **mpSettings**
- QString **mLastPath**
- QList< QString > **mPathRecentProjects**

### 4.13.1 Detailed Description

The main window of the program.

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

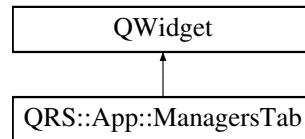
- /home/qinterfly/Library/Projects/QRodSystems/src/central/[mainwindow.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/central/[mainwindow.cpp](#)

## 4.14 QRS::App::ManagersTab Class Reference

A toolbar consisted of object designers.

```
#include <controltabs.h>
```

Inheritance diagram for QRS::App::ManagersTab:



### Signals

- void **actionDataObjectsTriggered** ()
- void **actionRodPropertiesTriggered** ()
- void **actionRodConstructorTriggered** ()

### Public Member Functions

- [ManagersTab](#) (QWidget \*parent=nullptr)  
*Managers tab constructor.*

#### 4.14.1 Detailed Description

A toolbar consisted of object designers.

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

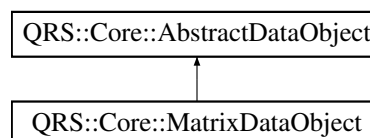
- /home/qinterfly/Library/Projects/QRodSystems/src/central/[controltabs.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/central/[controltabs.cpp](#)

## 4.15 QRS::Core::MatrixDataObject Class Reference

Matrix data object.

```
#include <matrixdataobject.h>
```

Inheritance diagram for QRS::Core::MatrixDataObject:



## Public Member Functions

- [MatrixDataObject](#) (QString const &name)  
*Construct a matrix data object.*
- [AbstractDataObject](#) \* [clone](#) () const override  
*Clone a matrix data object.*
- [DataItem](#) & [addItem](#) (DataValueType key) override  
*Insert a new item into [MatrixDataObject](#).*
- virtual void [import](#) (QTextStream &stream) override  
*Import a matrix data object from a file.*

## Static Public Member Functions

- static uint [numberInstances](#) ()
- static void [setNumberInstances](#) (uint numInstances)

## Static Private Attributes

- static uint [smNumInstances](#) = 0

## Additional Inherited Members

### 4.15.1 Detailed Description

Matrix data object.

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

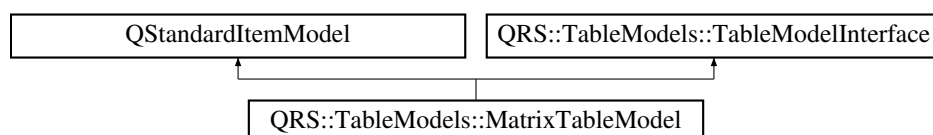
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[matrixdataobject.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[matrixdataobject.cpp](#)

## 4.16 QRS::TableModels::MatrixTableModel Class Reference

Table model to represent a matrix data object.

```
#include <matrixtablemodel.h>
```

Inheritance diagram for QRS::TableModels::MatrixTableModel:



## Public Member Functions

- **MatrixTableModel** (QWidget \*parent=nullptr)
- void **setDataObject** ([Core::AbstractDataObject](#) \*pDataObject)  
*Set a data object to represent.*
- bool **setData** (const QModelIndex &indexEdit, const QVariant &value, int role=Qt::EditRole) override  
*Set the data acquired from a delegate.*
- void **insertItemAfterSelected** (QItemSelectionModel \*pSelectionModel) override  
*Insert a new item after selected one.*
- void **insertLeadingItemAfterSelected** (QItemSelectionModel \*) override
- void **removeSelectedItem** (QItemSelectionModel \*pSelectionModel) override  
*Remove an array under selection.*
- void **removeSelectedLeadingItem** (QItemSelectionModel \*) override

## Private Member Functions

- void **updateContent** ()  
*Represent all items which a vector data object contains.*
- void **clearContent** ()  
*Clear previously created items.*

## Private Attributes

- [Core::AbstractDataObject](#) \* **mpDataObject** = nullptr

## Additional Inherited Members

### 4.16.1 Detailed Description

Table model to represent a matrix data object.

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

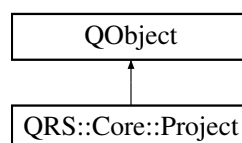
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/[matrixtablemodel.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/[matrixtablemodel.cpp](#)

## 4.17 QRS::Core::Project Class Reference

[Project](#) class to interact with a created system of rods.

```
#include <project.h>
```

Inheritance diagram for QRS::Core::Project:



## Public Slots

- bool [save](#) (QString const &dir, QString const &fileName)  
*Save a project to a file.*

## Signals

- void **dataObjectAdded** (QRS::Core::DataIDType id)
- void **dataObjectRemoved** (QRS::Core::DataIDType id)
- void **allDataObjectsChanged** ()
- void **modified** (bool modifiedState)

## Public Member Functions

- [Project](#) (QString const &name)  
*Construct a clean project with the user specified name.*
- [Project](#) (QString const &path, QString const &fileName)  
*Read a project from a file.*
- bool **isModified** () const
- DataIDType **numberDataObjects** () const
- std::unordered\_map< DataIDType, [AbstractDataObject](#) \* > [cloneDataObjects](#) () const  
*Clone data objects.*
- DataIDType [addDataObject](#) (DataObjectType type)  
*Create a data object with the specified type.*
- void [removeDataObject](#) (DataIDType id)  
*Remove a data object by id.*
- void [setDataObjects](#) (std::unordered\_map< DataIDType, [AbstractDataObject](#) \* > dataObjects, [HierarchyTree](#) const &hierarchyDataObjects)  
*Substitute current data objects with new ones.*
- [HierarchyTree](#) [cloneHierarchyDataObjects](#) () const  
*Clone a hierarchy of data objects.*
- QString const & **name** () const
- QString const & **filePath** () const
- void [importDataObjects](#) (QString const &path, QString const &fileName)  
*Import several data objects from a file.*

## Static Public Member Functions

- static QString const & **getFileExtension** ()

## Private Slots

- void [setModified](#) (bool modifiedState=true)  
*Set a modification state.*

## Private Attributes

- quint32 [mID](#)  
*Unique project identifier.*
- QString [mName](#)  
*Project name.*
- QString [mFilePath](#)  
*Path to a file where a project is stored.*
- bool [mIsModified](#)  
*Flag whether a project has been modified since last saving.*
- DataObjects [mDataObjects](#)  
*Data objects.*
- [HierarchyTree](#) [mHierarchyDataObjects](#)  
*Hierarchy of data objects.*

## Static Private Attributes

- static const QString [skProjectExtension](#) = ".qrs"  
*File extensionn.*

## Friends

- class **QRS::HierarchyModels::ProjectHierarchyModel**

### 4.17.1 Detailed Description

[Project](#) class to interact with a created system of rods.

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

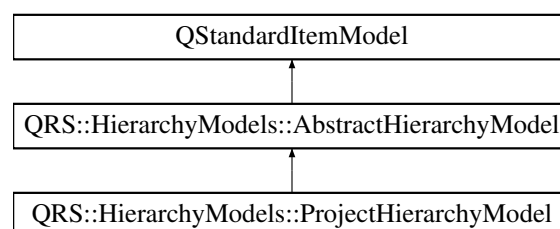
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[project.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[project.cpp](#)

## 4.18 QRS::HierarchyModels::ProjectHierarchyModel Class Reference

Project hierarchy representative.

```
#include <projecthierarchymodel.h>
```

Inheritance diagram for QRS::HierarchyModels::ProjectHierarchyModel:



## Public Member Functions

- **ProjectHierarchyModel** ([Core::Project](#) &project, QTreeView \*pView=nullptr)
- void [updateContent](#) () override  
*Update all the content.*
- void [clearContent](#) () override  
*Clear all the items.*

## Private Attributes

- Core::DataObjects & **mDataObjects**
- [Core::HierarchyTree](#) & **mHierarchyDataObjects**

## Additional Inherited Members

### 4.18.1 Detailed Description

Project hierarchy representative.

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

- /home/qinterfly/Library/Projects/QRodSystems/src/central/[projecthierarchymodel.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/central/[projecthierarchymodel.cpp](#)

## 4.19 QRS::Core::Array< T >::Row< U > Struct Template Reference

Proxy class to acquire a row by index.

## Public Member Functions

- **Row** (T \*pData)
- T & **operator[]** (IndexType iCol)

## Public Attributes

- T \* **pRow**

### 4.19.1 Detailed Description

```
template<typename T>
template<typename U>
struct QRS::Core::Array< T >::Row< U >
```

Proxy class to acquire a row by index.

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

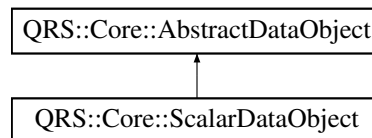
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[array.h](#)

## 4.20 QRS::Core::ScalarDataObject Class Reference

Scalar data object.

```
#include <scalardataobject.h>
```

Inheritance diagram for QRS::Core::ScalarDataObject:



### Public Member Functions

- [ScalarDataObject](#) (QString const &name)  
*Construct a scalar data object.*
- [AbstractDataObject \\* clone](#) () const override  
*Clone a scalar data object.*
- [DataItemType](#) & [addItem](#) (DataValueType key) override  
*Insert a new item into [ScalarDataObject](#).*
- virtual void [import](#) (QTextStream &stream) override  
*Import a scalar data object from a file.*

### Static Public Member Functions

- static uint [numberInstances](#) ()
- static void [setNumberInstances](#) (uint numInstances)

### Static Private Attributes

- static uint [smNumInstances](#) = 0

### Additional Inherited Members

#### 4.20.1 Detailed Description

Scalar data object.

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

- /home/qinterfly/Library/Projects/QRodSystems/src/core/[scalardataobject.h](#)
- /home/qinterfly/Library/Projects/QRodSystems/src/core/[scalardataobject.cpp](#)

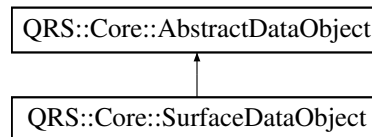


## 4.21 QRS::Core::SurfaceDataObject Class Reference

Surface data object.

```
#include <surfacedataobject.h>
```

Inheritance diagram for QRS::Core::SurfaceDataObject:



### Public Member Functions

- [SurfaceDataObject](#) (QString const &name)  
*Construct a surface data object.*
- [AbstractDataObject](#) \* [clone](#) () const override  
*Clone a surface data object.*
- [DataItemType](#) & [addItem](#) (DataValueType key) override  
*Insert a new item into [SurfaceDataObject](#).*
- DataKeyType [addLeadingItem](#) (DataValueType key)  
*Add a leading item.*
- void [removeLeadingItem](#) (DataValueType key)  
*Remove a leading item.*
- bool [changeLeadingItemKey](#) (DataKeyType oldKey, DataKeyType newKey)  
*Modify a leading item key.*
- DataHolder & [getLeadingItems](#) ()
- void [serialize](#) (QDataStream &stream) const override  
*Serialize additional data of a surface object.*
- virtual void [deserialize](#) (QDataStream &stream) override  
*Deserialize additional data of a surface object.*
- virtual void [import](#) (QTextStream &stream) override  
*Import a surface data object from a file.*

### Static Public Member Functions

- static uint [numberInstances](#) ()
- static void [setNumberInstances](#) (uint numInstances)

### Private Attributes

- DataHolder [mLeadingItems](#)

### Static Private Attributes

- static uint [smNumInstances](#) = 0

## Additional Inherited Members

### 4.21.1 Detailed Description

Surface data object.

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

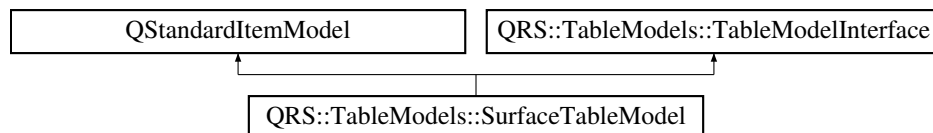
- [/home/qinterfly/Library/Projects/QRodSystems/src/core/surfacedataobject.h](#)
- [/home/qinterfly/Library/Projects/QRodSystems/src/core/surfacedataobject.cpp](#)

## 4.22 QRS::TableModels::SurfaceTableModel Class Reference

Table model to represent a surface data object.

```
#include <surfacetablemodel.h>
```

Inheritance diagram for QRS::TableModels::SurfaceTableModel:



## Public Member Functions

- **SurfaceTableModel** (QWidget \*parent=nullptr)
- void **setDataObject** ([Core::SurfaceDataObject](#) \*pDataObject)  
*Set a surface data object to represent.*
- bool **setData** (const QModelIndex &indexEdit, const QVariant &value, int role=Qt::EditRole) override  
*Set the data acquired from a delegate.*
- void **insertItemAfterSelected** (QItemSelectionModel \*pSelectionModel) override  
*Insert a new item after selected one.*
- void **removeSelectedItem** (QItemSelectionModel \*pSelectionModel) override  
*Remove an array under selection.*
- void **insertLeadingItemAfterSelected** (QItemSelectionModel \*pSelectionModel) override  
*Add a new leading item after selected one.*
- void **removeSelectedLeadingItem** (QItemSelectionModel \*pSelectionModel) override  
*Remove a selected leading item.*

## Private Member Functions

- void **updateContent** ()  
*Represent all items which a data object contains.*
- void **clearContent** ()  
*Clear previously created items.*

## Private Attributes

- `Core::SurfaceDataObject * mpDataObject = nullptr`

## Additional Inherited Members

### 4.22.1 Detailed Description

Table model to represent a surface data object.

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

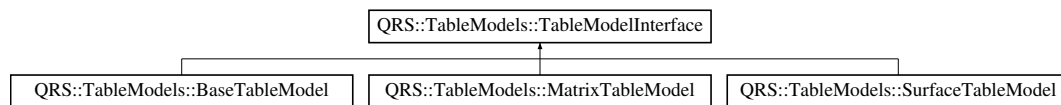
- `/home/qinterfly/Library/Projects/QRodSystems/src/managers/surfacetablemodel.h`
- `/home/qinterfly/Library/Projects/QRodSystems/src/managers/surfacetablemodel.cpp`

## 4.23 QRS::TableModels::TableModelInterface Class Reference

User interface to add and remove items.

```
#include <tablemodelinterface.h>
```

Inheritance diagram for QRS::TableModels::TableModelInterface:



## Public Member Functions

- virtual void **insertItemAfterSelected** (QItemSelectionModel \*pSelectionModel)=0
- virtual void **insertLeadingItemAfterSelected** (QItemSelectionModel \*pSelectionModel)=0
- virtual void **removeSelectedItem** (QItemSelectionModel \*pSelectionModel)=0
- virtual void **removeSelectedItemLeadingItem** (QItemSelectionModel \*pSelectionModel)=0

## Static Public Member Functions

- static QStandardItem \* **makeDoubleItem** (double value)  
*Helper function to make an item which holds a double value.*
- static QList< QStandardItem \* > **prepareRow** (Core::Array< double > &array, uint iRow)  
*Helper function to copy a row from an array.*
- static QList< QStandardItem \* > **prepareRow** (double const &key, Core::Array< double > &array, uint iRow)  
*Helper function to copy a row from an array and associate it with a key.*
- static QList< QStandardItem \* > **prepareRow** (QString const &name, Core::Array< double > &array, uint iRow)  
*Helper function to copy a row from an array and associate it with a name.*
- static QStandardItem \* **makeLabelItem** (QString const &name)  
*Helper function to create an item which holds a string and cannot be modified.*

### 4.23.1 Detailed Description

User interface to add and remove items.

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

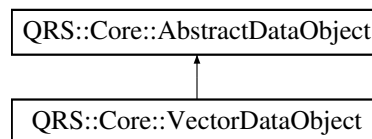
- [/home/qinterfly/Library/Projects/QRodSystems/src/managers/tablemodelinterface.h](#)
- [/home/qinterfly/Library/Projects/QRodSystems/src/managers/tablemodelinterface.cpp](#)

## 4.24 QRS::Core::VectorDataObject Class Reference

Vector data object.

```
#include <vectordataobject.h>
```

Inheritance diagram for QRS::Core::VectorDataObject:



### Public Member Functions

- [VectorDataObject](#) (QString const &name)  
*Construct a vector data object.*
- [AbstractDataObject \\* clone](#) () const override  
*Clone a vector data object.*
- [DataItemType](#) & [addItem](#) (DataValueType key) override  
*Insert a new item into [VectorDataObject](#).*
- virtual void [import](#) (QTextStream &stream) override  
*Import a vector data object from a file.*

### Static Public Member Functions

- static uint **numberInstances** ()
- static void **setNumberInstances** (uint numInstances)

### Static Private Attributes

- static uint **smNumInstances** = 0

## Additional Inherited Members

### 4.24.1 Detailed Description

Vector data object.

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

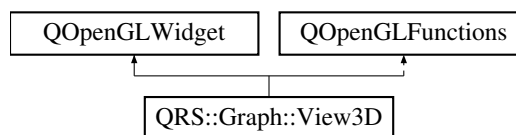
- [/home/qinterfly/Library/Projects/QRodSystems/src/core/vectordataobject.h](#)
- [/home/qinterfly/Library/Projects/QRodSystems/src/core/vectordataobject.cpp](#)

## 4.25 QRS::Graph::View3D Class Reference

A widget to represent the resulted rod system.

```
#include <view3d.h>
```

Inheritance diagram for QRS::Graph::View3D:



### Public Member Functions

- **View3D** (QWidget \*parent=nullptr)

### Protected Member Functions

- void [initializeGL](#) () override  
*Initialize a graphical scene.*
- void [paintGL](#) () override  
*Render its content.*

### Private Attributes

- bool **mCore**

### 4.25.1 Detailed Description

A widget to represent the resulted rod system.

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

- [/home/qinterfly/Library/Projects/QRodSystems/src/render/view3d.h](#)
- [/home/qinterfly/Library/Projects/QRodSystems/src/render/view3d.cpp](#)



## Chapter 5

# File Documentation

### 5.1 `/home/qinterfly/Library/Projects/QRod↔ Systems/src/central/controltabs.cpp` File Reference

Implementation of the ControlTabs class.

```
#include <QLayout>
#include <QToolBar>
#include <QIcon>
#include "controltabs.h"
```

#### 5.1.1 Detailed Description

Implementation of the ControlTabs class.

Author

Pavel Lakiza

Date

March 2021

### 5.2 `/home/qinterfly/Library/Projects/QRod↔ Systems/src/central/controltabs.h` File Reference

Declaration of the ControlTabs class.

```
#include <QWidget>
```

## Classes

- class [QRS::App::ManagersTab](#)  
*A toolbar consisted of object designers.*

### 5.2.1 Detailed Description

Declaration of the ControlTabs class.

#### Author

Pavel Lakiza

#### Date

March 2021

## 5.3 [/home/qinterfly/Library/Projects/QRod↔](#) Systems/src/central/logwidget.cpp File Reference

Implementation of the LogWidget class.

```
#include <QHeaderView>
#include <QTime>
#include <QTimer>
#include "logwidget.h"
```

## Enumerations

- enum **ColumnType** { kTime , kType , kMessage }

### 5.3.1 Detailed Description

Implementation of the LogWidget class.

#### Author

Pavel Lakiza

#### Date

May 2021



## 5.4 /home/qinterfly/Library/Projects/QRodSystems/src/central/logwidget.h File Reference

Declaration of the LogWidget class.

```
#include <QTableWidget>
```

### Classes

- class [QRS::App::LogWidget](#)  
*Log all the messages sent.*

#### 5.4.1 Detailed Description

Declaration of the LogWidget class.

##### Author

Pavel Lakiza

##### Date

May 2021

## 5.5 /home/qinterfly/Library/Projects/QRodSystems/src/central/mainwindow.cpp File Reference

Implementation of the MainWindow class.

```
#include <QDesktopWidget>
#include <QToolBar>
#include <QTableWidget>
#include <QTreeView>
#include <QTextEdit>
#include <QVBoxLayout>
#include <QSettings>
#include <QMessageBox>
#include <QFileDialog>
#include <QLabel>
#include "DockManager.h"
#include "DockWidget.h"
#include "DockAreaWidget.h"
#include "ads_globals.h"
#include "mainwindow.h"
#include "ui_mainwindow.h"
#include "controltabs.h"
#include "logwidget.h"
#include "uiconstants.h"
#include "projecthierarchymodel.h"
#include "managers/dataobjectsmanager.h"
#include "render/view3d.h"
```

## Functions

- void [moveToCenter](#) (QWidget \*pWidget)  
*Helper function to situate widgets at the center of their parent widgets.*

### 5.5.1 Detailed Description

Implementation of the MainWindow class.

#### Author

Pavel Lakiza

#### Date

May 2021

## 5.6 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/central/mainwindow.h File Reference

Declaration of the MainWindow class.

```
#include <QMainWindow>
#include "logwidget.h"
#include "../core/project.h"
```

## Classes

- class [QRS::App::MainWindow](#)  
*The main window of the program.*

## Functions

- void [QRS::App::throwMessage](#) (QtMsgType type, const QMessageLogContext &, const QString &message)  
*Log all the messages.*

### 5.6.1 Detailed Description

Declaration of the MainWindow class.

#### Author

Pavel Lakiza

#### Date

May 2021

## 5.7 /home/qinterfly/Library/Projects/QRod↵ Systems/src/central/projecthierarchymodel.cpp File Reference

Definition of the ProjectHierarchyModel class.

```
#include <QTreeView>
#include "projecthierarchymodel.h"
```

### 5.7.1 Detailed Description

Definition of the ProjectHierarchyModel class.

Author

Pavel Lakiza

Date

May 2021

## 5.8 /home/qinterfly/Library/Projects/QRod↵ Systems/src/central/projecthierarchymodel.h File Reference

Declaration of the ProjectHierarchyModel class.

```
#include "models/abstrachhierarchymodel.h"
#include "core/datatypes.h"
#include "core/project.h"
```

### Classes

- class [QRS::HierarchyModels::ProjectHierarchyModel](#)  
*Project hierarchy representative.*

### 5.8.1 Detailed Description

Declaration of the ProjectHierarchyModel class.

Author

Pavel Lakiza

Date

May 2021

## 5.9 /home/qinterfly/Library/Projects/QRod↵ Systems/src/central/uiconstants.h File Reference

Common graphical constants shared between several windows.

```
#include <QString>
```

### Variables

- const QString **QRS::UiConstants::Settings::skGeometry** = "geometry"
- const QString **QRS::UiConstants::Settings::skState** = "state"
- const QString **QRS::UiConstants::Settings::skDockingState** = "dockingState"

### 5.9.1 Detailed Description

Common graphical constants shared between several windows.

#### Author

Pavel Lakiza

#### Date

April 2021

## 5.10 /home/qinterfly/Library/Projects/QRod↵ Systems/src/core/abstractdataobject.cpp File Reference

Implementation of the AbstractDataObject class.

```
#include "abstractdataobject.h"
```

### 5.10.1 Detailed Description

Implementation of the AbstractDataObject class.

#### Author

Pavel Lakiza

#### Date

April 2021

## 5.11 /home/qinterfly/Library/Projects/QRodSystems/src/core/abstractdataobject.h File Reference

Declaration of the AbstractDataObject class.

```
#include <QString>
#include <QDataStream>
#include <unordered_map>
#include "array.h"
#include "datatypes.h"
```

### Classes

- class [QRS::Core::AbstractDataObject](#)  
*Data object which is designed in the way to be represented in a table easily.*

### Typedefs

- using [QRS::Core::DataItem](#) = Array< DataValueType >
- using [QRS::Core::DataHolder](#) = std::map< DataKeyType, DataItem >

### Functions

- QDataStream & [QRS::Core::operator<<](#) (QDataStream &stream, AbstractDataObject const &obj)  
*Print a data object to a stream.*
- QDataStream & [QRS::Core::operator>>](#) (QDataStream &stream, AbstractDataObject &obj)  
*Read a data object from a stream.*

#### 5.11.1 Detailed Description

Declaration of the AbstractDataObject class.

##### Author

Pavel Lakiza

##### Date

April 2021

## 5.12 /home/qinterfly/Library/Projects/QRodSystems/src/core/array.cpp File Reference

Implementation of the Array class.

```
#include "array.h"
```

### 5.12.1 Detailed Description

Implementation of the Array class.

Author

Pavel Lakiza

Date

March 2021

## 5.13 /home/qinterfly/Library/Projects/QRodSystems/src/core/array.h File Reference

Declaration of the Array class.

```
#include <QDebug>
```

### Classes

- class [QRS::Core::Array< T >](#)  
*Numerical array class.*
- struct [QRS::Core::Array< T >::Row< U >](#)  
*Proxy class to acquire a row by index.*

### Typedefs

- using [QRS::Core::IndexType](#) = unsigned int

### Functions

- template<typename K >  
QDebug [QRS::Core::operator<<](#) (QDebug stream, Array< K > &array)  
*Print all array values using the matrix format.*
- template<typename K >  
QDataStream & [QRS::Core::operator<<](#) (QDataStream &stream, Array< K > const &array)  
*Write an array to a stream.*
- template<typename K >  
QDataStream & [QRS::Core::operator>>](#) (QDataStream &stream, Array< K > &array)  
*Read an array from a stream.*

### 5.13.1 Detailed Description

Declaration of the Array class.

Author

Pavel Lakiza

Date

March 2021

## 5.14 /home/qinterfly/Library/Projects/QRodSystems/src/core/datatypes.h File Reference

Specification of data types used in a project.

### Typedefs

- using **QRS::Core::DataValueType** = double
- using **QRS::Core::DataKeyType** = double
- using **QRS::Core::DataIDType** = unsigned int

### Enumerations

- enum **DataObjectType** { **kScalar** , **kVector** , **kMatrix** , **kSurface** }

### 5.14.1 Detailed Description

Specification of data types used in a project.

Author

Pavel Lakiza

Date

March 2021

## 5.15 /home/qinterfly/Library/Projects/QRodSystems/src/core/hierarchynode.cpp File Reference

Implementation of the HierarchyNode class.

```
#include "hierarchynode.h"
```

### 5.15.1 Detailed Description

Implementation of the HierarchyNode class.

Author

Pavel Lakiza

Date

May 2021

## 5.16 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/core/hierarchynode.h File Reference

Declaration of the HierarchyNode class.

```
#include <QVariant>
#include <QDataStream>
#include <QSharedPointer>
```

### Classes

- class [QRS::Core::HierarchyNode](#)  
*Hierarchy representative.*

### 5.16.1 Detailed Description

Declaration of the HierarchyNode class.

Author

Pavel Lakiza

Date

May 2021

## 5.17 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/core/hierarchytree.cpp File Reference

Implementation of the HierarchyTree class.

```
#include "hierarchytree.h"
```



### 5.17.1 Detailed Description

Implementation of the HierarchyTree class.

Author

Pavel Lakiza

Date

May 2021

## 5.18 /home/qinterfly/Library/Projects/QRodSystems/src/core/hierarchytree.h File Reference

Declaration of the HierarchyTree class.

```
#include <QDebug>
#include "hierarchynode.h"
```

### Classes

- class [QRS::Core::HierarchyTree](#)  
*Hierarchy of data objects (n-array tree)*

### Functions

- QDebug [QRS::Core::operator<<](#) (QDebug stream, HierarchyTree &tree)  
*Print a tree structure.*
- QDataStream & [QRS::Core::operator<<](#) (QDataStream &stream, HierarchyTree const &tree)  
*Write a tree structure to a stream.*

### 5.18.1 Detailed Description

Declaration of the HierarchyTree class.

Author

Pavel Lakiza

Date

April 2021

## 5.19 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/core/matrixdataobject.cpp File Reference

Implementation of the MatrixDataObject class.

```
#include "matrixdataobject.h"
```

### Variables

- const IndexType **skNumElements** = 3

#### 5.19.1 Detailed Description

Implementation of the MatrixDataObject class.

Author

Pavel Lakiza

Date

April 2021

## 5.20 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/core/matrixdataobject.h File Reference

Declaration of the MatrixDataObject class.

```
#include "abstractdataobject.h"
```

### Classes

- class [QRS::Core::MatrixDataObject](#)  
*Matrix data object.*

#### 5.20.1 Detailed Description

Declaration of the MatrixDataObject class.

Author

Pavel Lakiza

Date

April 2021

## 5.21 /home/qinterfly/Library/Projects/QRodSystems/src/core/project.cpp File Reference

Implementation of the Project class.

```
#include <QDebug>
#include <QRandomGenerator>
#include <QFileInfo>
#include <QDir>
#include <QDataStream>
#include <QDateTime>
#include "project.h"
#include "scalarsdataobject.h"
#include "vectordataobject.h"
#include "matrixdataobject.h"
#include "surfacedataobject.h"
#include "utilities.h"
```

### Functions

- [AbstractDataObject \\* createDataObject](#) (DataObjectType type)  
*Helper function to create DataObject instance by a type and name.*

#### 5.21.1 Detailed Description

Implementation of the Project class.

##### Author

Pavel Lakiza

##### Date

May 2021

## 5.22 /home/qinterfly/Library/Projects/QRodSystems/src/core/project.h File Reference

Declaration of the Project class.

```
#include <QObject>
#include <unordered_map>
#include <memory>
#include "abstractdataobject.h"
#include "array.h"
#include "hierarchytree.h"
```

## Classes

- class [QRS::Core::Project](#)  
*Project class to interact with a created system of rods.*

## Typedefs

- using **QRS::Core::DataObjects** = std::unordered\_map< DataIDType, std::shared\_ptr< AbstractDataObject  
> >

### 5.22.1 Detailed Description

Declaration of the Project class.

#### Author

Pavel Lakiza

#### Date

May 2021

## 5.23 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/core/scalardataobject.cpp File Reference

Implementation of the ScalarDataObject class.

```
#include "scalardataobject.h"
```

### 5.23.1 Detailed Description

Implementation of the ScalarDataObject class.

#### Author

Pavel Lakiza

#### Date

April 2021

## 5.24 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/core/scalardataobject.h File Reference

Declaration of the ScalarDataObject class.

```
#include "abstractdataobject.h"
```

## Classes

- class [QRS::Core::ScalarDataObject](#)  
*Scalar data object.*

### 5.24.1 Detailed Description

Declaration of the ScalarDataObject class.

#### Author

Pavel Lakiza

#### Date

April 2021

## 5.25 /home/qinterfly/Library/Projects/QRod↵ Systems/src/core/surfacedataobject.cpp File Reference

Implementation of the SurfaceDataObject class.

```
#include "surfacedataobject.h"
```

### 5.25.1 Detailed Description

Implementation of the SurfaceDataObject class.

#### Author

Pavel Lakiza

#### Date

April 2021

## 5.26 /home/qinterfly/Library/Projects/QRod↵ Systems/src/core/surfacedataobject.h File Reference

Declaration of the SurfaceDataObject class.

```
#include "abstractdataobject.h"
```

## Classes

- class [QRS::Core::SurfaceDataObject](#)  
*Surface data object.*

### 5.26.1 Detailed Description

Declaration of the SurfaceDataObject class.

#### Author

Pavel Lakiza

#### Date

April 2021

## 5.27 /home/qinterfly/Library/Projects/QRodSystems/src/core/utilities.cpp File Reference

Implementation of utilities.

```
#include <QDebug>
#include <QString>
#include <QFile>
#include <QDir>
#include <QPair>
#include "utilities.h"
```

### 5.27.1 Detailed Description

Implementation of utilities.

#### Author

Pavel Lakiza

#### Date

May 2021

## 5.28 /home/qinterfly/Library/Projects/QRodSystems/src/core/utilities.h File Reference

Declaration of utilities.

```
#include <QSharedPointer>
#include "datatypes.h"
```

## Functions

- QPair< Core::DataObjectType, QSharedPointer< QFile > > [QRS::Utilities::File::getDataObjectFile](#) (QString const &path, QString const &fileName)  
*Retrieve a pair consisted of a data object file and its type.*
- QString [QRS::Utilities::File::loadFileContent](#) (QString const &path)  
*Load a style sheet.*

### 5.28.1 Detailed Description

Declaration of utilities.

Author

Pavel Lakiza

Date

May 2021

## 5.29 /home/qinterfly/Library/Projects/QRod↵ Systems/src/core/vectordataobject.cpp File Reference

Implementation of the VectorDataObject class.

```
#include "vectordataobject.h"
```

## Variables

- const IndexType **skNumElements** = 3

### 5.29.1 Detailed Description

Implementation of the VectorDataObject class.

Author

Pavel Lakiza

Date

April 2021

## 5.30 /home/qinterfly/Library/Projects/QRod↩ Systems/src/core/vectordataobject.h File Reference

Declaration of the VectorDataObject class.

```
#include "abstractdataobject.h"
```

### Classes

- class [QRS::Core::VectorDataObject](#)  
*Vector data object.*

#### 5.30.1 Detailed Description

Declaration of the VectorDataObject class.

##### Author

Pavel Lakiza

##### Date

April 2021

## 5.31 /home/qinterfly/Library/Projects/QRodSystems/src/main/main.cpp File Reference

The startup function.

```
#include <QFile>
#include <QApplication>
#include "mainwindow.h"
#include "utilities.h"
```

### Functions

- int [main](#) (int argc, char \*argv[])  
*Entry point.*

#### 5.31.1 Detailed Description

The startup function.

##### Author

Pavel Lakiza

##### Date

May 2021



## 5.32 /home/qinterfly/Library/Projects/QRod↵ Systems/src/managers/basetablemodel.cpp File Reference

Implementation of the BaseTableModel class.

```
#include <QTreeView>
#include "basetablemodel.h"
#include "abstractdataobject.h"
```

### 5.32.1 Detailed Description

Implementation of the BaseTableModel class.

**Author**

Pavel Lakiza

**Date**

March 2021

## 5.33 /home/qinterfly/Library/Projects/QRod↵ Systems/src/managers/basetablemodel.h File Reference

Declaration of the BaseTableModel class.

```
#include <QStandardItemModel>
#include "tablemodelinterface.h"
```

### Classes

- class [QRS::TableModels::BaseTableModel](#)  
*Table model to represent either a scalar or vector data object.*

### 5.33.1 Detailed Description

Declaration of the BaseTableModel class.

**Author**

Pavel Lakiza

**Date**

March 2021

## 5.34 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/managers/dataobjectshierarchyitem.cpp File Reference

Definition of the DataObjectsHierarchyItem class.

```
#include "dataobjectshierarchyitem.h"
#include "core/abstractdataobject.h"
#include "core/hierarchytree.h"
```

### Functions

- QIcon [getDataObjectIcon](#) (DataObjectType type)  
*Helper function to assign appropriate data object icon.*

#### 5.34.1 Detailed Description

Definition of the DataObjectsHierarchyItem class.

##### Author

Pavel Lakiza

##### Date

May 2021

## 5.35 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/managers/dataobjectshierarchyitem.h File Reference

Declaration of the DataObjectsHierarchyItem class.

```
#include <unordered_map>
#include "models/abstrachierarchyitem.h"
#include "core/datatypes.h"
```

### Classes

- class [QRS::HierarchyModels::DataObjectsHierarchyItem](#)  
*Item to represent a hierarchy of data objects.*

### Typedefs

- using [QRS::mapDataObjects](#) = std::unordered\_map< Core::DataIDType, Core::AbstractDataObject \* >

### 5.35.1 Detailed Description

Declaration of the DataObjectsHierarchyItem class.

Author

Pavel Lakiza

Date

May 2021

## 5.36 /home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectshierarchymodel.cpp File Reference

Definition of the DataObjectsHierarchyModel class.

```
#include <QTreeView>
#include <QMimeData>
#include "dataobjectshierarchymodel.h"
#include "dataobjectshierarchyitem.h"
#include "core/abstractdataobject.h"
#include "core/hierarchytree.h"
```

### 5.36.1 Detailed Description

Definition of the DataObjectsHierarchyModel class.

Author

Pavel Lakiza

Date

May 2021

## 5.37 /home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectshierarchymodel.h File Reference

Declaration of the DataObjectsHierarchyModel class.

```
#include "models/abstrachhierarchymodel.h"
#include "dataobjectshierarchyitem.h"
```

## Classes

- class [QRS::HierarchyModels::DataObjectsHierarchyModel](#)  
*Tree model to represent and modify a hierarchy of data objects.*

### 5.37.1 Detailed Description

Declaration of the DataObjectsHierarchyModel class.

#### Author

Pavel Lakiza

#### Date

May 2021

## 5.38 [/home/qinterfly/Library/Projects/QRod](#)↵ Systems/src/managers/dataobjectsmanager.cpp File Reference

Implementation of the DataObjectsManager class.

```
#include <QTreeView>
#include <QSettings>
#include <QHBoxLayout>
#include <QToolBar>
#include <QListWidget>
#include <QTextEdit>
#include <QPushButton>
#include <QSpacerItem>
#include <QMessageBox>
#include <QShortcut>
#include <QFileDialog>
#include "DockManager.h"
#include "DockWidget.h"
#include "DockAreaWidget.h"
#include "dataobjectsmanager.h"
#include "ui_dataobjectsmanager.h"
#include "central/uiconstants.h"
#include "core/project.h"
#include "core/scalardataobject.h"
#include "core/vectordataobject.h"
#include "core/matrixdataobject.h"
#include "core/surfacedataobject.h"
#include "core/utilities.h"
#include "basetablemodel.h"
#include "matrixtablemodel.h"
#include "surfacetablemodel.h"
#include "doublespinboxitemdelegate.h"
#include "dataobjectshierarchymodel.h"
```

## Functions

- void [setToolBarShortcutHints](#) (QToolBar \*pToolBar)  
*Helper function to add a shortcut hint to all actions which a toolbar contains.*
- QIcon [getDataObjectIcon](#) (DataObjectType type)  
*Helper function to assign appropriate data object icon.*

## Variables

- const QString **skDataObjectsWindow** = "DataObjectsManager"

### 5.38.1 Detailed Description

Implementation of the DataObjectsManager class.

#### Author

Pavel Lakiza

#### Date

March 2021

## 5.39 /home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectsmanager.h File Reference

Declaration of the DataObjectsManager class.

```
#include <QSignalMapper>
#include <QDialog>
#include <unordered_map>
#include "../core/datatypes.h"
#include "../core/hierarchytree.h"
```

## Classes

- class [QRS::Managers::DataObjectsManager](#)  
*Manager to create objects of different types: scalars, vectors, matroces and surfaces.*

## Typedefs

- using **QRS::Managers::mapDataObjects** = std::unordered\_map< Core::DataIDType, Core::AbstractDataObject \* >

### 5.39.1 Detailed Description

Declaration of the DataObjectsManager class.

#### Author

Pavel Lakiza

#### Date

March 2021

## 5.40 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/managers/doublespinboxitemdelegate.cpp File Reference

Implementation of the DoubleSpinBoxItemDelegate class.

```
#include <QDoubleSpinBox>
#include "doublespinboxitemdelegate.h"
```

### 5.40.1 Detailed Description

Implementation of the DoubleSpinBoxItemDelegate class.

#### Author

Pavel Lakiza

#### Date

March 2021

## 5.41 [/home/qinterfly/Library/Projects/QRod](#) Systems/src/managers/doublespinboxitemdelegate.h File Reference

Declaration of the DoubleSpinBoxItemDelegate class.

```
#include <QStyledItemDelegate>
```

### Classes

- class [QRS::Managers::DoubleSpinBoxItemDelegate](#)

*Class to specify how table values can be edited.*

### 5.41.1 Detailed Description

Declaration of the DoubleSpinBoxItemDelegate class.

#### Author

Pavel Lakiza

#### Date

March 2021

## 5.42 /home/qinterfly/Library/Projects/QRod↵ Systems/src/managers/matrixtablemodel.cpp File Reference

Implementation of the MatrixTableModel class.

```
#include <QTreeView>
#include "core/abstractdataobject.h"
#include "matrixtablemodel.h"
```

### 5.42.1 Detailed Description

Implementation of the MatrixTableModel class.

#### Author

Pavel Lakiza

#### Date

March 2021

## 5.43 /home/qinterfly/Library/Projects/QRod↵ Systems/src/managers/matrixtablemodel.h File Reference

Declaration of the MatrixTableModel class.

```
#include <QStandardItemModel>
#include "tablemodelinterface.h"
```

### Classes

- class [QRS::TableModels::MatrixTableModel](#)  
*Table model to represent a matrix data object.*

### 5.43.1 Detailed Description

Declaration of the MatrixTableModel class.

#### Author

Pavel Lakiza

#### Date

March 2021

## 5.44 /home/qinterfly/Library/Projects/QRod↵ Systems/src/managers/surfacetablemodel.cpp File Reference

Implementation of the SurfaceTableModel class.

```
#include <QTreeView>
#include "surfacetablemodel.h"
#include "core/surfacedataobject.h"
```

### 5.44.1 Detailed Description

Implementation of the SurfaceTableModel class.

#### Author

Pavel Lakiza

#### Date

March 2021

## 5.45 /home/qinterfly/Library/Projects/QRod↵ Systems/src/managers/surfacetablemodel.h File Reference

Declaration of the SurfaceTableModel class.

```
#include <QStandardItemModel>
#include "tablemodelinterface.h"
```

### Classes

- class [QRS::TableModels::SurfaceTableModel](#)

*Table model to represent a surface data object.*



### 5.45.1 Detailed Description

Declaration of the SurfaceTableModel class.

Author

Pavel Lakiza

Date

March 2021

## 5.46 /home/qinterfly/Library/Projects/QRod↵ Systems/src/managers/tablemodelinterface.cpp File Reference

Implementation of static functions of TableModelInterface.

```
#include <QStandardItem>
#include "tablemodelinterface.h"
#include "core/array.h"
```

### 5.46.1 Detailed Description

Implementation of static functions of TableModelInterface.

Author

Pavel Lakiza

Date

May 2021

## 5.47 /home/qinterfly/Library/Projects/QRod↵ Systems/src/managers/tablemodelinterface.h File Reference

Declaration of the TableModelInterface.

```
#include <QItemSelection>
```

### Classes

- class [QRS::TableModels::TableModelInterface](#)

*User interface to add and remove items.*

### 5.47.1 Detailed Description

Declaration of the TableModelInterface.

#### Author

Pavel Lakiza

#### Date

May 2021

## 5.48 /home/qinterfly/Library/Projects/QRod↵ Systems/src/models/abstrachierarchyitem.cpp File Reference

Definition of the AbstractHierarchyItem class.

```
#include "abstrachierarchyitem.h"  
#include "core/hierarchynode.h"
```

### 5.48.1 Detailed Description

Definition of the AbstractHierarchyItem class.

#### Author

Pavel Lakiza

#### Date

May 2021

## 5.49 /home/qinterfly/Library/Projects/QRod↵ Systems/src/models/abstrachierarchyitem.h File Reference

Declaration of the AbstractHierarchyItem class.

```
#include <QStandardItem>
```

### Classes

- class [QRS::HierarchyModels::AbstractHierarchyItem](#)  
*Item to represent a hierarchy of elements of the same type.*

## Enumerations

- enum **HierarchyItemType** { **kDataObjects** = QStandardItem::UserType }

### 5.49.1 Detailed Description

Declaration of the AbstractHierarchyItem class.

Author

Pavel Lakiza

Date

May 2021

## 5.50 /home/qinterfly/Library/Projects/QRodSystems/src/models/abstrachierarchyitem.cpp File Reference

Definition of the AbstractHierarchyModel class.

```
#include <QTreeView>
#include <QMimeData>
#include "abstrachierarchyitem.h"
#include "abstrachierarchyitem.h"
#include "core/hierarchyitem.h"
```

### 5.50.1 Detailed Description

Definition of the AbstractHierarchyModel class.

Author

Pavel Lakiza

Date

May 2021

## 5.51 /home/qinterfly/Library/Projects/QRodSystems/src/models/abstrachierarchyitem.h File Reference

Declaration of the AbstractHierarchyModel class.

```
#include <QStandardItemModel>
```

## Classes

- class [QRS::HierarchyModels::AbstractHierarchyModel](#)

*Hierarchy model which enables one to drag and drop elements of the same type.*

### 5.51.1 Detailed Description

Declaration of the AbstractHierarchyModel class.

#### Author

Pavel Lakiza

#### Date

May 2021

## 5.52 /home/qinterfly/Library/Projects/QRodSystems/src/render/view3d.cpp File Reference

Implementation of the View3D class.

```
#include <QOpenGLContext>
#include <QOpenGLPaintDevice>
#include <QPainter>
#include "view3d.h"
```

### 5.52.1 Detailed Description

Implementation of the View3D class.

#### Author

Pavel Lakiza

#### Date

March 2021

## 5.53 /home/qinterfly/Library/Projects/QRodSystems/src/render/view3d.h File Reference

Declaration of the View3D class.

```
#include <QOpenGLWidget>
#include <QOpenGLFunctions>
```

## Classes

- class [QRS::Graph::View3D](#)

*A widget to represent the resulted rod system.*

### 5.53.1 Detailed Description

Declaration of the View3D class.

#### Author

Pavel Lakiza

#### Date

March 2021

