QRodSystems

0.0.1

Generated by Doxygen 1.9.1

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

File Index

3.1 File List

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

Chapter 4

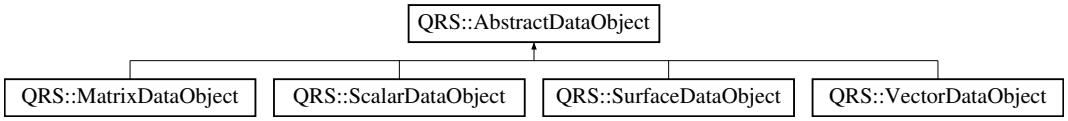
Class Documentation

4.1 QRS::AbstractDataObject Class Reference

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

`#include <abstractdataobject.h>`

Inheritance diagram for QRS::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 **numberObjects** ()
- static void **setNumberObjects** (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 designied 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::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::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.

4.2.1 Detailed Description

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

Numerical array class.

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

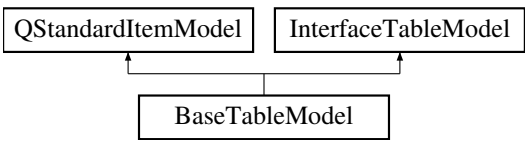
- /home/qinterfly/Library/Projects/QRodSystems/src/core/array.h
- /home/qinterfly/Library/Projects/QRodSystems/src/core/array.cpp

4.3 BaseTableModel Class Reference

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

`#include <basetablemodel.h>`

Inheritance diagram for BaseTableModel:



Public Member Functions

- **BaseTableModel** (QWidget *parent=nullptr)
- void setDataObject (QRS::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 *selectionModel) override

  Insert a new item after selected one.
- void **insertLeadingItemAfterSelected** (QItemSelectionModel *) override
- void removeSelectedItem (QItemSelectionModel *selectionModel) 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

- QRS::AbstractDataObject * **mpDataObject** = nullptr

Additional Inherited Members

4.3.1 Detailed Description

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

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

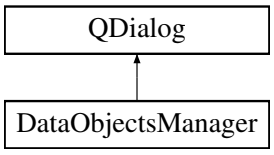
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/basetablemodel.h
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/basetablemodel.cpp

4.4 DataObjectsManager Class Reference

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

`#include <dataobjectsmanager.h>`

Inheritance diagram for DataObjectsManager:



Public Slots

- void apply ()

  Apply all the changes made by user.
- QRS::DataIDType addScalar ()

  Add a scalar object.
- QRS::DataIDType addVector ()

  Add a vector object.
- QRS::DataIDType addMatrix ()

  Add a matrix object.
- QRS::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 removeSelectedDataObject ()

  Remove a selected data object.
- void importDataObjects ()

  Import data objects from a file.

Public Member Functions

- **DataObjectsManager** (QRS::Project &project, QSettings &settings, QString &lastPath, QWidget *parent=nullptr)
- void closeEvent (QCloseEvent *event) override

  Save settings and delete handling widgets before closing the window.
- void selectDataObject (int index)

  Select a data object from the list.
- mapDataObjects const & **getDataObjects** ()

Private Types

- using **mapDataObjects** = std::unordered_map< QRS::DataIDType, QRS::AbstractDataObject * >

Private Slots

- void representSelectedDataObject ()

  Represent a selected data object according to its type.
- void renameDataObject (QListWidgetItem *item)

  Rename a data object.

Private Member Functions

- 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 (QRS::AbstractDataObject *dataObject)

  Helper function to insert data objects into the manager.
- void addListDataObjects (QRS::AbstractDataObject *dataObject)

  Add a data object to the list.
- 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**
- QListWidget * **mpListDataObjects**
- QTreeView * **mpDataTable**
- QRS::Project & **mProject**
- QSettings & **mSettings**
- mapDataObjects **mDataObjects**
- QString & **mLastPath**
- InterfaceTableModel * **mpInterfaceTableModel** = nullptr
- BaseTableModel * **mpBaseTableModel**
- MatrixTableModel * **mpMatrixTableModel**
- SurfaceTableModel * **mpSurfaceTableModel**

4.4.1 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:

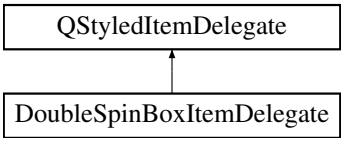
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectsmanager.h
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/dataobjectsmanager.cpp

4.5 DoubleSpinBoxItemDelegate Class Reference

Class to set how table values can be edited.

`#include <doublespinboxitemdelegate.h>`

Inheritance diagram for 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 ∗editor, const QModelIndex &index) const override

  Specify data to show.
- void setModelData (QWidget ∗editor, QAbstractItemModel ∗model, const QModelIndex &index) const override

  Set data to a model.
- void updateEditorGeometry (QWidget ∗editor, const QStyleOptionViewItem &option, const QModelIndex &index) const override

  Set a geometry to render.

4.5.1 Detailed Description

Class to set 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

4.6 QRS::HierarchyNode Class Reference

Hierarchy representative.

`#include <hierarchytree.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.

Private Attributes

- HierarchyNode * **mpParent** = nullptr
- HierarchyNode * **mpFirstChild** = nullptr
- HierarchyNode * **mpNextSibling** = nullptr
- NodeType **mType**
- QVariant **mValue**

Friends

- class **HierarchyTree**

4.6.1 Detailed Description

Hierarchy representative.

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

4.7 QRS::HierarchyTree Class Reference

Hierarchy of data objects (n-aray tree)

`#include <hierarchytree.h>`

Public Member Functions

- HierarchyTree ()
  Tree constructor.
- ~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 changeNodeValue (HierarchyNode::NodeType type, QVariant const &oldValue, QVariant const &newValue)
  Change the value of a node.
- HierarchyNode & **root** ()

Private Member Functions

- HierarchyNode * find (HierarchyNode *pBaseNode, HierarchyNode::NodeType type, QVariant const &value) const

  Find a node by type and value.
- void removeNode (HierarchyNode *pNode)

  Remove a node and all its subnodes.
- void removeChildren (HierarchyNode *pNode)

  Remove all subnodes.
- void printNode (uint level, HierarchyNode *pNode, QDebug stream)

  Print a current node and all its subnodes.

Private Attributes

- HierarchyNode * **mpRootNode**

Friends

- QDebug operator<< (QDebug stream, HierarchyTree &tree)

  Print a tree structure.

4.7.1 Detailed Description

Hierarchy of data objects (n-aray tree)

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

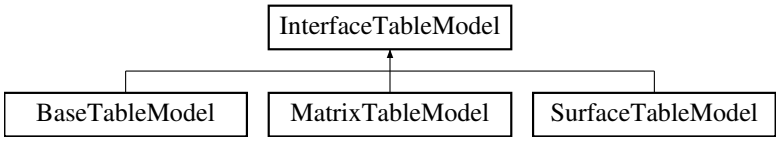
- /home/qinterfly/Library/Projects/QRodSystems/src/core/hierarchytree.h
- /home/qinterfly/Library/Projects/QRodSystems/src/core/hierarchytree.cpp

4.8 InterfaceTableModel Class Reference

User interface to add and remove items.

`#include <interfacetablemodel.h>`

Inheritance diagram for InterfaceTableModel:



Public Member Functions

- virtual void **insertItemAfterSelected** (QItemSelectionModel ∗selectionModel)=0
- virtual void **insertLeadingItemAfterSelected** (QItemSelectionModel ∗selectionModel)=0
- virtual void **removeSelectedItem** (QItemSelectionModel ∗selectionModel)=0
- virtual void **removeSelectedLeadingItem** (QItemSelectionModel ∗selectionModel)=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 (QRS::Array< double > &array, uint iRow)

  Helper function to copy a row from an array.
- static QList< QStandardItem ∗ > prepareRow (double const &key, QRS::Array< double > &array, uint iRow)

  Helper function to copy a row from an array and associate it with an key.
- static QList< QStandardItem ∗ > prepareRow (QString const &name, QRS::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 modfifed.

4.8.1 Detailed Description

User interface to add and remove items.

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

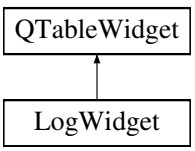
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/interfacetablemodel.h
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/interfacetablemodel.cpp

4.9 LogWidget Class Reference

Log all the messages sent.

`#include <logwidget.h>`

Inheritance diagram for LogWidget:



Public Member Functions

- **LogWidget** (QWidget ∗parent=nullptr)
- void log (QtMsgType messageType, const QString &message)

  Represent a message sent.

4.9.1 Detailed Description

Log all the messages sent.

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

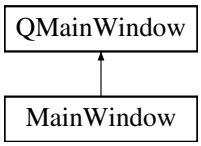
- /home/qinterfly/Library/Projects/QRodSystems/src/central/logwidget.h
- /home/qinterfly/Library/Projects/QRodSystems/src/central/logwidget.cpp

4.10 MainWindow Class Reference

The main window of the program.

`#include <mainwindow.h>`

Inheritance diagram for 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 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 *event) 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 properies of selected objercts.
- 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**
- QSharedPointer< QSettings > **mpSettings**
- DataObjectsManager * **mpDataObjectsManager** = nullptr
- QRS::Project * **mpProject**
- QString **mLastPath**
- QList< QString > **mPathRecentProjects**

4.10.1 Detailed Description

The main window of the program.

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

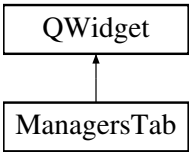
- /home/qinterfly/Library/Projects/QRodSystems/src/central/mainwindow.h
- /home/qinterfly/Library/Projects/QRodSystems/src/central/mainwindow.cpp

4.11 ManagersTab Class Reference

A toolbar consisted of object designers.

`#include <controltabs.h>`

Inheritance diagram for ManagersTab:



Signals

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

Public Member Functions

- ManagersTab (QWidget *parent=nullptr)

  Managers tab constructor.

4.11.1 Detailed Description

A toolbar consisted of object designers.

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

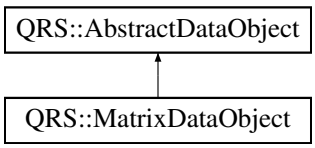
- /home/qinterfly/Library/Projects/QRodSystems/src/central/controltabs.h
- /home/qinterfly/Library/Projects/QRodSystems/src/central/controltabs.cpp

4.12 QRS::MatrixDataObject Class Reference

Matrix data object.

`#include <matrixdataobject.h>`

Inheritance diagram for QRS::MatrixDataObject:



Public Member Functions

- MatrixDataObject (QString const &name)

  Construct a matrix data object.
- AbstractDataObject * clone () const override

  Clone a matrix data object.
- DataItemType & 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

4.12.1 Detailed Description

Matrix data object.

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

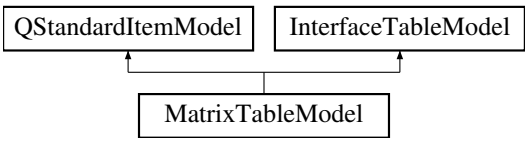
- /home/qinterfly/Library/Projects/QRodSystems/src/core/matrixdataobject.h
- /home/qinterfly/Library/Projects/QRodSystems/src/core/matrixdataobject.cpp

4.13 MatrixTableModel Class Reference

Table model to represent a matrix data object.

`#include <matrixtablemodel.h>`

Inheritance diagram for MatrixTableModel:



Public Member Functions

- **MatrixTableModel** (QWidget *parent=nullptr)
- void setDataObject (QRS::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 *selectionModel) override

  Insert a new item after selected one.
- void **insertLeadingItemAfterSelected** (QItemSelectionModel *) override
- void removeSelectedItem (QItemSelectionModel *selectionModel) 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

- QRS::AbstractDataObject * **mpDataObject** = nullptr

Additional Inherited Members

4.13.1 Detailed Description

Table model to represent a matrix data object.

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

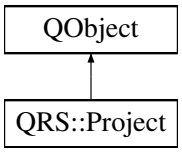
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/matrixtablemodel.h
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/matrixtablemodel.cpp

4.14 QRS::Project Class Reference

Project class to interact with a created system of rods.

`#include <project.h>`

Inheritance diagram for QRS::Project:



Public Slots

- bool save (QString const &dir, QString const &fileName)

  Save a project to a file.

Signals

- void **dataObjectAdded** (QRS::DataIDType id)
- void **dataObjectRemoved** (QRS::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::shared_ptr< AbstractDataObject > getDataObject (DataIDType id)

  Retrieve a data object by identificator.
- 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)

  Substitute current data objects with new ones.
- 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.

4.14.1 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

4.15 QRS::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**

4.15.1 Detailed Description

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

Proxy class to acquire a row by index.

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

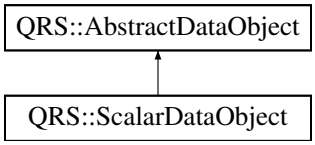
- /home/qinterfly/Library/Projects/QRodSystems/src/core/array.h

4.16 QRS::ScalarDataObject Class Reference

Scalar data object.

`#include <scalardataobject.h>`

Inheritance diagram for QRS::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

4.16.1 Detailed Description

Scalar data object.

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

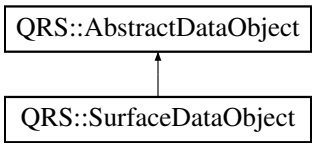
- /home/qinterfly/Library/Projects/QRodSystems/src/core/scalardataobject.h
- /home/qinterfly/Library/Projects/QRodSystems/src/core/scalardataobject.cpp

4.17 QRS::SurfaceDataObject Class Reference

Surface data object.

`#include <surfacedataobject.h>`

Inheritance diagram for QRS::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

4.17.1 Detailed Description

Surface data object.

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

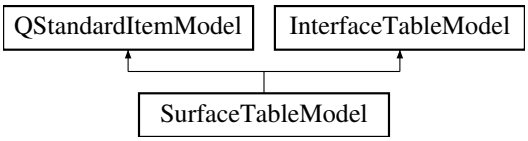
- /home/qinterfly/Library/Projects/QRodSystems/src/core/surfacedataobject.h
- /home/qinterfly/Library/Projects/QRodSystems/src/core/surfacedataobject.cpp

4.18 SurfaceTableModel Class Reference

Table model to represent a surface data object.

`#include <surfacetablemodel.h>`

Inheritance diagram for SurfaceTableModel:



Public Member Functions

- **SurfaceTableModel** (QWidget *parent=nullptr)
- void setDataObject (QRS::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 *selectionModel) override

  Insert a new item after selected one.
- void removeSelectedItem (QItemSelectionModel *selectionModel) override

  Remove an array under selection.
- void insertLeadingItemAfterSelected (QItemSelectionModel *selectionModel) override

  Add a new leading item after selected one.
- void removeSelectedLeadingItem (QItemSelectionModel *selectionModel) 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

- QRS::SurfaceDataObject * **mpDataObject** = nullptr

Additional Inherited Members

4.18.1 Detailed Description

Table model to represent a surface data object.

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

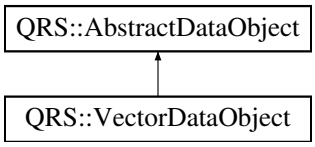
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/surfacetablemodel.h
- /home/qinterfly/Library/Projects/QRodSystems/src/managers/surfacetablemodel.cpp

4.19 QRS::VectorDataObject Class Reference

Vector data object.

`#include <vectordataobject.h>`

Inheritance diagram for QRS::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

4.19.1 Detailed Description

Vector data object.

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

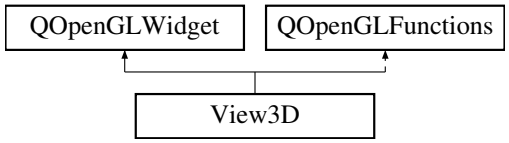
- /home/qinterfly/Library/Projects/QRodSystems/src/core/vectordataobject.h
- /home/qinterfly/Library/Projects/QRodSystems/src/core/vectordataobject.cpp

4.20 View3D Class Reference

A widget to represent the resulted rod system.

`#include <view3d.h>`

Inheritance diagram for 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**

4.20.1 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 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

March 2021

5.4 /home/qinterfly/Library/Projects/QRod↩ Systems/src/central/logwidget.h File Reference

Declaration of the LogWidget class.

```
#include <QTableWidget>
```

Classes

- class LogWidget

  Log all the messages sent.

5.4.1 Detailed Description

Declaration of the LogWidget class.

Author

Pavel Lakiza

Date

March 2021

5.5 /home/qinterfly/Library/Projects/QRod↩ Systems/src/central/mainwindow.cpp File Reference

Implementation of the MainWindow class.

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

April 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 MainWindow

  The main window of the program.

5.6.1 Detailed Description

Declaration of the MainWindow class.

Author

Pavel Lakiza

Date

March 2021

5.7 /home/qinterfly/Library/Projects/QRod↩ Systems/src/central/uiconstants.h File Reference

Common graphical constants shared between several windows.

```
#include <QString>
```

Variables

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

5.7.1 Detailed Description

Common graphical constants shared between several windows.

Author

Pavel Lakiza

Date

April 2021

5.8 /home/qinterfly/Library/Projects/QRod↩ Systems/src/core/abstractdataobject.cpp File Reference

Implementation of the AbstractDataObject class.

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

5.8.1 Detailed Description

Implementation of the AbstractDataObject class.

Author

Pavel Lakiza

Date

April 2021

5.9 /home/qinterfly/Library/Projects/QRod↩ Systems/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::AbstractDataObject

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

Typedefs

- using **QRS::DataItemType** = Array< DataValueType >
- using **QRS::DataHolder** = std::map< DataKeyType, DataItemType >

Functions

- QDataStream & QRS::operator<< (QDataStream &stream, AbstractDataObject const &obj)

  Print a data object to a stream.
- QDataStream & QRS::operator>> (QDataStream &stream, AbstractDataObject &obj)

  Read a data object from a stream.

5.9.1 Detailed Description

Declaration of the AbstractDataObject class.

Author

Pavel Lakiza

Date

April 2021

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

Implementation of the Array class.

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

5.10.1 Detailed Description

Implementation of the Array class.

Author

Pavel Lakiza

Date

March 2021

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

Declaration of the Array class.

```
#include <QDebug>
```

Classes

- class QRS::Array< T >
  Numerical array class.
- struct QRS::Array< T >::Row< U >
  Proxy class to acquire a row by index.

Typedefs

- using **QRS::IndexType** = unsigned int

Functions

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

5.11.1 Detailed Description

Declaration of the Array class.

Author

Pavel Lakiza

Date

March 2021

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

Specification of data types used in a project.

Typedefs

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

Enumerations

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

5.12.1 Detailed Description

Specification of data types used in a project.

Author

Pavel Lakiza

Date

March 2021

5.13 /home/qinterfly/Library/Projects/QRod↩ Systems/src/core/hierarchytree.cpp File Reference

Implementation of the HierarchyNode and HierarchyTree classes.

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

5.13.1 Detailed Description

Implementation of the HierarchyNode and HierarchyTree classes.

Author

Pavel Lakiza

Date

April 2021

5.14 /home/qinterfly/Library/Projects/QRod↩ Systems/src/core/hierarchytree.h File Reference

Declaration of the HierarchyNode and HierarchyTree classes.

```
#include <QVariant>
#include <QDebug>
```

Classes

- class QRS::HierarchyNode
  Hierarchy representative.
- class QRS::HierarchyTree
  Hierarchy of data objects (n-aray tree)

Functions

- QDebug QRS::operator<< (QDebug stream, HierarchyTree &tree)
  Print a tree structure.

5.14.1 Detailed Description

Declaration of the HierarchyNode and HierarchyTree classes.

Author

Pavel Lakiza

Date

April 2021

5.15 /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

5.15.1 Detailed Description

Implementation of the MatrixDataObject class.

Author

Pavel Lakiza

Date

April 2021

5.16 /home/qinterfly/Library/Projects/QRod↩ Systems/src/core/matrixdataobject.h File Reference

Declaration of the MatrixDataObject class.

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

Classes

- class QRS::MatrixDataObject
  Matrix data object.

5.16.1 Detailed Description

Declaration of the MatrixDataObject class.

Author

Pavel Lakiza

Date

April 2021

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

Implementation of the QRS::Project class.

```
#include <QDebug>
#include <QRandomGenerator>
#include <QFileInfo>
#include <QDir>
#include <QDataStream>
#include <QDateTime>
#include "project.h"
#include "scalardataobject.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.

5.17.1 Detailed Description

Implementation of the QRS::Project class.

Author

Pavel Lakiza

Date

April 2021

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

Declaration of the QRS::Project class.

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

Classes

- class QRS::Project

  Project class to interact with a created system of rods.

Typedefs

- using **QRS::DataObjects** = std::unordered_map< DataIDType, std::shared_ptr< AbstractDataObject > >

5.18.1 Detailed Description

Declaration of the QRS::Project class.

Author

Pavel Lakiza

Date

April 2021

5.19 /home/qinterfly/Library/Projects/QRod↩ Systems/src/core/scalardataobject.cpp File Reference

Implementation of the ScalarDataObject class.

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

5.19.1 Detailed Description

Implementation of the ScalarDataObject class.

Author

Pavel Lakiza

Date

April 2021

5.20 /home/qinterfly/Library/Projects/QRod↩ Systems/src/core/scalardataobject.h File Reference

Declaration of the ScalarDataObject class.

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

Classes

- class QRS::ScalarDataObject

  Scalar data object.

5.20.1 Detailed Description

Declaration of the ScalarDataObject class.

Author

Pavel Lakiza

Date

April 2021

5.21 /home/qinterfly/Library/Projects/QRod↩ Systems/src/core/surfacedataobject.cpp File Reference

Implementation of the SurfaceDataObject class.

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

5.21.1 Detailed Description

Implementation of the SurfaceDataObject class.

Author

Pavel Lakiza

Date

April 2021

5.22 /home/qinterfly/Library/Projects/QRod↩ Systems/src/core/surfacedataobject.h File Reference

Declaration of the SurfaceDataObject class.

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

Classes

- class QRS::SurfaceDataObject

  Surface data object.

5.22.1 Detailed Description

Declaration of the SurfaceDataObject class.

Author

Pavel Lakiza

Date

April 2021

5.23 /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"
```

5.23.1 Detailed Description

Implementation of utilities.

Author

Pavel Lakiza

Date

April 2021

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

Declaration of utilities.

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

Functions

- QPair< QRS::DataObjectType, QSharedPointer< QFile > > Utilities::File::getDataObjectFile (QString const &path, QString const &fileName)

  Retrieve a pair consisted of a data object file and its type.

5.24.1 Detailed Description

Declaration of utilities.

Author

Pavel Lakiza

Date

April 2021

5.25 /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

5.25.1 Detailed Description

Implementation of the VectorDataObject class.

Author

Pavel Lakiza

Date

April 2021

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

Declaration of the VectorDataObject class.

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

Classes

- class QRS::VectorDataObject

  Vector data object.

5.26.1 Detailed Description

Declaration of the VectorDataObject class.

Author

Pavel Lakiza

Date

April 2021

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

The startup function.

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

Functions

- void throwMessage (QtMsgType type, const QMessageLogContext &, const QString &message)

  Log all the messages.
- int main (int argc, char ∗argv[ ])

  Entry point.

5.27.1 Detailed Description

The startup function.

Author

Pavel Lakiza

Date

March 2021

5.28 /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"
```

5.28.1 Detailed Description

Implementation of the BaseTableModel class.

Author

Pavel Lakiza

Date

March 2021

5.29 /home/qinterfly/Library/Projects/QRod↩ Systems/src/managers/basetablemodel.h File Reference

Declaration of the BaseTableModel class.

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

Classes

- class BaseTableModel

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

5.29.1 Detailed Description

Declaration of the BaseTableModel class.

Author

Pavel Lakiza

Date

March 2021

5.30 /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 "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"
```

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"

5.30.1 Detailed Description

Implementation of the DataObjectsManager class.

Author

Pavel Lakiza

Date

March 2021

5.31 /home/qinterfly/Library/Projects/QRod↩ Systems/src/managers/dataobjectsmanager.h File Reference

Declaration of the DataObjectsManager class.

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

Classes

- class DataObjectsManager

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

5.31.1 Detailed Description

Declaration of the DataObjectsManager class.

Author

Pavel Lakiza

Date

March 2021

5.32 /home/qinterfly/Library/Projects/QRod↩ Systems/src/managers/doublespinboxitemdelegate.cpp File Reference

Implementation of the DoubleSpinBoxItemDelegate class.

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

5.32.1 Detailed Description

Implementation of the DoubleSpinBoxItemDelegate class.

Author

Pavel Lakiza

Date

March 2021

5.33 /home/qinterfly/Library/Projects/QRod↩ Systems/src/managers/doublespinboxitemdelegate.h File Reference

Declaration of the DoubleSpinBoxItemDelegate class.

```
#include <QStyledItemDelegate>
```

Classes

- class DoubleSpinBoxItemDelegate
  Class to set how table values can be edited.

5.33.1 Detailed Description

Declaration of the DoubleSpinBoxItemDelegate class.

Author

Pavel Lakiza

Date

March 2021

5.34 /home/qinterfly/Library/Projects/QRod↩ Systems/src/managers/interfacetablemodel.cpp File Reference

Implementation of static functions of InterfaceTableModel.

```
#include <QStandardItem>
#include "interfacetablemodel.h"
#include "array.h"
```

5.34.1 Detailed Description

Implementation of static functions of InterfaceTableModel.

Author

Pavel Lakiza

Date

March 2021

5.35 /home/qinterfly/Library/Projects/QRod↩ Systems/src/managers/interfacetablemodel.h File Reference

Interface of a table model.

```
#include <QItemSelection>
```

Classes

- class InterfaceTableModel

  User interface to add and remove items.

5.35.1 Detailed Description

Interface of a table model.

Author

Pavel Lakiza

Date

March 2021

5.36 /home/qinterfly/Library/Projects/QRod↩ Systems/src/managers/matrixtablemodel.cpp File Reference

Implementation of the MatrixTableModel class.

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

5.36.1 Detailed Description

Implementation of the MatrixTableModel class.

Author

Pavel Lakiza

Date

March 2021

5.37 /home/qinterfly/Library/Projects/QRod↩ Systems/src/managers/matrixtablemodel.h File Reference

Declaration of the MatrixTableModel class.

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

Classes

- class MatrixTableModel

  Table model to represent a matrix data object.

5.37.1 Detailed Description

Declaration of the MatrixTableModel class.

Author

Pavel Lakiza

Date

March 2021

5.38 /home/qinterfly/Library/Projects/QRod↩ Systems/src/managers/surfacetablemodel.cpp File Reference

Implementation of the SurfaceTableModel class.

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

5.38.1 Detailed Description

Implementation of the SurfaceTableModel class.

Author

Pavel Lakiza

Date

March 2021

5.39 /home/qinterfly/Library/Projects/QRod↩ Systems/src/managers/surfacetablemodel.h File Reference

Declaration of the SurfaceTableModel class.

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

Classes

- class SurfaceTableModel

  Table model to represent a surface data object.

5.39.1 Detailed Description

Declaration of the SurfaceTableModel class.

Author

Pavel Lakiza

Date

March 2021

5.40 /home/qinterfly/Library/Projects/QRod↩ Systems/src/render/view3d.cpp File Reference

Implementation of the View3D class.

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

5.40.1 Detailed Description

Implementation of the View3D class.

Author

Pavel Lakiza

Date

March 2021

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

Declaration of the View3D class.

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

Classes

- class View3D

  A widget to represent the resulted rod system.

5.41.1 Detailed Description

Declaration of the View3D class.

Author

Pavel Lakiza

Date

March 2021