Doxygen

Use in MITK

dkfz. GERMAN CANCER RESEARCH CENTER IN THE HELMHOLTZ ASSOCIATION

What is doxygen

- Doxygen is documentation system for
  - generating on-/offline documentation from documented source files
  - extracting code structure from undocumented source files
  - generating user documentation

Choices

- There are two ways to document the code
    1. Special documentation block in front of the declaration or definition. For members directly after works too.
    2. Special documentation block somewhere else (another file or another location) with a structural command in the documentation block. A structural command links a documentation block to a certain entity that can be documented (e.g. a member, class, namespace or file).
- Documentation blocks can be put anywhere but inside the body of a function or inside a normal C style comment block.

Choices

dkfz.

- Doxygen allows for several different styles:
  - /**
    * …text …
    */

  - /*!
    * …text …
    */

  - ///
    /// … text …
    ///

  - //!
    //! … text …
    //!

- Intermediate * are optional

- Last two
  - need at least two lines, otherwise it is considered the brief description
  - Ended by blank line

Choices

dkfz.

```
/**
 * Text before the list
 *  - list item 1
 *    - sub item 1
 *      - sub sub item 1
 *      - sub sub item 2
 *      .
 *      The dot above ends the sub sub item list.
 *       More text for the first sub item
 *    .
 *    The dot above ends the first sub item.
 *    More text for the first list item
 *    - sub item 2
 *    - sub item 3
 * - list item 2
 * .
 * More text in the same paragraph.
 *
 * More text in a new paragraph.
 */
```

```
/*!
 * A list of events:
 *   <ul>
 *   <li> mouse events
 *     <ol>
 *     <li>mouse move event
 *     <li>mouse click event\n
 *         More info about the click event.
 *     <li>mouse double click event
 *     </ol>
 *   <li> keyboard events
 *     <ol>
 *     <li>key down event
 *     <li>key up event
 *     </ol>
 *   </ul>
 * More text here.
 */
```

Brief, detailed and commands

dkfz.

- Each item can have a brief and a detailed description
- Multiple detailed descriptions will be joined
- Commands start with \ or @ (user's choice)

Parsing

- During parsing the following steps take place:
  - The special commands inside the documentation are executed.
  - If a line starts with some whitespace followed by one or more asterisks () and then optionally more whitespace, then all whitespace and asterisks are removed.
  - All resulting blank lines are treated as a paragraph separators. This saves you from placing newparagraph commands yourself in order to make the generated documentation readable.
  - Links are created for words corresponding to documented classes (unless the word is preceded by a %; then the word will not be linked and the % sign is removed).
  - Links to members are created when certain patterns are found in the text.
  - HTML tags that are in the documentation are interpreted and converted to LATEX equivalents for the LATEX output.

Examples – Source vs Result

dkfz.

```
virtual void    InsertNode (int index, const mitk::DataTreeNode *_DataTreeNode)
                Inserts a new node at the given index. If the index does not exist, the _DataTreeNode is
                simply appended to the combobox.
```

```
void QmitkDataStorageComboBox::InsertNode ( int                    index,
                                            const mitk::DataTreeNode * _DataTreeNode
                                          )                                [protected, virtual]
```

Inserts a new node at the given index. If the index does not exist, the _DataTreeNode is simply appended to the combobox.

This function is used by AddNode() and SetNode() because they just to the same: 1. If node is replaced (that is when index exists), the itk::Event observer will be removed 2. Check Node against Predicate 3. Register for itk::Events on the node 4. Insert Node and show in combobox

property of the node because this is the only thing we are interested in !!!!!

Definition at line 278 of file QmitkDataStorageComboBox.cpp.

References Find(), mitk::DataTreeNode::GetProperty(), mitk::BaseProperty::GetValueAsString(), HasIndex(), m_AutoSelectNewNodes, m_Nodes, m_NodesDeleteObserverTags, m_NodesModifiedObserverTags, m_Predicate, m_PropertyToNode, OnDataTreeNodeDeleteOrModified(), and RemoveNode().

Referenced by AddNode(), and SetNode().

```
00279 {
00280    // check new or updated node first
00281    if(m_Predicate.IsNotNull() && !m_Predicate->CheckNode(_DataTreeNode))
```

///

/// \brief Inserts a new node at the given index. If the index does not exist, the _DataTreeNode is simply appended to the combobox.

///

/// This function is used by AddNode() and SetNode() because they just to the same:

/// 1. If node is replaced (that is when index exists),

///   the itk::Event observer will be removed

/// 2. Check Node against Predicate

/// 3. Register for itk::Events on the node

/// 4. Insert Node and show in combobox

virtual void InsertNode(int index, const mitk::DataTreeNode* _DataTreeNode);

/*! \mainpage A simple manual

This manual is divided in the following sections:
- \subpage intro
- \subpage advanced "Advanced usage"
*/

//--

/*! \page intro Introduction
This page introduces the user to the topic.
\section disclaimer Disclaimer
This section contains the disclaimer
\section content
Here is general information.
\subsection donot Things not to do
Advice.
\subsection do Things to do
More Advice.

Now you can proceed to the \ref advanced
    "advancedsection".
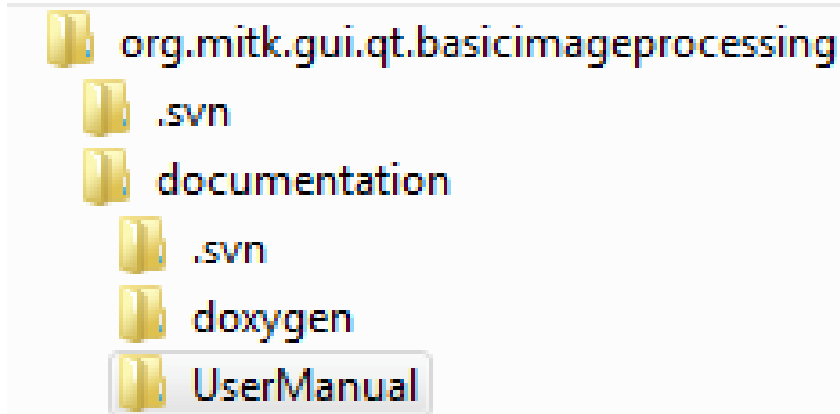*/

//--

/*! \page advanced Advanced Usage
This page is for advanced users.
Make sure you have first read \ref intro
    "the introduction".
*/

User manual in MITK

dkfz.

```
org.mitk.gui.qt.basicimageprocessing
  .svn
  documentation
    .svn
    doxygen
    UserManual
```

```
/**
\bundlemainpage{<BUNDLE_IDENTIFIER>}
<BUNDLE_NAME>

\image html <ICON_FILE> "Icon of the Module"

\section <SECTION_IDENTIFIER> <SECTION_NAME>

Text
…
*/
```

- Doxygen can generate QT help files for runtime access (F1)
- Put .dox file in the *UserManual* subdirectory of *documentation*
- Do **not** use pages, only sections

Doxygen documentation

- Online manual
  http://www.stack.nl/~dimitri/doxygen/manual.html
- Command reference
  http://www.stack.nl/~dimitri/doxygen/commands.html