

BugSquashing:

How to use XML in MITK ?

Bastian Graser
23th Feb. 2011

What is XML?

- Standard format
- Easy-to-read
- To save/load parameter sets, etc
- In MITK: TinyXml & QT

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>Tove</to>
  <from adress="Sydney">Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me!</body>
  <image src="kiss.jpg" />
</note>
```

TinyXML

```
#include "tinyxml.H"
```

```
// read in file
```

```
TiXmlDocument xmlDoc("c:\\Temp\\bla.xml");  
xmlDoc.LoadFile();
```

```
#include "tinyxml.H"
```

```
// read in file
```

```
TiXmlDocument xmlDoc("c:\\Temp\\bla.xml");  
xmlDoc.LoadFile();
```

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<note>
```

```
  <to>Tove</to>
```

```
  <from adress="Sydney">Jani</from>
```

```
  <heading>Reminder</heading>
```

```
  <body>Don't forget me!</body>
```

```
  <image src="kiss.jpg" />
```

```
</note>
```

```
#include "tinyxml.H"
```

```
// read in file
```

```
TiXmlDocument xmlDoc("c:\\Temp\\bla.xml");  
xmlDoc.LoadFile();
```

```
// parse
```

```
TiXmlElement* el = xmlDoc.FirstChildElement("note");  
el = el->FirstChildElement();  
el = el->NextSiblingElement();  
std::string s = el->Attribute("adress");
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<note>  
  <to>Tove</to>  
  <from adress="Sydney">Jani</from>  
  <heading>Reminder</heading>  
  <body>Don't forget me!</body>  
  <image src="kiss.jpg" />  
</note>
```

```
#include "tinyxml.H"
```

// read in file

```
TiXmlDocument xmlDoc("c:\\Temp\\bla.xml");  
xmlDoc.LoadFile();
```

// parse

```
TiXmlElement* el = xmlDoc.FirstChildElement("note");  
el = el->FirstChildElement();  
el = el->NextSiblingElement();  
std::string s = el->Attribute("adress");
```

// write

```
TiXmlElement * newEl = new TiXmlElement("CC");  
newEl->LinkEndChild( new TiXmlText ("Anna"));  
el->LinkEndChild(newEl);  
xmlDoc.SaveFile("c:\\Temp\\bla.xml");
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<note>  
  <to>Tove</to>  
  <from adress="Sydney">Jani</from>  
  <heading>Reminder</heading>  
  <body>Don't forget me!</body>  
  <image src="kiss.jpg" />  
</note>
```



```
#include "tinyxml.H"
```

// read in file

```
TiXmlDocument xmlDoc("c:\\Temp\\bla.xml");  
xmlDoc.LoadFile();
```

// parse

```
TiXmlElement* el = xmlDoc.FirstChildElement("note");  
el = el->FirstChildElement();  
el = el->NextSiblingElement();  
std::string s = el->Attribute("adress");
```

// write

```
TiXmlElement * newEl = new TiXmlElement("CC");  
newEl->LinkEndChild( new TiXmlText ("Anna"));  
el->LinkEndChild(newEl);  
xmlDoc.SaveFile("c:\\Temp\\bla.xml");
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<note>  
  <to>Tove</to>  
  <from adress="Sydney">Jani  
    <cc>Anna</cc> </from>  
  <heading>Reminder</heading>  
  <body>Don't forget me!</body>  
  <image src="kiss.jpg" />  
</note>
```

QT

// adapt CMAKE

```
SET (QT_USE_QTXML TRUE)  
INCLUDE(${QT_USE_FILE})
```

```
#include <QDomDocument>
```

```
// read file
```

```
QDomDocument doc ("Adressbook");
```

```
QFile file ("c:\\Temp\\bla.xml");
```

```
file.open(QIODevice::ReadOnly);
```

```
doc.setContent(&file);
```

```
file.close();
```

// parse

```
QDomElement root = doc.documentElement();  
QDomElement el = root.firstChildElement();  
el = el.nextSiblingElement();  
QString s = el.attribute("adress", "" );
```

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<note>
```

```
  <to>Tove</to>
```

```
  <from adress="Sydney">Jani</from>
```

```
  <heading>Reminder</heading>
```

```
  <body>Don't forget me!</body>
```

```
  <image src="kiss.jpg" />
```

```
</note>
```

// parse

```
QDomElement root = doc.documentElement();  
QDomElement el = root.firstChildElement();  
el = el.nextSiblingElement();  
QString s = el.attribute("adress", "" );
```

// write

```
QDomElement newEl = doc.createElement("cc");  
QDomText text = doc.createTextNode("Anna");  
el.appendChild(newEl);  
newEl.appendChild(text);
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<note>  
  <to>Tove</to>  
  <from adress="Sydney">Jani</from>  
  <heading>Reminder</heading>  
  <body>Don't forget me!</body>  
  <image src="kiss.jpg" />  
</note>
```

// parse

```
QDomElement root = doc.documentElement();  
QDomElement el = root.firstChildElement();  
el = el.nextSiblingElement();  
QString s = el.attribute("adress", "" );
```

// write

```
QDomElement newEl = doc.createElement("cc");  
QDomText text = doc.createTextNode("Anna");  
el.appendChild(newEl);  
newEl.appendChild(text);
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<note>  
  <to>Tove</to>  
  <from adress="Sydney">Jani  
    <cc>Anna</cc> </from>  
  <heading>Reminder</heading>  
  <body>Don't forget me!</body>  
  <image src="kiss.jpg" />  
</note>
```

// parse

```
QDomElement root = doc.documentElement();  
QDomElement el = root.firstChildElement();  
el = el.nextSiblingElement();  
QString s = el.attribute("adress", "" );
```

// write

```
QDomElement newEl = doc.createElement("cc");  
QDomText text = doc.createTextNode("Anna");  
el.appendChild(newEl);  
newEl.appendChild(text);
```

```
file.open(QIODevice::ReadWrite);  
QTextStream ts( &file );  
ts << doc.toString();  
file.close();
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<note>  
  <to>Tove</to>  
  <from adress="Sydney">Jani  
    <cc>Anna</cc> </from>  
  <heading>Reminder</heading>  
  <body>Don't forget me!</body>  
  <image src="kiss.jpg" />  
</note>
```


- TinyXml
 - <http://www.grinninglizard.com/tinyxmldocs/index.html>
 - Easy
 - When wrong -> crash
- QT
 - We need QT anyway
 - Little bit more complex
 - When wrong -> error msg, continues
 - Nice small Tutorial:
http://www.digitalfanatics.org/projects/qt_tutorial/chapter09.html