



Testing Infrastructure :

GUI Testing

Benjamin Long & Jean-Christophe Fillion-Robin
Kitware Inc.
January 10th 2012

Introduction :

- What is *QtTesting* ?
- How it works ?
- Test for each widgets
- RenderWindow and GUI Testing
- Slicer integration
- What's next ?

What is “QtTesting” ?

Pros :

- “Hight-level” events instead of “Low-level” events
 - Low-level events (Mouse events, ...)
 - Hight-level events: Signals emitted by widgets (“clicked”, “currentItemChanged”, ...)
- Record and playback scripts regardless any coordinates.
- Be able to play a same script under every computer.
- Easy integration.

Cons :

- Create additional handlers for highly “customized” widget.

How it works ?

➤ *Every types of widgets get their own translator/player*

- Translators :

- Files to record “High-level” events from one widget

- Players :

- Files to playback events to one widget

➤ *Scripts look like to*

```
<event widget="qSlicerMainWindow/menubar" command="activate" arguments="menuFile"/>  
<event widget="qSlicerMain.../menuFile" command="activate" arguments="actionFileLoadScene"/>
```

Test for each widgets:

- Playing/Recording capabilities tested for every Widget

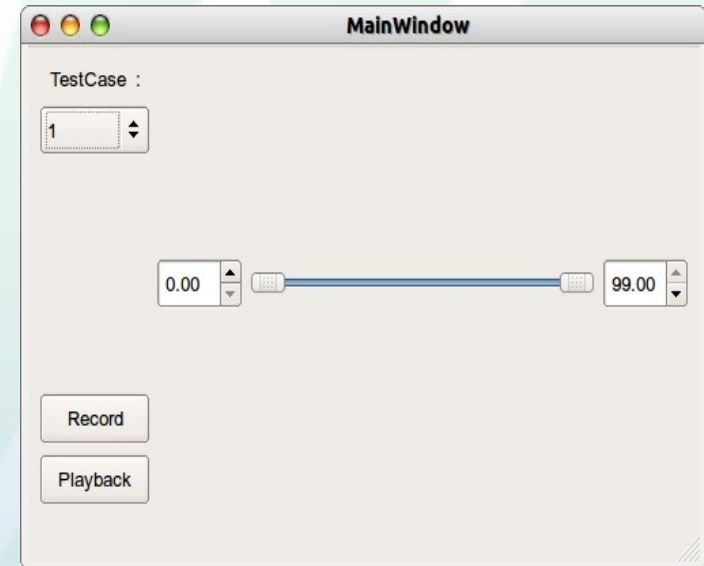
- C++ test very easy to create
- Test is composed of TestCases

Every TestCases:

- Play an XML file
- Current values == Expected values ?

- Why ?

- Be able to know if we need new Translators/Players
- Easy way to test our translators and our players



RenderWindow and GUI Testing:

Limits :

- We still record some Mouse events.
 - Ex : CTKVTKRenderWindow
 - The application has to have the same size.
- All translators/players are not yet implemented.
- No methods to test the scripts played on slicer4.

Slicer integration :

- Just initialize the variable `ctkQtTestingUtility`

```
this->TestUtility = new ctkQtTestingUtility(this);  
this->TestUtility->addEventObserver("xml", new ctkXMLEventObserver(this));  
this->TestUtility->addEventSource("xml", new ctkXMLEventSource(this));
```

- This variable :
 - ✓ Knows all the translators/players
 - ✓ Can write the script, by the `ctkEventObserver`.
 - ✓ Can read the script, by the `ctkEventSource`.

What's next ?

- Define Slicer metrics to validate GUI testing (Screenshots, diff, volume comparison, ...)
- Execute GUI tests and submit result to CDash
- Re-play macro independently of the underlying file system.
- Record event as a sequence of python statements.
- Finer control over the “events” player (pause, stop, progress feedback)

For more details :

- Wiki : http://www.commonk.org/index.php/Documentation/Widget_testing

Thank you !