

Creating custom scicos blocks using the `xml2sci` application

Roberto Bucher
Scuola Universitaria Professionale
della Svizzera Italiana (SUPSI)

November 26, 2006

Abstract

This document explains how the two applications `qxml2sci` and `xml2sci` can be used to automatically create scicos custom blocks. The `qxml2sci` creates a XML intermediate file, which is transformed by the `xml2sci` application into scilab code.

1 Installing the two packages

1. Get the tarball from `www.dti.supsi.ch/~bucher/`.
2. `untar` the packages.
3. Create a link under `/usr/local/bin` pointing to the `qxml2sci` application.

The files under `xml2sci` can be simply compiled using `make`. Use `kdevelop` to compile the files under `qxml2sci`.

2 Create a custom block

Remark Most of these steps can be performed by simply clicking with the right mouse button on the field.

1. Start `qxml2sci`. The first time you enter this application you are asked to insert the path to the `xml2sci` executable.
2. Create a new empty document.
3. Choose the *blocktype*
 - *input* means a block which has only output ports
 - *output* means a block which has only input ports
 - *inout* means a block which can have input and output ports
 - The default number of input and output ports is 1. User can parametrize this values by creating a variable called *inport* (for output blocks) or *outport* (for input block).
4. Choose a name for your block Now you are ready to insert all the needed interface parameters for your block.

3 Create the parameters

As last step the user should fill the field *dialog* which has different subitems

desc Is the text of the dialog box used by scicos to enter the parameters

item Contains all the fields needed to create a new variable for block parameters

varname The name of the variable

prompt The Prompt message of the dialog box for this variable

type String (str) or number (vec)

init Default init value

passingAs This field can be used to decide how to pass the variable to the C interface

- “no”: Dont’ pass it
- “rpar”: Pass the variable as double using the rpar vector
- “ipar”: Pass the variable as int number using the ipar vector
- “string”: Pass the variable as string using the ipar vector
- “hex_addr”: Pass the variable as integer value after transforming the string ox the Hex value (0xXXXX) to an integer value using the ipar vector.

visibilty This field can be used to decide if the value of the variable compares in the Scicos block (“true” or “false”)