

SAGA-ADG O. Weidner, A. Merzky - Center for Computation & Technology

Version: 0.2

December 14, 2011

SAGA C++ Adaptor Development Guide

Abstract

This documents explains how to write new adaptors for SAGA

Copyright Notice

Copyright © CCT / Louisiana State Univeristy(2007-2011).
All Rights Reserved.

Contents

1	Writing new SAGA adaptors	3
1.1	Overview	3
	References	5

1 Writing new SAGA adaptors

1.1 Overview

The SAGA C++ sources include an adaptor generator, which allows to easily create stubs for custom adaptors. The script is located in

`adaptors/generator/generator.pl`

and is installed into `\$SAGA_LOCATION/bin/`. Calling that script without any arguments will print a help screen, which provides a number of details on the command line arguments etc.

The exemplary shell session shown below demonstrates the use of the adaptor generator, and results in a complete file adaptor:

```
# cd saga-core-src/adaptors/

# ./generator/saga-adaptor-generator.pl -s ssh -n sshfs -t file -d .

suite:          ssh
type:           file
ftype:          sshfs_file
name:           sshfs
directory:      ./ssh/ssh_sshfs_file

copying files:   ...
fixing file names: .....
fixing files:    .....

You can now cd to ./ssh/ssh_sshfs_file,
and run 'make; make install'.

Note that you need to set SAGA_LOCATION before,
and point it to your SAGA installation tree.

# cd ssh/ssh_sshfs_file

# make
compiling  ssh_sshfs_file_adaptor.o
compiling  ssh_sshfs_file_dir_impl.o
compiling  ssh_sshfs_file_dir_nsdire_impl.o
compiling  ssh_sshfs_file_dir_nsentry_impl.o
compiling  ssh_sshfs_file_dir_perm_impl.o
compiling  ssh_sshfs_file_file_impl.o
compiling  ssh_sshfs_file_file_nsentry_impl.o
compiling  ssh_sshfs_file_file_perm_impl.o
```

```
liblinking    libsaga_adaptor_ssh_sshfs_file.so
liblinking    libsaga_adaptor_ssh_sshfs_file.a (static)

# make install
installing    lib
installing    lib (static)
installing    adaptor ini
```

When running any SAGA application, that adaptor will get loaded and will receive requests to perform remote operations. That can be confirmed by setting `SAGA_VERBOSE` in the application environment (see SAGA installation guide).

Of course, that adaptor will not be able to perform any meaningful operation – it is just a stub, and will simply throw `NotImplemented` exceptions for all calls. However, it is now straight forward to fill the stub with the respective functionality.

FIXME: Need to add details about class hierarchy, adaptor data, and instance data. Also, some details about adaptor makefiles, configure support etc might be useful.

Note that each adaptor will end up in a separate shared library. Since a typical SAGA installation will use multiple adaptors, and thus multiple adaptor libraries will be loaded into the same address space, the adaptor programmer needs to make sure to choose unique symbol names (i.e. to choose a unique and descriptive symbol name space), to avoid runtime symbol clashes.

Appendix