GWD-R.xx
SAGA-RG

Andre Merzky
Vrije Universiteit, Amsterdam

Version: 0.1

November 14, 2010

SAGA Resource Management API

Status of This Document

This document provides information to the grid community, proposing a standard for an extension to the Simple API for Grid Applications (SAGA). As such it depends upon the SAGA Core API Specification [1]. This document is supposed to be used as input to the definition of language specific bindings for this API extension, and as reference for implementors of these language bindings. Distribution of this document is unlimited.

Abstract

This document specifies a Resource Management API extension to the Simple API for Grid Applications (SAGA), a high level, application-oriented API for grid application development. This Resource Management (RM) API is motivated by a number of use cases collected by the OGF SAGA Research Group in GFD.70 [2], and by requirements derived from these use cases, as specified in GFD.71 [3]). It allows to interface to Grid resource discovery systems, and also allows to perform reservation upon discovered resources. The results of both discovery and information can be consumed by the SAGA Core job API package, to specify resources to submit jobs to.

Contents

1 Introduction

A significant number of SAGA use cases [2] ask for the possibility to poresistently store application level meta data. In difference to data storage in files, these meta data are usually small, and structured as key-value-pairs. The main use case for this API extension is that an application stores some state information, and that these state information are either used by other applications, or by a later running instance of the same application.

For example, an application which allows to stream data (i.e. uses the SAGA Stream API [1], may store the endpoint contact URL in an Advert, and another application which wants to connect to the first one may obtain this information from the advert service.

In order to simplify this particular (and other) use cases, the Advert API allows to store `saga::objects` in Adverts (i.e. these objects get serialized, stored, retreived, and deserialized). An application can thus store the `saga::stream` instance itself in an Advert, and can another application can retrieve this instance for connection setup. **FIXME: check**.

Adverts are defined as an entry in the Adverts name space, i.e. as an entry in an `saga::advert_dir`. Similar to `saga::logical_file` it has meta data attached (key-value pairs). As described above, an `saga::advert` can also have attached serialized `saga::objects`.

1.1 Notational Conventions

In structure, notation and conventions, this documents follows those of the SAGA Core API specification [1], unless noted otherwise.

1.2 Security Considerations

As the SAGA API is to be implemented on different types of Grid (and non-Grid) middleware, it does not specify a single security model, but rather provides hooks to interface to various security models – see the documentation of the `saga::context` class in the SAGA Core API specification [1] for details.

A SAGA implementation is considered secure if and only if it fully supports (i.e. implements) the security models of the middleware layers it builds upon, and neither provides any (intentional or unintentional) means to by-pass these security models, nor weakens these security models' policies in any way.

2 SAGA Advert API

2.1 Introduction

2.1.1 `saga::advert` and `saga::advert_dir` URLs

The `saga::advert` and `saga::advert_dir` URLs used in the Stream API [1].

2.1.2 Classes

The SAGA Advert API consists of two classes: a `advert` class, encapsulating information to be stored persitently; and a `advert_dir` class, representing the name space directories adverts are organized in.

2.2 Specification

```
package saga.adverts
{
  class advert : implements saga::ns_entry
                 implements saga::attributes
              // ...
  {
    CONSTRUCTOR  (in    session       session,
                  in    string        url,
                  out   msg           obj);
    DESTRUCTOR   (in    msg           obj);

    // Attributes:
    // extensible: yes
  }

  class advert_dir : implements   saga::ns_dir
                     implements   attributes
                   // ...
  {
    CONSTRUCTOR   (in    session       session,
                   in    string        url,
                   out   sender        obj);
    DESTRUCTOR    (in    sender        obj);
```

```
      // Attributes:
      //   extensible: yes
    }
  }
```

2.3 Specification Details

3 Intellectual Property Issues

3.1 Contributors

This document is the result of the joint efforts of several contributors. The authors listed here and on the title page are those committed to taking permanent stewardship for this document. They can be contacted in the future for inquiries about this document.

Andre Merzky
andre@merzky.net
Vrije Universiteit
Dept. of Computer Science
De Boelelaan 1083
1081HV Amsterdam
The Netherlands

The initial version of the presented SAGA API was drafted by members of the SAGA Research Group. Members of this group did not necessarily contribute text to the document, but did contribute to its current state. Additional to the authors listed above, we acknowledge the contribution of the following people, in alphabetical order:

Andrei Hutanu (LSU), Hartmut Kaiser (LSU), Pascal Kleijer (NEC), Thilo Kielmann (VU), Gregor von Laszewski (ANL), Shantenu Jha (LSU), and John Shalf (LBNL).

3.2 Intellectual Property Statement

The OGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the OGF Secretariat.

The OGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the OGF Executive Director.

3.3 Disclaimer

This document and the information contained herein is provided on an "As Is" basis and the OGF disclaims all warranties, express or implied, including but not limited to any warranty that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

3.4 Full Copyright Notice

FIXME: everything

References

[1] T. Goodale, S. Jha, H. Kaiser, T. Kielmann, P. Kleijer, A. Merzky, J. Shalf, and C. Smith. A Simple API for Grid Applications (SAGA). Grid Forum Document GFD.xx, 2007. Global Grid Forum.

[2] A. Merzky and S. Jha. A Collection of Use Cases for a Simple API for Grid Applications. Grid Forum Document GFD.70, 2006. Global Grid Forum.

[3] A. Merzky and S. Jha. A Requirements Analysis for a Simple API for Grid Applications. Grid Forum Document GFD.71, 2006. Global Grid Forum.