

SAGA API Extension: Advert 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). That extension provides access to persistent storage for application level meta data. As SAGA extension 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.

Copyright Notice

Copyright © Open Grid Forum (2007). All Rights Reserved.

Abstract

This document specifies an Adverts API extension to the Simple API for Grid Applications (SAGA), a high level, application-oriented API for grid application development. This Adverts 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 information systems, and, more importantly, allows to persistently store application specific meta data in a name space hierarchy.

Contents

1	Introduction	3
1.1	Notational Conventions	3
1.2	Security Considerations	3
2	SAGA Advert API	4
2.1	Introduction	4
2.2	Specification	4
2.3	Specification Details	5
3	Intellectual Property Issues	6
3.1	Contributors	6
3.2	Intellectual Property Statement	6
3.3	Disclaimer	7
3.4	Full Copyright Notice	7
	References	8

1 Introduction

A significant number of SAGA use cases [2] ask for the possibility to persistently 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 ??), may store the endpoint contact URL as an advert, and another application which wants to connect to the first one may obtain this information from the advert service. This allows, amongst others, for simple and environment independent bootstrapping of distributed ensembles of applications.

In order to simplify the use of the advert service, in particular for bootstrapping, the advert API allows to store `saga::objects` in adverts (i.e. these objects get serialized, stored, retrieved, and deserialized).

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

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 exact rendering of the advert namespace is up to the SAGA implementation, and how valid URLs are formed (i.e. what schemas are supported) is not specified in this document. Implementations SHOULD, however, strive to support the generic URL schema 'any'.

2.1.2 Classes

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

2.2 Specification

```
package saga.adverts
{
    class advert : extends          saga::ns_entry
                        extends      saga::attributes
                        // from ns_entry saga::object
                        // from ns_entry saga::async
                        // from ns_entry saga::permissions
                        // from object  saga::error_handler
    {
        CONSTRUCTOR      (in    session    session,
                          in    string      url,
                          out   msg         obj);
        DESTRUCTOR       (in    msg         obj);

        store_object     (in  saga::object obj);
        retrieve_object   (out saga::object obj);

        // Attributes (extensible):
    }

    class advert_directory : extends saga::ns_directory
```

```
        extends saga::attributes
        // from ns_directory saga::ns_entry
        // from ns_entry     saga::object
        // from ns_entry     saga::async
        // from ns_entry     saga::permissions
        // from object       saga::error_handler
    {
        CONSTRUCTOR    (in session      session,
                       in string      url,
                       out sender     obj);
        DESTRUCTOR     (in sender      obj);

        // find adverts based on name, object type, and meta data
        find            (in string      name_pattern,
                       in array<string> attr_pattern,
                       in saga::object_type type = 0,
                       in int          flags = Recursive,
                       out array<saga::url> names );

        // Attributes (extensible):
    }
}
```

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

Copyright (C) Open Grid Forum (2007). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the OGF or other organizations, except as needed for the purpose of developing Grid Recommendations in which case the procedures for copyrights defined in the OGF Document process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the OGF or its successors or assignees.

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.