The OpenMath-based protocol for symbolic software composability

University of St Andrews

Alexander Konovalov[1], Steve Linton[1], Dan Roozemond[2] {*alexk,sal*}*@cs.st-and.ac.uk, D.A.Roozemond@tue.nl*

1:School of Computer Science, University of St Andrews *2:Eindhoven University of Technology*

Technische Universiteit Eindhoven University of Technology

We address modern needs of symbolic computations:

- infrastructure for implementing parallel algorithms
- tools for combining capabilities available in several different systems
- deployment of symbolic computation resources, such as databases or specialized software as Web services or Grid services accessible over the Internet

SCIEnce – Symbolic Computation Infrastructure in Europe (http://www.symbolic-computation.org)

The **SCIEnce** project unites efforts of developers of computational algebra systems

- **GAP** (`http://www.gap-system.org`)
- **KANT/KASH** (`http://www.math.tu-berlin.de/~kant/kash.html`)
- **Maple** (`http://www.maplesoft.com`)
- **MuPAD** (`http://www.mupad.de`)

together with experts in parallel/distributed computing from the University of St Andrews, Heriot Watt University, Edinburgh and Institute e-Austria Timisoara, OpenMath experts from the Eindhoven University of Technology, and experts in symbolic computation from Research Institute for Symbolic Computation, Linz and Centre National de la Recherche Scientifique, Palaiseau.

SCSCP – Symbolic Computation Software Composability Protocol:

We are working towards providing an OpenMath-based protocol for inter-CAS communication by which a computer algebra system (CAS) may offer services for the following clients:

- a Web server which passes on the same services as Web services using SOAP/HTTP protocols to a variety of possible clients;
- Grid middleware;
- another instance of the same CAS (in a parallel computing context);
- another CAS running on the same computer system or remotely.

As well as transmission of actual mathematical objects, we support transmission of references, which can be used in subsequent requests. For example, we envisage the possibility to construct and manipulate large mathematical objects on remote grids with sending over the Internet only the properties the user is interested in. Proofs of concept are available in KANT/KASH and GAP.

SCSCP specifies:

Semantical and technical descriptions of OpenMath-encoded messages to and from CAS:

- remote procedure call
- returning result if the procedure was completed successfully
- returning a signal about procedure termination

and also allowed sequences of these messages.

cascall1 (CAS call) - OpenMath Content Dictionary:

defines OpenMath symbols for the description of the management of mathematical queries:

- three main kinds of messages:
  - `procedure_call`, `procedure_completed`, `procedure_terminated`
- options that may be added to the `procedure_call` message:
  - `option_runtime`
  - `option_debuglevel`
  - `option_min_memory`, `option_max_memory`
  - `option_return_object`, `option_return_cookie`
- information that may be supplied with the result:
  - `info_runtime`, `info_memory`
  - `cookie`
- standard errors
  - `error_runtime`, `error_memory`
  - `error_system_specific`

Example of `procedure_call` message

The procedure `GroupIdentificationService` accepts the list of permutations and returns the catalogue number in the GAP Small Groups Library of the group they generate (unspecified options will take their default values):

```
<OMOBJ>
    <OMATTR>
        <OMATP>
            <OMS cd="cascall1" name="call_ID"/>
            <OMSTR>2</OMSTR>
            <OMS cd="cascall1" name="option_runtime" />
            <OMI>300000</OMI>
        </OMATP>
        <OMA>
            <OMS cd="cascall1" name="procedure_call"/>
            <OMSTR>GroupIdentificationService</OMSTR>
            <OMA>
                <OMS cd="list1" name="list"/>
                <OMA>
                    <OMS cd="permut1" name="Permutation"/>
                    <OMI> 2</OMI>
                    <OMI> 3</OMI>
                    <OMI> 1</OMI>
                </OMA>
                <OMA>
                    <OMS cd="permut1" name="Permutation"/>
                    <OMI> 2</OMI>
                    <OMI> 1</OMI>
                    </OMA>
                </OMA>
            </OMA>
    </OMATTR>
</OMOBJ>
```

Example of `procedure_completed` message

`GroupIdentificationService` evaluated the `procedure_call` message and determined that the permutation group has the number [6,1]:

```
<OMOBJ>
    <OMATP>
        <OMS cd="cascall1" name="call_ID"/>
        <OMSTR>2</OMSTR>
    </OMATP>
    <OMA>
        <OMS cd="cascall1" name="procedure_completed"/>
        <OMA>
            <OMS cd="linalg2" name="vector"/>
            <OMI> 6</OMI>
            <OMI> 1</OMI>
        </OMA>
    </OMA>
</OMOBJ>
```

GAP implementation of SCSCP client and server:

- communicating using sockets via TCP/IP protocol
- parsing OpenMath code supports:
  - `cascall1` OM CD
  - a number of symbols from other CDs
  - OM attributes (OMATTR, OMATP)
  - OM references (OMR) to work with remote objects
- user-level functionality:
  - installing procedures available as SCSCP services
  - running the SCSCP server
  - sending request to the server and getting result
  - procedures STORE and RETRIEVE to work with remote objects

GAP session example with communication by SCSCP:

The user access the SCSCP procedure `GroupIdentificationService`, installed on the remote server, using the user's function `IdGroupWS`. The background OpenMath-encoded communication between CASs is hidden from the user.

```
gap> IdGroupWS(SymmetricGroup(3));
#I  Creating a socket ...
#I  Connecting to a remote socket via TCP/IP ...
#I  Got connection initiation message PROTOCOL_X_VERSION 0 CAS_PID 2
#I  Request sent, waiting for reply ...
#I  Got back: object [ 6, 1 ] with attributes [ [ "call_ID", "2" ] ]
[ 6, 1 ]
gap>
```

SCIENCE Symbolic Computation Infrastructure for Europe

The project **SCIEnce** – Symbolic Computation Infrastructure in Europe (http://www.symbolic-computation.org) is supported by the EU Framework VI programme grant RII3-CT-2005-026133.

References:

A. Konovalov, S. Linton. *Symbolic Computation Software Composability Protocol Specification*. CIRCA preprint 2007/5, University of St Andrews (`http://www-circa.mcs.st-and.ac.uk/pre-prints.html`).

OpenMath (`http://www.openmath.org`).

D. Roozemond. *OpenMath Content Dictionary* `cascall1` (`http://www.win.tue.nl/SCIEnce/cds/cascall1.html`).