Interoperability of a Collaborative Online Visualization and Steering (COVS) Framework using VISIT

Morris Riedel, Wolfgang Frings, Sonja Dominiczak, Daniel Mallmann, Thomas Eickermann, Thomas Düssel, Paul Gibbon, Rainer Spurzem
Joint work of FZJ expertise

People with different expertise work together for COVS

COVS

Morris Riedel
Daniel Mallmann
Wolfgang Frings
Sonja Dominiczak
Thomas Düssel
Thomas Eickermann
Paul Gibbon
PEPC Application

Distributed Systems and Grid Computing
Capability Computing and Visualization
Communication Systems
Computer Simulations

FZJ department: Central Institute of Applied Mathematics (ZAM)
Outline

- COVS Framework
- COVS with UNICORE&VISIT
  - Architectural Design
  - Proof-of-concept Implementation
- COVS and VISIT Steering
- Lessons learned
- Hot Issue: COVS with UNICORE & Globus in AstroGrid-D
- Demand for Standardization (and thus interoperability)
- Summary
- Acknowledgements
COVS Framework
What is COVS?

- **Collaborative Online Visualization and Steering** (COVS)
  - Tool for analyzing and better understanding of parallel applications that run on a supercomputer or cluster
  - Visualize complex scientific datasets (vectors, arrays,…) (schematic representations, non-photo-realistic representations)

- Visualization NOT after computation
  - But during its computation (online) (not post-processing)
  - More insights in the computing process of the application
  - Allows a wider range of control through steering of the application (Influence the behavior of a scientific application while running)

- COVS allows…
  - …geographically dispersed participants to steer applications
  - …to seamlessly run parallel applications and share visualizations
First ideas of the architecture...

UniGrids

OK!
Easy going then!
(But is it really so easy?)
COVS Framework Design

• Core Building blocks of the COVS Framework (SSH-based)

• Re-usable for any visualization and parallel application (simulation) that base upon the same communication library

• Goal: User does not have to deal with hostnames, usernames or passwords
COVS Framework & SSH

- Several steps that provide an SSH connection within the Grid
  - Grid middleware for establishment is crucial for end-users transparency
  - no hostnames/ usernames

- Using simple SSH mechanisms
  - authorized_keys mechanism
  - public key exchange

- SSH port has to be open
Proof-of-concept Implementation
Projects and infrastructures that use **UNICORE**

Core middleware of

One of the three middleware systems of

One of the four major middleware systems of
COVS in context of UNICORE 6

- COVS Grid services
- Just one Higher-level service within UNICORE 6
- WS-RF compliant (OASIS Standard)
- Connection & Session Management
Communication Library VISIT

- Visualization Interface Toolkit (VISIT)
  - Bi-directional data transfer of scientific and steering data
  - Components that provide collaborative visualizations (e.g. Multiplexer)
- (Unique) design of VISIT: The visualization acts as a server
  - Prevent failures or slow operations of the visualization from the simulation progress
COVS with UNICORE & VISIT
Steering Workshop OGF20 7th May 2007

Proof-of-concept

- Grid Middleware
- Intel GPE
- UNICORE Client
- VISIT Communication Library
Managing Sessions (1)

- Manage COVS Sessions
- Join or Create Sessions
- Monitor Session status
Managing Sessions (2)

- Connect/disconnect participant
- Monitor session status
- Monitor conn. perform.

![GPE Client - COVS](image)

- Session name: Nobdy-with nobdy session, Morris, 2007-02-10
- Participants: 2
- Comments: Today's visualization session analyzing in particular the im
- Starttime: 2007-02-10, 8:35:07 CET
- Session role: Master, Participant

**Participants**
- Morris
  - Status: Connected
  - Roles: Master, Participant
  - Bandwidth: 56.738 [Mb/s]
  - Message size: 512,000 [B/s]
  - Number of messages: 58
- Wolfgang
  - Status: Connected
  - Roles: Participant, Steerer, Collaborator
  - Bandwidth: 55.670 [Mb/s]
  - Message size: 512,000 [B/s]
  - Number of messages: 58

**Simulation**
- Comments: Nobdy64 + simulation, 256 cpu application

**Start Session**
- Abort Session
- Delete Session
- Connect Participant
- Disconnect Participant
- Pause Simulation
- Continue Simulation
- Request Steerer Role
- Request Collaborator Role
- Refresh
- Abort
COVS Framework Use-case

GPE Grid Client

COVS Grid Bean

Security & Contact Information Exchange via a simple protocol over named pipes

Xnbody Scientific Visualization

Visualization Technology VTK

VISIT server

client tier

job submit & information management exchange

scientific data & steering commands transfer

firewall

server tier

COVS Grid Services

UNICORE

Job Control

RMS

PEPC Parallel Simulation

VISIT Client

target system tier
COVS and VISIT Steering
COVS and VISIT Steering (1)

- Design idea: Simulation acts as a client
  - Not disturb progress of simulation (highest priority in HPC)
  - VISIT server is instrumented in the code of visualization
  - VISIT client is instrumented in the code of simulation
- VISIT-enabled visualizations (e.g. Xnbody)
  - Provide steering parameters in the GUI of the end-users
  - (e.g. positions of black hole in relation to stars)
- VISIT collects the steering parameters from visualizations
  - Proprietary VISIT protocol, but integers, floats, etc. possible
  - Steering parameters affect the run-time of the application
- Issue: Steering parameters different in applications
COVS and VISIT Steering (2)

... conn = commlib_connect(endpoint, ...);
...
while (SimTime)
{
  ...
  /* id = 2, dim = 1 */
  commlib_recv(conn, 2, SimTime,
               &parm, DATATYPE_INT, 1);
  work(...);
  /* id = 1, dim = 3 */
  commlib_send(conn, 1, SimTime,
               data, DATATYPE_DOUBLE, 3,
               nx, ny, nz);
} ...

commlib_disconnect(conn);
...
Lessons Learned & ‘Hot Issues’
Lessons learned

• Easier installation of VISIT libraries
  • Installation via SSH accounts not always possible (DEISA)
• Sometimes troubles using SSH
  • In some Grids, SSH port and access is totally closed or n/a
    • (e.g. German Weather forecast service - DWD)
  • Potential solutions: e.g. gLogin system from GUP, Linz
• Network bandwidth extremely important
  • Requirements for advance network reservation
  • Potential solutions: e.g. VIOLA Metascheduler
• Simulations have to be computed at a specific time(!)
  • Requirements for advance reservation of computational time
  • Potential solutions: e.g. VIOLA Metascheduler
Hot Issue: AstroGrid-D (1)

- D-Grid is the German national Grid infrastructure
  - AstroGrid-D is the astrophysics community of D-Grid
- State-of-the art
  - Some within AstroGrid-D used UNICORE/VISIT until now
  - Steer with COVS framework the nbody6 (parallel) simulation
    - e.g. Rainer Spurzem, Andreas Ernst (University of Heidelberg)
- Hot Issues:
  - Some clusters use UNICORE – some clusters use Globus 4
  - Some scientists work with UNICORE – some use Globus 4
  - Both UNICORE and Globus 4 are not interoperable for COVS
- Requirement for Interoperability of COVS services
  - Interoperable COVS services available in all middlewares
Hot Issue: AstroGrid-D (2)

- nbody6 simulation with Xnbody visualization
Demand for Standardization

- No existing Open Grid Standards in the area of COVS
  - VISIT contributed use cases to SAGA-WG
  - An overall standard (session management, etc.) missing
- Push of interoperability
  - COVS services within Grid middleware (deployed by default)
  - COVS services available in UNICORE, Globus, gLite, …
- Contribute to SAGA-WG
  - Initial interactions, so far no ‘focused work’ in COVS area
  - SAGA Steering APIs or Standards good starting point
- At least from UNICORE/VISIT perspective unfunded efforts
  - Plans to put COVS into Proposal for an EU call (get funding)
Summary and Acknowledgements
Summary

• COVS Framework implementation with UNICORE / VISIT
  • Stable and will be ready for production usage in UNICORE 6
  • Betas successfully demonstrated at numerous places
    • EuroPar’06, Supercomputing 2006, OGF18+19, DEISA Trainings
    • A detailed COVS thesis is available for those that are interested
• Requirements for standardization and interoperability
  • E.g. AstroGrid-D community needs UNICORE & Globus 4
    • Steering of nbody6 parallel simulation with Xnbody visualization
  • UNICORE, Globus, (+gLite) lack interoperable COVS services
  • There is no widely accepted common COVS framework
• Grid visualization community closer together (collaborate!)
  • Contribute/Develop (to) standards in this area…
Acknowledgements

• Open Middleware Infrastructure Institute for Europe

• Participation in this workshop is partially funded by the OMII – Europe project under EC grant RIO31844-OMII-EUROPE, duration May 2006 - April 2008

• ZAM/NIC of Forschungszentrum Jülich (FZJ) in the HELMHOLTZ association

Forschungszentrum Jülich
in der Helmholtz-Gesellschaft