

SAGA: A Producer and Consumer of Standards for Distributed Computing

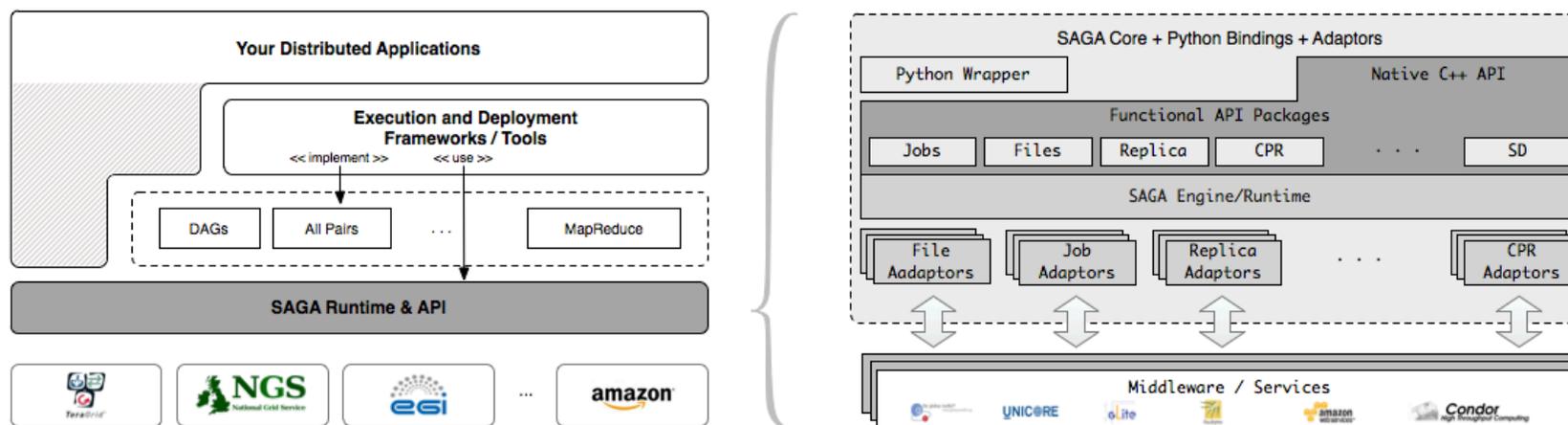
Shantenu Jha, Andre Merzky

<http://saga-project.org>

MAGIC, 01 Feb, 2012

What is SAGA?

- ▣ Simple, integrated, stable, uniform and community-standard
 - Simple and Stable: 80:20 restricted scope
 - Integrated: Similar semantics & style across primary functional areas
 - Uniform: Same interface for different distributed systems
 - The building blocks upon which to construct “consistent” higher-levels of functionality and abstractions
 - OGF-standard, “official” Access Layer/API of EGI, NSF-XD



SAGA: Consumer of Standards

- **Component: saga-core implements**
 - SAGA Core API Specification" - OGF, GFD.90
 - SAGA Advert API Extension" - OGF, GFD.177
 - SAGA Service Discovery API Extension" - OGF, GFD.144
 - SAGA Information Service Navigator API Extension - OGF, final draft
- **Component: saga-adaptor-globus uses**
 - gridftp - OGF, GFD.20
 - GSS-API - OGF, GFD.24
 - X509 - ITU-T, IETF, others
- **Component: saga-adaptor-bes uses**
 - Basic Execution Service / HPC Basic Profile - OGF, GFD.114
 - JSDL - OGF, GFD.136
 - JSDL HPC - OGF, GFD.111
 - JSDL SPMD - OGF, GFD.115
 - X509 - ITU-T, IETF, others

SAGA: The Standard Experience

- **SAGA: The standard has helped in end-to-end integration**
 - SAGA on every major Production CI, eg XSEDE, EGI, OSG
 - Academic and non-Academic, e.g, Airbus, BT
- **Standards as an *important strategy* for sustainability**
 - Service Discovery: We helped define the API (OGF), package developed by RAL(Fisher), gLite adaptors and our implementation used by CERN, now extending to KEK/NAREGI
 - Other examples of implementations and adaptor sharing..
- **Standards as one possible route for interoperability**
 - E.g., SAGA deployed and used on FutureGrid, as an access layer into many major middleware systems; simplify access to every major Production CI without AA....
 - <http://www.saga-project.org/interop-demos>

Standards Road-Map

- ▣ Focus has been on simple job/task management/data standards have a role to play in higher-level abstractions
- ▣ Pilot-Jobs as one of the most successful distributed computing abstractions
 - With more than a dozen plus PJ systems out there, need to void situation analagous to Workflow Systems
 - After defining an underlying model of pilot-jobs (P^*) can define a “well defined API” to Pilot-Jobs (called Pilot-API)
- ▣ Pilot-API is now part of (emerging) OGF/SAGA resource-package, where we find semantic commonality with OCCl