

# SAGA Session - I

S Jha, T Kielmann, A Merzky

# OGF IPR Policies Apply

- “I acknowledge that participation in this meeting is subject to the OGF Intellectual Property Policy.”
- Intellectual Property Notices Note Well: All statements related to the activities of the OGF and addressed to the OGF are subject to all provisions of Appendix B of GFD-C.1, which grants to the OGF and its participants certain licenses and rights in such statements. Such statements include verbal statements in OGF meetings, as well as written and electronic communications made at any time or place, which are addressed to:
  - the OGF plenary session,
  - any OGF working group or portion thereof,
  - the OGF Board of Directors, the GFSG, or any member thereof on behalf of the OGF,
  - the ADCOM, or any member thereof on behalf of the ADCOM,
  - any OGF mailing list, including any group list, or any other list functioning under OGF auspices,
  - the OGF Editor or the document authoring and review process
- Statements made outside of a OGF meeting, mailing list or other function, that are clearly not intended to be input to an OGF activity, group or function, are not subject to these provisions.
- Excerpt from Appendix B of GFD-C.1: “Where the OGF knows of rights, or claimed rights, the OGF secretariat shall attempt to obtain from the claimant of such rights, a written assurance that upon approval by the GFSG of the relevant OGF document(s), any party will be able to obtain the right to implement, use and distribute the technology or works when implementing, using or distributing technology based upon the specific specification(s) under openly specified, reasonable, non-discriminatory terms. The working group or research group proposing the use of the technology with respect to which the proprietary rights are claimed may assist the OGF secretariat in this effort. The results of this procedure shall not affect advancement of document, except that the GFSG may defer approval where a delay may facilitate the obtaining of such assurances. The results will, however, be recorded by the OGF Secretariat, and made available. The GFSG may also direct that a summary of the results be included in any GFD published containing the specification.”
- OGF Intellectual Property Policies are adapted from the IETF Intellectual Property Policies that support the Internet Standards Process.

# Agenda

- SAGA: Introduction (20+ min)
- Presentation by Sylvain Reynaud, IN2P3, France
- Open Discussion
  - Current SAGA efforts
  - Community Needs:
    - Adaptors
    - Deployment, Language Bindings..
  - Updates on other SAGA projects

# SAGA: In a Nutshell

- Applications must be able to utilize Infrastructure
  - “e-Infrastructure that will enable novel and different research” linked with e-Science
- But (how) can applications be developed to utilize complex, dynamic – qualitative and quantitative – infrastructure?

# SAGA: A Quick Tour



- There exists a lack of:
  - Programming interfaces that provide common grid functionality at the correct level of abstraction for applications
  - Ability to hide underlying complexity of infrastructure, varying semantics, heterogeneities and changes from the application-developer
- **Simple, integrated, stable, uniform** and **high-level** interface
  - Simple: 80:20 restricted scope
  - Integrated: Similar semantics & style across commonly used distributed functional requirements
  - Stable: Standardization
  - Uniform: Same interface for different distributed systems
- SAGA: Provides the high-level abstraction that application developers need that will work across different distributed systems
  - Shields details of lower level middle-ware and system issues
  - Enables the details of distribution to be left out

# Copy a File “before SAGA”: Globus GASS



```
int copy_file (char const* source, char const* target)
{
  globus_url_t          source_url;
  globus_io_handle_t    dest_io_handle;
  globus_ftp_client_operationattr_t source_ftp_attr;
  globus_result_t       result;
  globus_gass_transfer_requestattr_t source_gass_attr;
  globus_gass_copy_attr_t source_gass_copy_attr;
  globus_gass_copy_handle_t gass_copy_handle;
  globus_gass_copy_handleattr_t gass_copy_handleattr;
  globus_ftp_client_handleattr_t ftp_handleattr;
  globus_io_attr_t        io_attr;
  int                     output_file = -1;

  if ( globus_url_parse (source_URL, &source_url) != GLOBUS_SUCCESS ) {
    printf ("can not parse source_URL \"%s\"\n", source_URL);
    return (-1);
  }

  if ( source_url.scheme_type != GLOBUS_URL_SCHEME_GSIFTP &&
        source_url.scheme_type != GLOBUS_URL_SCHEME_FTP &&
        source_url.scheme_type != GLOBUS_URL_SCHEME_HTTP &&
        source_url.scheme_type != GLOBUS_URL_SCHEME_HTTPS ) {
    printf ("can not copy from %s - wrong prot\n", source_URL);
    return (-1);
  }

  globus_gass_copy_handleattr_init (&gass_copy_handleattr;
  globus_gass_copy_attr_init (&source_gass_copy_attr);

  globus_ftp_client_handleattr_init (&ftp_handleattr;
  globus_io_fileattr_init (&io_attr);

  globus_gass_copy_attr_set_io (&source_gass_copy_attr, &io_attr);
  globus_gass_copy_attr_set_io (&io_attr);
  globus_gass_copy_handleattr_set_ftp_attr (&gass_copy_handleattr,
  globus_gass_copy_handleattr_set_ftp_attr (&ftp_handleattr);
  globus_gass_copy_handle_init (&gass_copy_handle,
  globus_gass_copy_handleattr (&gass_copy_handleattr);
```

```
if (source_url.scheme_type == GLOBUS_URL_SCHEME_GSIFTP ||
    source_url.scheme_type == GLOBUS_URL_SCHEME_FTP ) {
  globus_ftp_client_operationattr_init (&source_ftp_attr);
  globus_gass_copy_attr_set_ftp (&source_gass_copy_attr,
  globus_gass_copy_attr (&source_ftp_attr);
}
else {
  globus_gass_transfer_requestattr_init (&source_gass_attr,
  globus_gass_attr (&source_gass_attr);
  globus_gass_copy_attr_set_gass (&source_gass_copy_attr,
  globus_gass_copy_attr (&source_gass_attr);
}

output_file = globus_libc_open ((char*) target,
  O_WRONLY | O_TRUNC | O_CREAT,
  S_IRUSR | S_IWUSR | S_IRGRP |
  S_IWGRP);

if ( output_file == -1 ) {
  printf ("could not open the file \"%s\"\n", target);
  return (-1);
}

/* convert stdout to be a globus_io_handle */
if ( globus_io_file_posix_convert (output_file, 0,
  globus_io_handle (&dest_io_handle)

  != GLOBUS_SUCCESS) {
  printf ("Error converting the file handle\n");
  return (-1);
}

result = globus_gass_copy_register_url_to_handle (
  &gass_copy_handle, (char*)source_URL,
  &source_gass_copy_attr, &dest_io_handle,
  my_callback, NULL);
if ( result != GLOBUS_SUCCESS ) {
  printf ("error: %s\n", globus_object_printable_to_string
  (globus_error_get (result)));
  return (-1);
}
globus_url_destroy (&source_url);
return (0);
}
```

# SAGA Example: Copy a File

## High-level, uniform



```
#include <string>
#include <saga/saga.hpp>

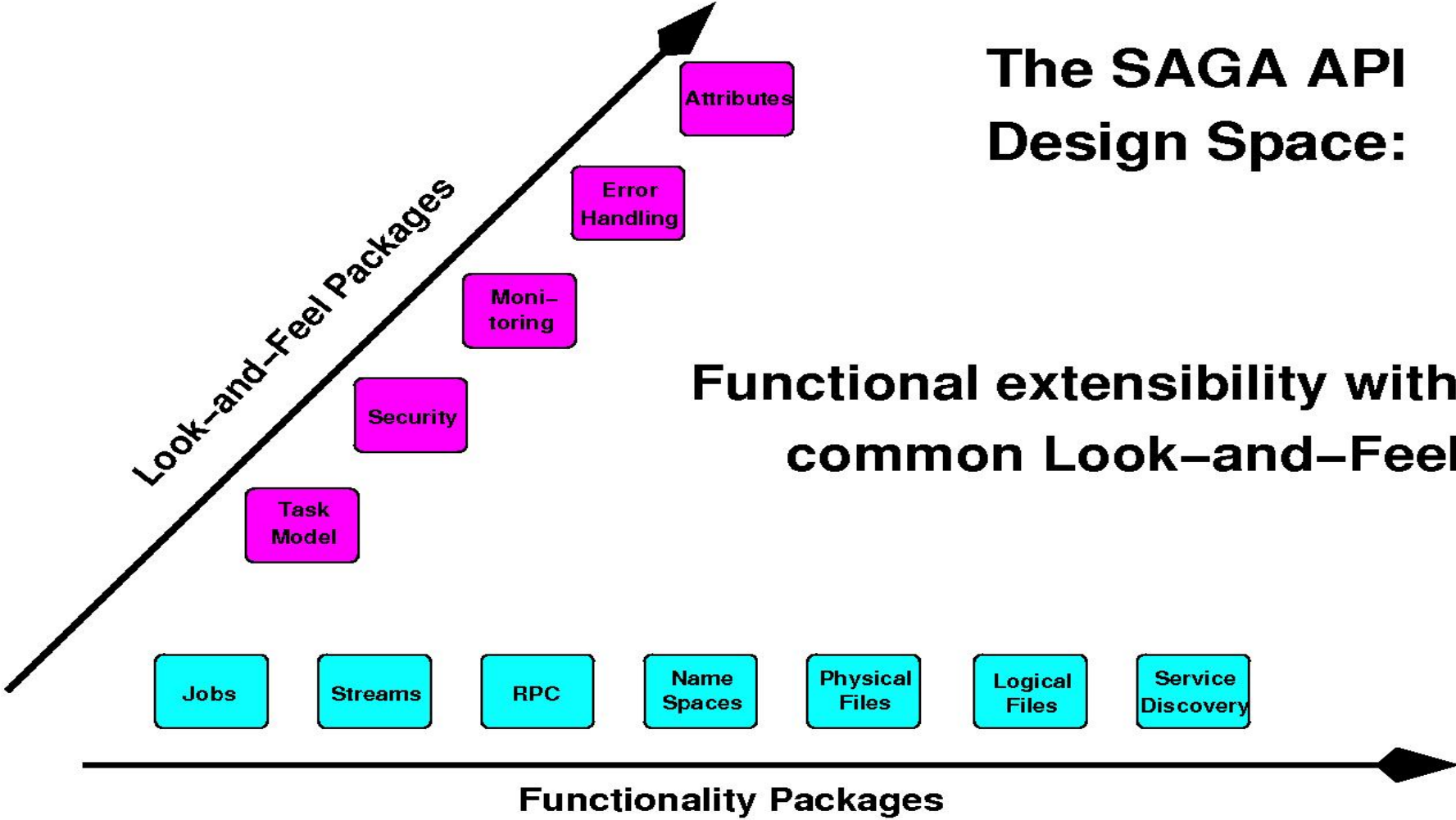
void copy_file(std::string source_url, std::string target_url)
{
    try {
        saga::file f(source_url);
        f.copy(target_url);
    }
    catch (saga::exception const &e) {
        std::cerr << e.what() << std::endl;
    }
}
```

# SAGA Interface

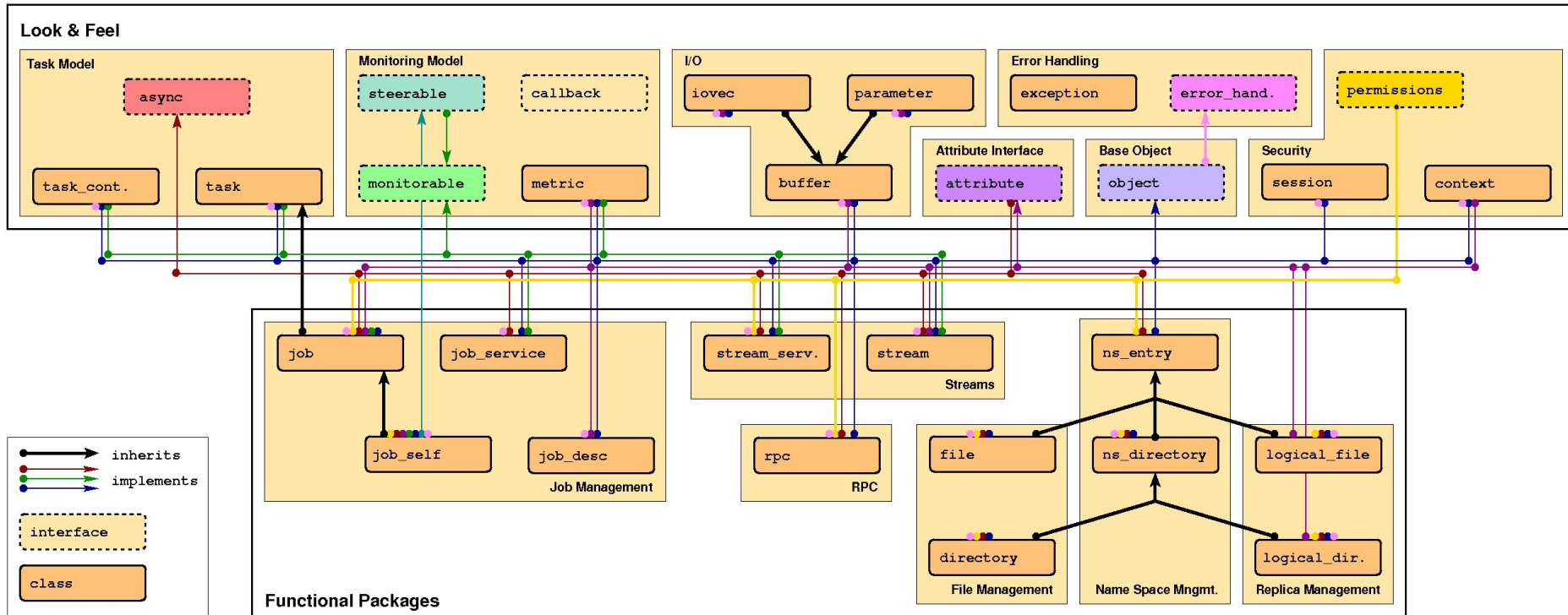
## Job Submission API

```
01: // Submitting a simple job and wait for completion
02: //
03: saga::job_description jobdef;
04: jobdef.set_attribute ("Executable", "job.sh");
05:
06: saga::job_service js;
07: saga::job job = js.create_job ("remote.host.net", jobdef);
08:
09: job.run();
10:
11: while( job.get_state() == saga::job::Running )
12: {
13:     std::cout << "Job running with ID: "
14:               << job.get_attribute("JobID") << std::endl;
15:     sleep(1);
16: }
```

# SAGA API Design Overview



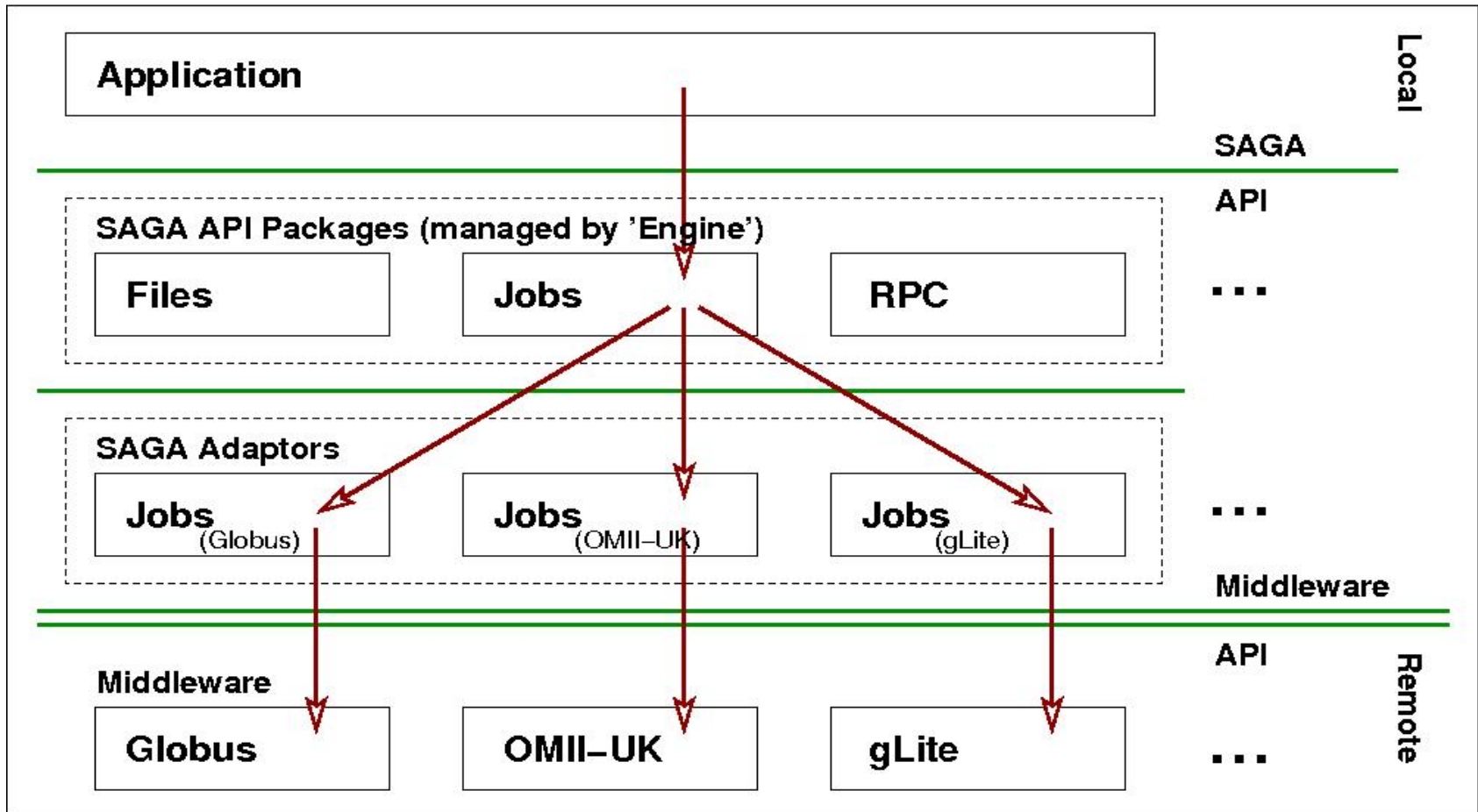
# SAGA: Class Diagram



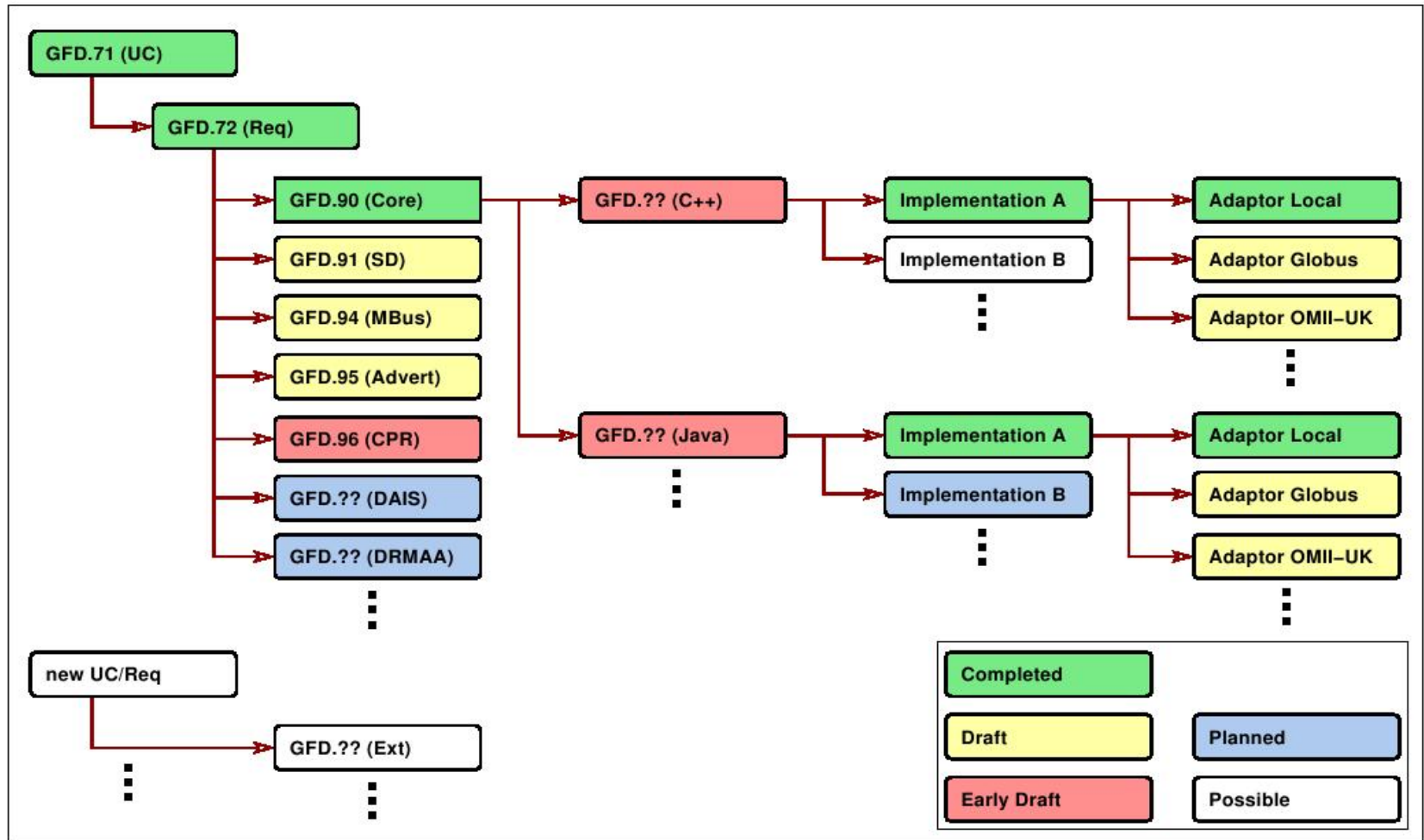
In the works: CPR, Information Services, Service Discovery, Messaging....

# Typical(?) SAGA Implementation

## Role of Adaptors (middleware binding)



# SAGA: The Landscape



# Let's move on...

- Talk about the IN2P3 implementation (Sylvain Reynaud)
- Discussion about (implementation) status and future activities (Shantenu Jha)