



GAT and the GAT API Design and Overview

Tom Goodale
goodale@cct.lsu.edu

LSU Center for
Computation and
Technology



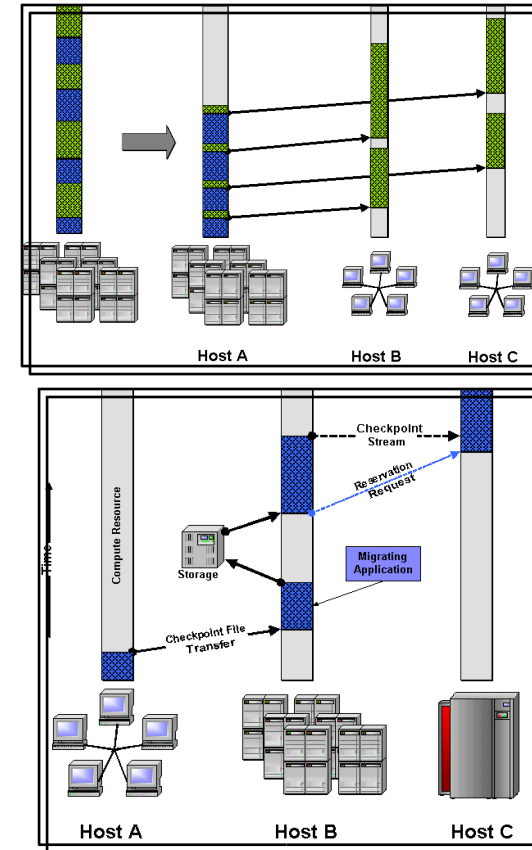
EU Funded by 5th Framework (January 2002 until April 2005):

PSNC/AEI/ZIB/MASARYK/
SZTAKI/ISUFI/Cardiff/VU/NTUA
Chicago/ISI/Wisconsin/Sun/Compaq

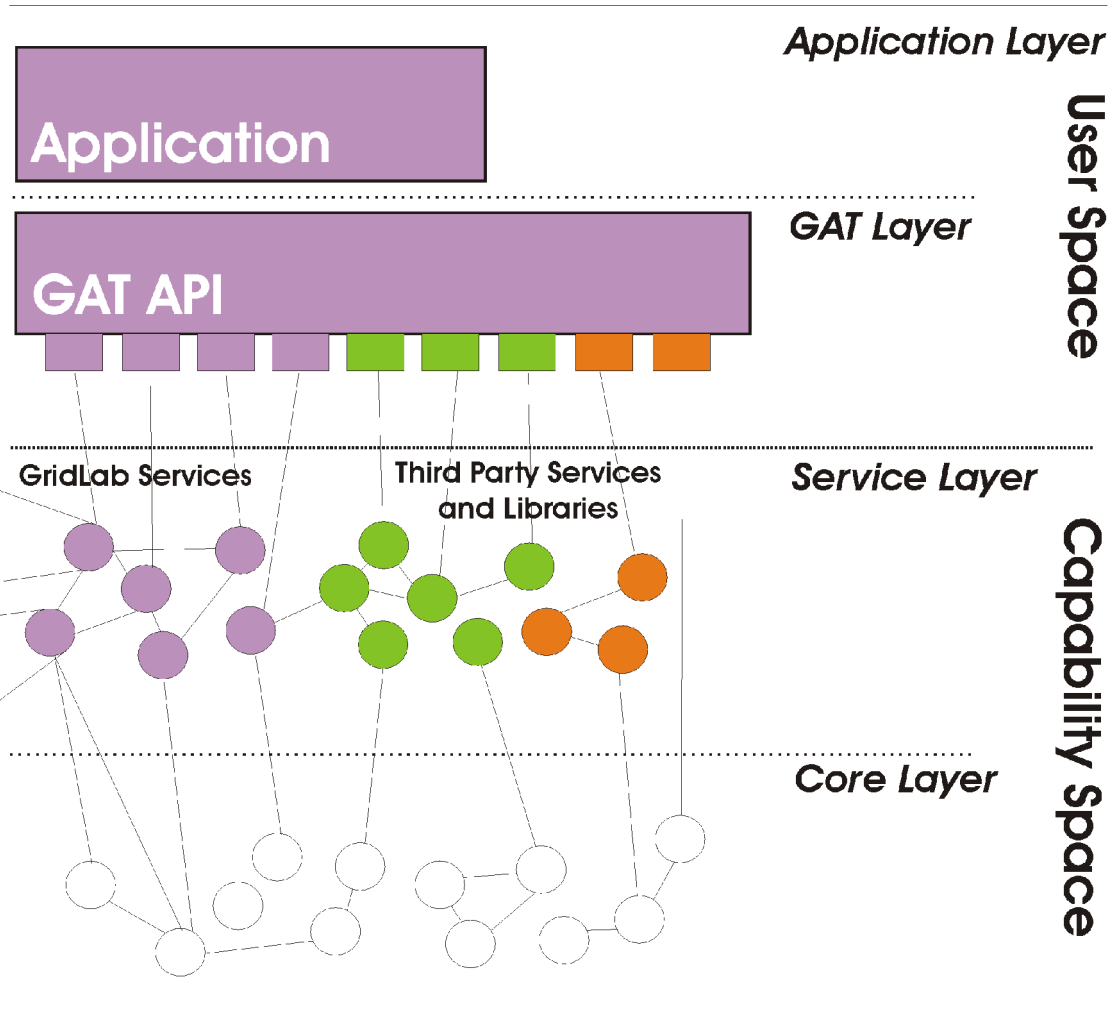
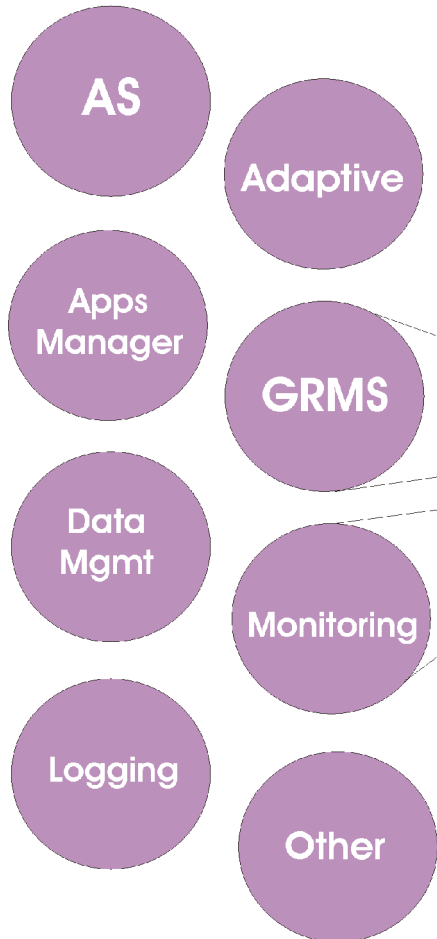
12 Work Packages covering:

- Grid Portals
- Mobile Users
- Different Grid Services
- Applications
- (Development) Test Bed

Grid Application Toolkit (GAT)



GridLab Services



- GridLab is developing many services which live in the layer between the GAT and lower-level Grid middleware, such as Globus, or on top of operating-system services.
 - Resource Brokering
 - Data Management
 - Monitoring
 - Information
 - Adaptive Components
 - Mobile users
 - Security
- Most are Web or Grid Services.

- Globus provides GRAM, which allows remote submission of jobs.
- Need a layer on top of this which intelligently picks the correct resource for a given jobs based upon the characteristics of the job, the user's desires for completion time, cost, etc, and the state of the resources.

- Simulations generally produce large quantities of data.
- Need a service, or set of services which manage this data
 - Replicating it to where it is needed,
 - Cataloguing it
 - Providing remote access
 - Transparent, secure copying of data between resources
 - Etc
- Also provides mechanisms facilitating remote visualisation of data sets.

- In order to make intelligent decisions about the state of resources or applications, we need to monitor them.
 - Network performance
 - Application performance
 - Resource load
 - ...

- Information service builds upon the Globus MDS to provide GridLab-specific schemas and registration web services.
- Adaptive components archives and analyses monitoring and other data in order to provide models of the behaviour of the system (network, applications, resources) which can be used to make intelligent decisions.
- Mobile users service provides access to SMS and other notification processes which can be used to support people on the move.
- Security provides mechanisms for authentication authorisation to GridLab services and resources.

GAT: Grid Application Toolkit

- API and Toolkit for developing portable Grid applications independently of the underlying Grid infrastructure and available services
 - Implements the GAT-API
 - Used by applications (different languages)
- GAT Adaptors
 - Connect to capabilities/services
- GAT Engine
 - Provides the function bindings for the GAT-API

Application

'Is there a better resource I could be using?'

SOAP

WSDL

CORBA

OGSA

Other

Monitoring

Security

Profiling

Information

Logging

Notification

*Data
Management*

*Resource
Management*

*Application
Manager*

Migration

GLOBUS

UNICORE

*Other Grid
Infrastructure?*

The Same Application ...

Laptop

Application

GAT



No network!

Super Computer

Application

GAT



Firewall issues!

The Grid

Application

GAT



Application

'Is there a better resource I could be using?'

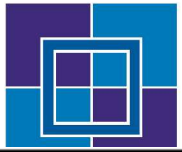
newmachine = GAT_Resource.Find()



The Grid

- Application makes GAT API calls for operations which may be Grid-related.
- Application links against the GAT Engine
- Application runs irrespective of actual underlying infrastructure deployment
 - Engine loads adaptors which are valid in the environment extant when the application starts
 - Adaptors try to do Grid operations on request, on failure another adaptor provided function may be called.
- Application can thus be compiled, linked and tested without any Grid services
- Same application executable can run in a full Grid environment

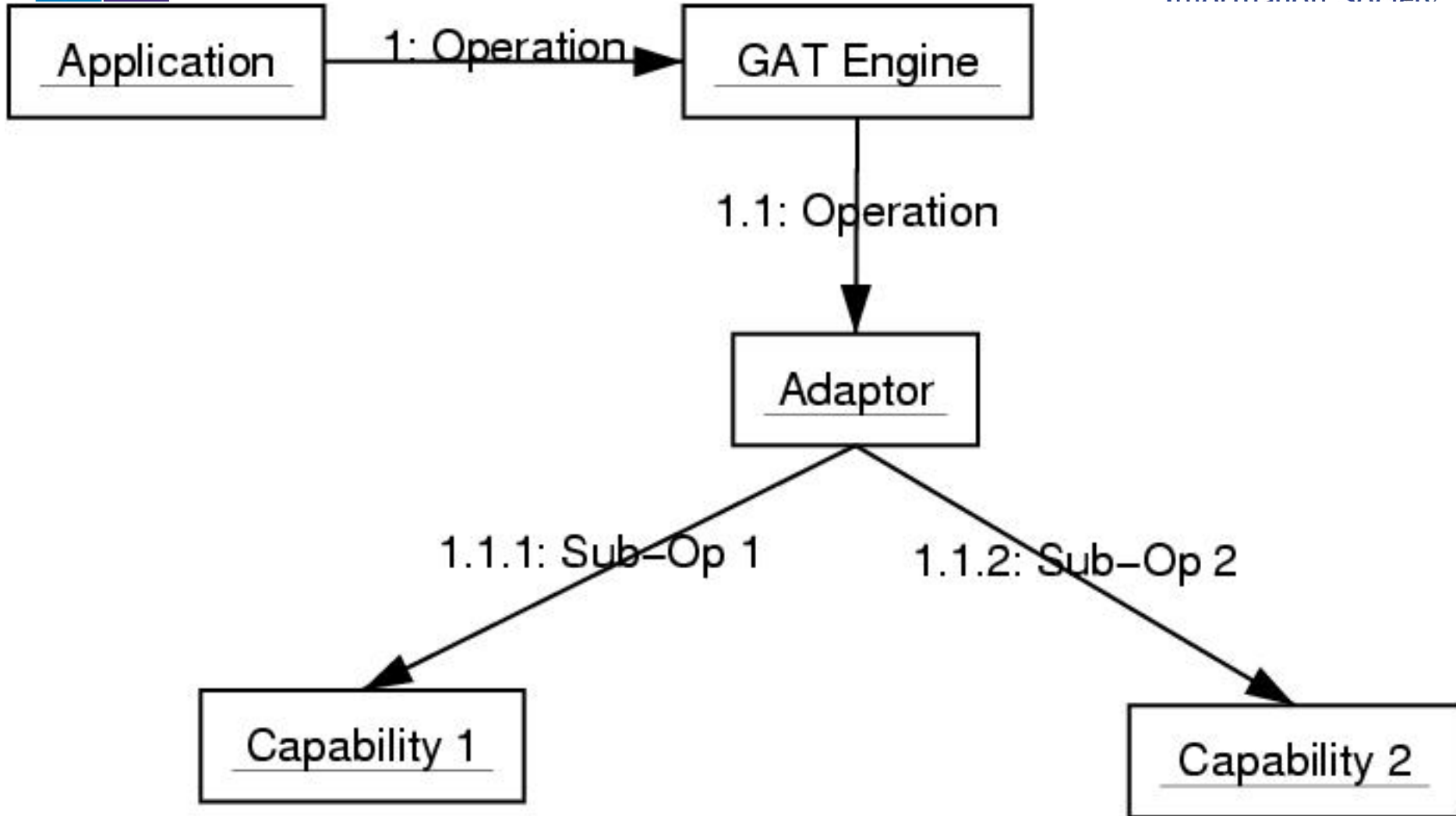
- The GAT uses whatever underlying Grid infrastructure there is and that people have developed adaptors for.
- GAT is not about replacing already developed infrastructure, but instead to provide a simple, clear interface which can be used with many different infrastructures.
 - Different versions of Globus
 - Condor
 - Unicore
 - ...



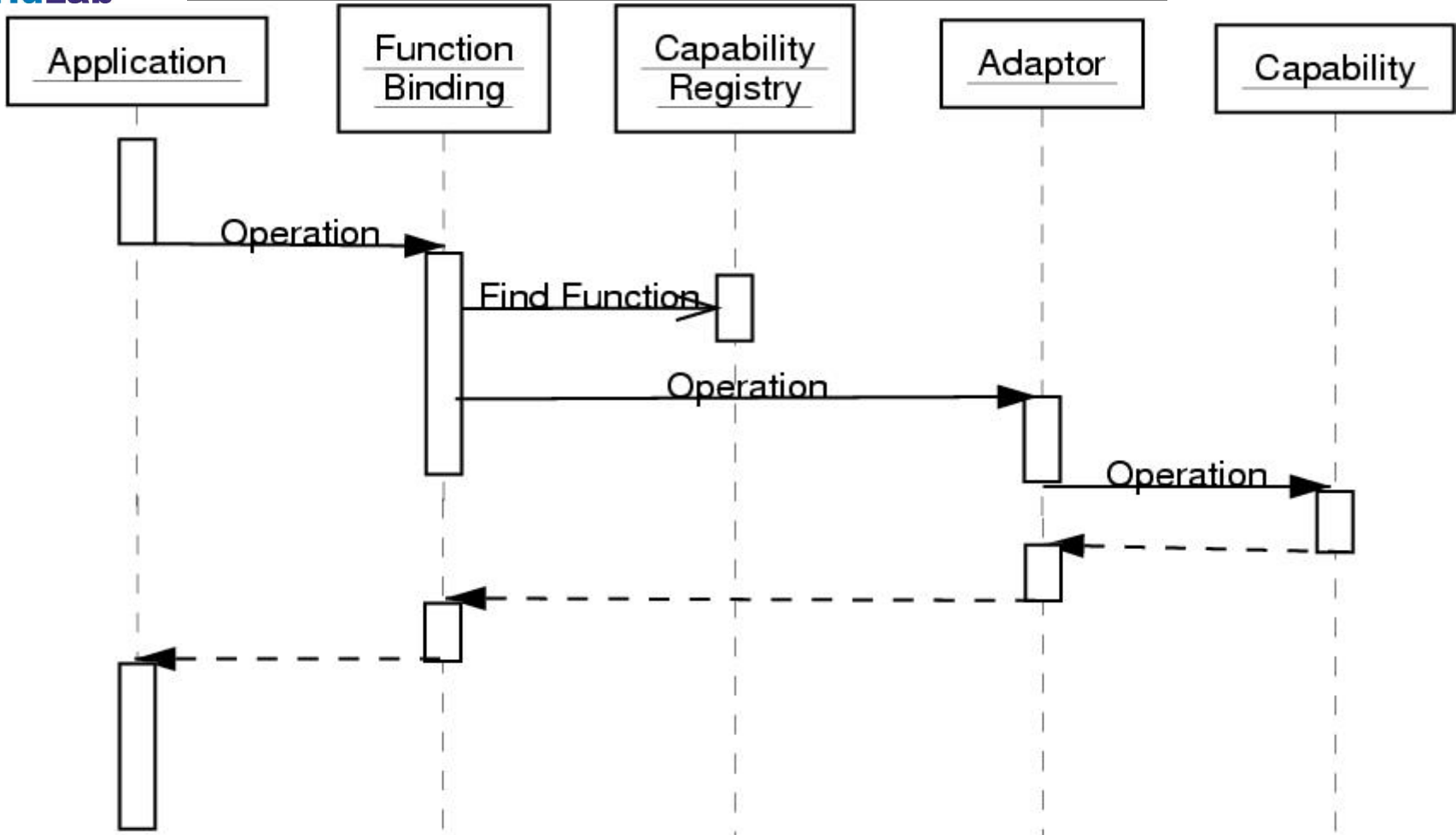
Engine Calling Adaptor



Information Society



An API Call



- The GAT must support applications written in any language which people write Grid Applications in:
 - C, C++, Fortran, Java, Perl, Python, ...
- The use of the GAT API should be as natural as possible for users of these languages.
- It must also not require a steep learning curve to move from the API in one language to the API in another language
 - APIs in different languages should be as similar as possible

- Examined many scenarios and other projects to identify the 'fundamental' operations from an application developers perspective.
- Considered the required level of abstraction for these operations and hence the level of detail of information which must be passed in or returned by the GAT API calls.
- Produced an abstract, language-independent API
 - Use this to provide language-specific APIs.

- We have an API specification, and a C reference version of the GAT Engine and adaptors which implement this specification.
- Simple adaptors available for all of the GAT capabilities.
- GridLab adaptors being developed for all capabilities
- Work started on a Unicore adaptor.
- Have first cut at C++ bindings
- Have Java implementation

- Review implementation and resolve issues which arose with API specification.
 - Re-issue and re-review API-specification
- Document C, C++ and Java bindings.
- Finish users and adaptor-developer guides.
- Develop Fortran, Perl, and Python APIs
 - Will use SWIG to provide bindings to C reference implementation where possible.

- Work with service developers to provide adaptors.
- Use GAT to Grid-enable as large a class of applications as possible
 - already in discussions with lots of application groups.
- Review:
 - There are bound to be problems with provided functionality and the usability of the specification, and there will be feature requests.

- Release version 2 of API specification after review.
- Revise the language specific API specifications in line with this new version.
- Update all implementations to the new version
 - C reference implementation
 - Adaptors
 - Other language wrappers
 - Java native implementation
- Update user code to use new revision.

- Feed experiences into GGF SAGA-RG
 - should bring in many more user groups and service developers to make a more powerful and useful API
 - when SAGA API is developed, provide an implementation of it
- GridLab 2 ?
 - Follow up to GridLab project
 - Not sure when it will be, but it will happen
- We will continue development of the GAT in any case

- The GAT API can be split into two parts
 - Objects and operations involving potentially remote objects
 - Necessary infrastructure junk.
- The remote objects cover various sorts of operations
 - Files
 - Inter-process-communication
 - Monitoring and event handling
 - Information publishing and retrieval
 - Resource and job management

- GATFile
 - Basic operations on a remote file
 - Copy, move, delete
- GATFileStream
 - Open, close, read, write, seek
- GATLogicalFile
 - Front end to a “Replica Catalogue”
 - Each LogicalFile is a reference to a set of identical files scattered around the Grid.

- GATPipe
 - Basically a socket which the process can read from and write to
- GATEndpoint
 - Represents one end of a communication channel
 - Can be advertised
- GATPipeListener
 - The listening end of a socket
 - Generates new GATPipes on each connection

- GATMetric
 - A quantity about a remote process which can be measured
- GATMetricEvent
 - Created when new metric data is available
- GATMetricListener
 - Listens for new metric data to be available

- GATRequest
 - Some application-level event
 - e.g. Checkpoint or user-defined metric
- GATRequestListener
 - Listens for incoming requests
- GATRequestNotifier
 - Tells the remote requester that a request is finished and reports results

- GATAdvert
 - Contains the information about some GAT object
- GATAdvertService
 - Accesses adverts

- GATResourceDescription
 - Hardware or software
- GATSoftwareDescription
 - What we want to run
- GATJobDescription
 - What to run and where
- GATResourceBroker
 - Find resources
 - Start jobs on resources

- GATResource
 - Opaque object indicating a found resource
- GATJob
 - A remote job instance
- GATReservation
 - A reserved resource

- GridLab web page
 - <http://www.gridlab.org>
- GAT
 - <http://www.gridlab.org/WorkPackages/wp-1>
 - <http://www.gridlab.org/WorkPackages/wp-1/Codes>