Apollo

Core specification

Confidential

Revision number: 1

Date:

Name: P. van der Velde

The core part of Apollo provides services to the User Interface (UI) and the Project Sub-system. Services provided are

* Plug-in & component tracking
* Creation of new projects, loading of existing projects
* Creation of project elements (data / generators / visualisers)

Furthermore some default services are provided like

* Messaging
* Exception handling
* Commands
* Logging
* Start-up and shutdown
* Security + license handling
* Provision of default utilities like class loaders for injection into different AppDomains

Note that all interfacing between the core and other sections must be done through default interfaces and remotable information (no objects other than the .NET framework default ones)

In order to keep the core flexible and relatively easy to maintain a ‘non-pluggable’ plug-in architecture will be used. This means that the core will only load default additions, i.e. additions that are known at compile time. Obviously the core is a trusted part of the system and should thus only load our own stuff. Also extension of the core is (currently) not important.

All capabilities of the core are encapsulated in a ‘service’

* Describe start up
* Describe shut-down
* Describe messaging
* Describe logging

Note that using services for everything means that we have to specify relations and dependencies. Some services will have to be loaded before others. Also we have to do circular reference checking etc.

How will we connect services? Commands seem like they might be useful here. However how will we invoke these commands? They will have to be remotable and the results will have to be too.

Note that there are two levels of services. The base level services like from the second set and the more complex systems from the first set.