

DESCRIPTION OF THE PRISM SYSTEM ARCHITECTURE AND USER INTERFACE.

Patrice Constanza (1), C. Larsson (2), H. Thiemann(1), N. Wedi (2)
 Mail: constanza@dkrz.de / Fax: + 49 40 41 173 476.

The PRISM work package 4b: systems and networking.

The purpose of the PRISM system is to enable users to perform numerical experiments, coupling interchangeable model components, eg Atmosphere, ocean, biosphere, chemistry etc., using standardised interfaces. The general architecture provides the infrastructure to configure, submit, monitor and subsequently postprocess, archive and diagnose the results of these coupled model experiments. Such choice of architectural design allows these activities to be done remotely through the internet. The configuration provider processes are accessed through a central site but the services can be distributed to other sites The execution process is local to the model provider. This type of architecture is described as directory centric.

PRISM Targets:

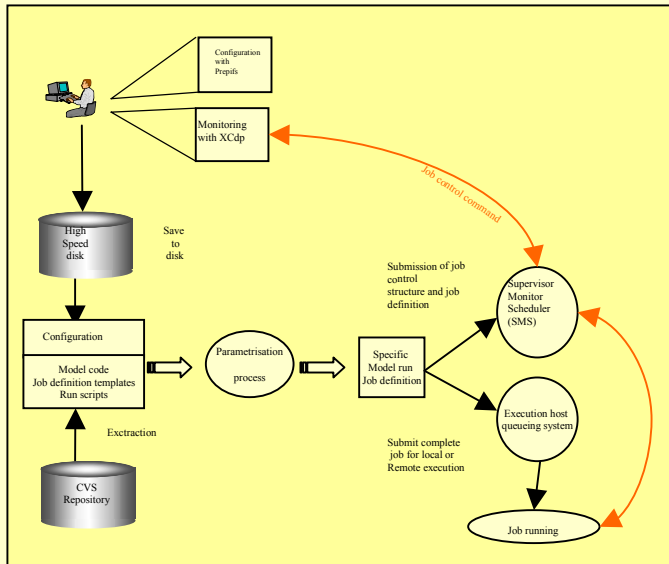
- **Users** – Running model experiments on remote hosts.
- **Developers** – Software developers creating models.

PRISM Requirements

- **Users** - Support and pre-tested configurations to minimise errors.
- **Developers** - Full control of all aspects of the compiling and building environment.

PRISM Functionality:

- Model and Configurations.
- Diagnostics.
- Archiving.
- Source code archive and versioning.
- Experiment database

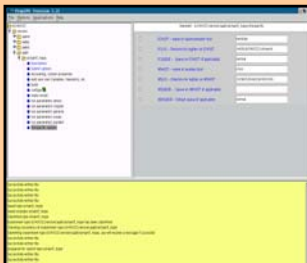
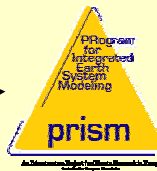


PRISM architecture features:

- **Extendability**
- **Collaboration** –Exchange of configurations.
- **Support**-Tested configurations.
- **Accessibility** -Remote configuration and job control.

PRISM Standards:

- Standard host type configurations
- Standard compile structure
 - configure
 - extract sources
 - build
- Standard run structure
 - get data
 - run model
 - diagnostics
 - archive
- Consistency knowledge database
- Source code repository
- Standard archive
- Standard diagnostics package
- Standard Model I/O and descriptors.



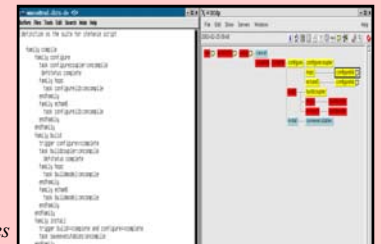
Prepifs enables

- To select individual model components to couple.
- To configure the specific Model Input and Output Configuration (SMIOC).
- To compose the Specific Coupling Configuration (SCC).
- To select pre-/post processing options.
- To select the site and computing resources

Prepifs will read and display the defaults options specified by the model administrator in a XML format (compiling options, Build Options, Running parameters...). Then the user can select and modify the experiment setup.

SMS enables to run a large number of programs with:

- Dependencies on one another.
- Dependencies in time.
- A good tolerance of hardware and software failures.
- Good restart capabilities



The colours represent the status of the different tasks:

Red : Abort yellow : completed blue : queued
 green : repeated Orange : suspended

A SMS setup only monitors the running or the compiling of the user experiment. Three type of files are necessary to create such setup (.Def files, .SMS files, Scripts).

