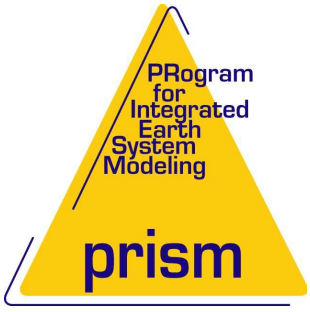
PRISM Support Initiative



PRISM User Survey Final Report Part 1: User experiences and expectations

Graham D. Riley & Rupert W. Ford

PSI Management Report
05

November 2005

PRISM User Survey Final Report Part 1:
User experiences and expectations
Version 3.0

Graham D. Riley & Rupert W. Ford
Manchester Informatics Ltd (Computer Science),
Kilburn Building,
The University of Manchester,
Manchester, M13 9PL, UK
email: [graham.riley, rupert.ford]@manchester.ac.uk

Overview

A survey of institutions and groups involved in Earth System Modelling in Europe has been undertaken on behalf of the PRISM Sustained Initiative (PSI). The aim of the survey was, first, to gather information on the experiences of people who have used or developed the tools which have emerged from the EU-funded PRISM project and, second, to gather ideas about how future tool and infrastructure developments might be organised and sustained on behalf of the ESM community. The groups involved in the survey are listed in Appendix A. The survey involved the completion of a questionnaire by the groups and three groups were identified by the PSI Team to be visited for more in-depth discussions. These were ECMWF, IPSL and MPI-MET. This version of the report includes comments and answers by PRISM tool developers, marked by *CBD* (comments by developers)

Some high-level points which have emerged clearly are:

- There is clearly very positive support in the community to co-ordinate and participate in activities which serve and support Earth System modellers in doing their science, both within the EU and the wider world (US and Japan, for example).
- There is a very strong view of the *priority of science* in the activities in institutions. Infrastructure effort takes second place to achieving scientific results (and the results are typically driven by IPCC requirements).

The structure of this report is as follows: Section 1 reviews the scope and remit of the Survey, as expressed in the original letter soliciting the survey and the motivational text included in the questionnaire. A summary of the key aims is given. Section 2 presents statistics, mostly in graphical form, related to the responses. In Section 3, the main results from the Survey are presented. This section contains sub-sections for each part of the questionnaire. In each subsection, the consolidated responses from all the questionnaires are gathered. Each subsection is preceded by a summary of the responses. This summary gives an indication of the number of responses which made the same (or similar) points and thus gives a good overview of the points which are thought to be important by the community. Section 4 summarises points made during the visits which took place in conjunction with the Survey.

The authors have deliberately tried to avoid adding any editorial view or perspective in this document. The principal aim of the document is to reflect back to the community what has been said and to give some indication of the strength of feeling on common points and issues.

Contents

1. Scope and Remit

This section briefly discusses the remit of the survey undertaken on behalf of the PRISM Sustained Initiative (PSI) team. The original letter sent by the PSI team to solicit proposals for the undertaking of the survey is included in Section 1.1. The cover note discussing the scope of the survey, which appeared in the questionnaire sent to members of the community, is included in Section 1.2. The main points are summarised in Section 1.3.

1.1. Original letter requesting the undertaking of a survey (audit)

As you certainly know, the PRISM community is, under the auspecies of ENES, maintaining and supporting the software infrastructure developed during the PRISM project. More details on this PRISM Sustained Initiative (PSI) can be found at http://prism.enes.org/sustained/

The PRISM software available today includes:
-the OASIS3 and OASIS4 couplers
-the Standard Compiling Environment
-the Standard Running Environment
-the Graphical User Interface and Web Services
-the Data Processing and Visualisation tools

One important step we want to achieve in the months to come is to review the experience of all the groups in the community that used or tested the PRISM software. Those groups should be interviewed in order to gather their experience, remarks, and ideas for further evolution, and a summary report should be produced. The groups from the FP5 project that should be consulted are: CERFACS, ECMWF, ETHZ, INGV, IPSL, KNMI, MetOffice, Meteo-France, MPI-BGC, MPI-M&D, MPI-MET, NERSC, PIK, SMHI, UCL, U.Reading. Other groups such as the Modeling Systems Group from GFDL and INPE/CPTEC from Brasil also started using the PRISM software and could be interviewed too.

1.2. Form of words in the Survey Questionnaire

PRISM Software Tools Questionnaire

INTRODUCTION AND GENERAL QUESTIONS

Version: 31st August 2005
Introduction

This questionnaire is designed to capture feedback on the software tools developed in the PRISM project. This feedback will be used to help direct PRISM tool development in the recently formed PRISM Sustained Initiative (PSI). There are also plans to follow up a sample of the replies by email, telephone and/or in person.

The PRISM tools that have been developed may be considered to support the overall process of a user configuring and creating a coupled model, then compiling and running it, and finally

processing and visualising the resulting output. The questions contained in this document are loosely structured around this premise. However, the views of the developers and integrators of PRISM tools are also extremely important. If you are in this latter category then please ignore any questions that are not relevant to you.

We appreciate that your time is precious and the questionnaire has been designed so that it should take around 30 minutes to complete in the case where people have experience with only a single PRISM tool and around 1 hour for the majority of respondents although please feel free to spend as long as you feel is necessary.

We want to gather the views of individuals as much as possible, but we would prefer the main contact person at a site to collate responses where several individuals at a site contribute to answering (separate sections of) the questionnaire. If this is not possible at your institution then individuals may submit their own response.

1.3. Summary of the Aims of the Survey

“To review the experience of all the groups in the community that used or tested the PRISM software. Those groups should be interviewed in order to gather their experience, remarks, and ideas for further evolution.”

“ To capture feedback on the tools produced in the PRISM project” (but not explicitly the *processes* used in the project in which the tools were produced).

“To direct tool development in PSI” (but not explicitly to address the processes to be used to achieve this).

“Aim to capture the views of users, but also those of developers and integrators of tools.” The views of individuals were of prime importance but also institutional views were sought.

Thus, the main focus of activity, and hence in the questionnaire, was to solicit the *technical issues* associated with the tools produced and used in the PRISM project and to identify, and prioritise, the *technical issues* to be addressed by the PSI in the future. The questionnaire also contained a specific section soliciting views on the *role* of PSI in the future.

In addition to technical feedback, several groups have made significant comments about how the PRISM project operated (i.e. the processes involved) and about how PSI should be organised and how it should operate in the future (i.e. its processes).

1.4. Method

A questionnaire (two phases) agreed with PSI Core and tested on two target sites (Met Office and MPI-M&D)

In undertaking the development of the questionnaire and in the follow-up discussions, we have chosen not to consider current PSI documents and exiting processes and structures but rather to focus on capturing the thoughts and requirements of members of the community that PSI aims to include (and serve).

The report is in principal anonymous. It is clearly not possible to anonymise all comments (and this is almost certainly not desirable). In the following sections (particularly in Section 3) the order of quotations from the completed questionnaires in the subsections bear no real correlation to the responders who made the quotes.

2. Survey Response - Statistics

Twenty five sites were initially contacted and requested to take part in the Prism Survey. The contact list was drawn up by the PSI core team and initial contact was made by Sophie Valke. Twenty of these sites are directly involved in Earth System Modelling and five of the sites are Vendors. It was decided to solicit the views of the Vendors as, although they are not direct users of the software, they have made significant contributions to the development of the Prism Software and would be expected, therefore, to have useful insight into the tools. Of the twenty five sites solicited, nineteen completed the questionnaire. The information on who was contacted and who replied is detailed in the following table.

-
-

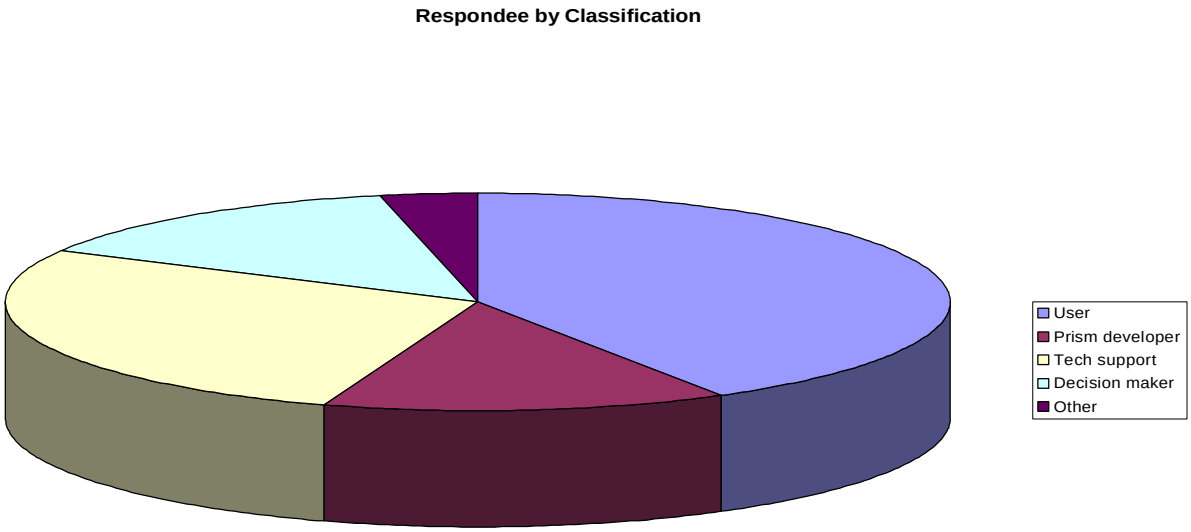
Count	*Sites Contacted*	*Completed Questionnaire*
1	*ECMWF*	*Yes*
2	*MPI-BGC*	*No*
3	*MPI-M&D*	*Yes*
4	*MPI-MET*	*Yes*
5	*IPSL*	*Yes*
6	*CERFACS*	*Yes*
7	*INGV*	*Yes*
8	*KNMI*	*No*
9	*The Met Office*	*Yes*
10	*Meteo-France*	*Yes*
11	*CSCS*	*No*
12	*NERSC*	*Yes*
13	*PIK*	*No*
14	*SMHI*	*Yes*
15	*UCL*	*No*
16	*CGAM*	*Yes*
17	*IFM-GEOMAR Kiel*	*Yes*
18	*ESMF*	*Yes*
19	*INPE*	*No*
20	*Met. Inst. of the FU-Berlin*	*Yes*
21	*CCRLE*	*Yes*
22	*NEC*	*Yes*
23	*Fujitsu*	*Yes*
24	*SGI*	*Yes*
25	*CRAY*	*Yes*
	Number of Sites Contacted	25
	Number of Contributions	19

-

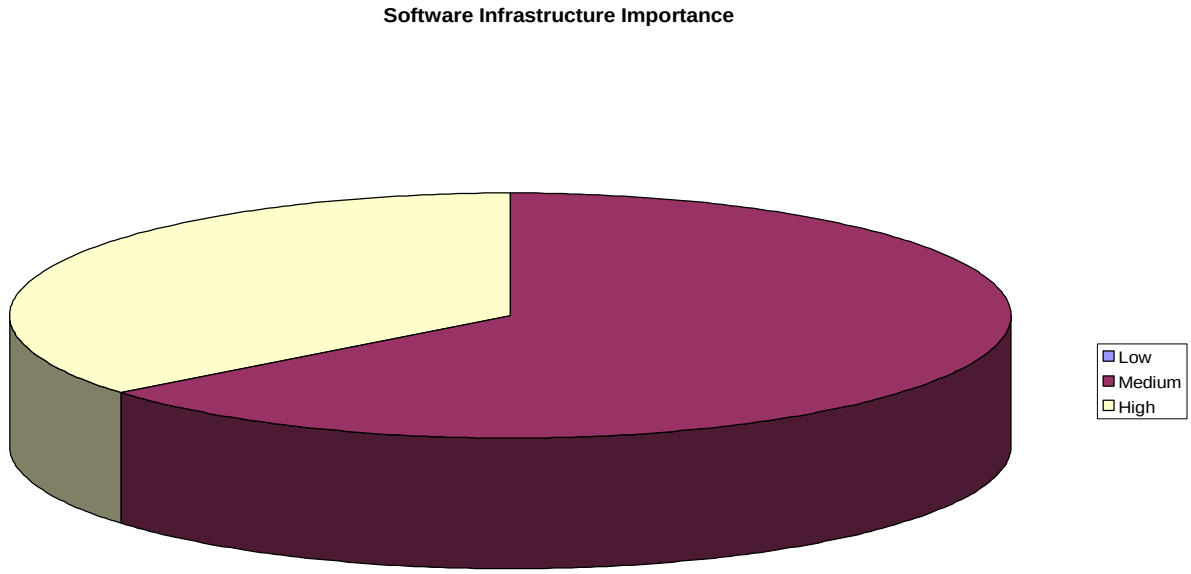
The sites that did not complete the Questionnaire gave the following reasons for not responding:

MPI-BGC	Currently in a restructuring phase
KNMI	No response
CSCS	Not enough time and little experience
PIK	No one available to fill in questionnaire
UCL	Do not wish to participate
INPE	Not yet ready to use Prism

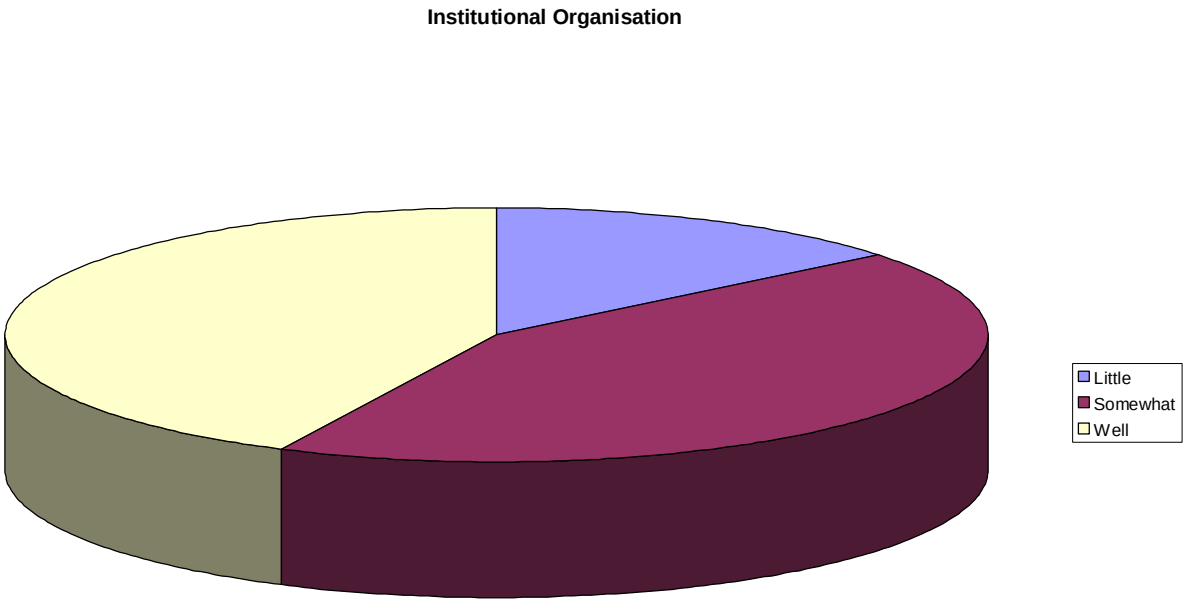
The following chart presents the relative type of experience of the responders. Note that individuals may have specified that their experience covers more than one of the categories and, further, due to the fact that many institutional responses were collated from a number of individual responses, responses often specify more than one of the categories. It can be seen that most responses are from Users but also that there is representation from all classifications.



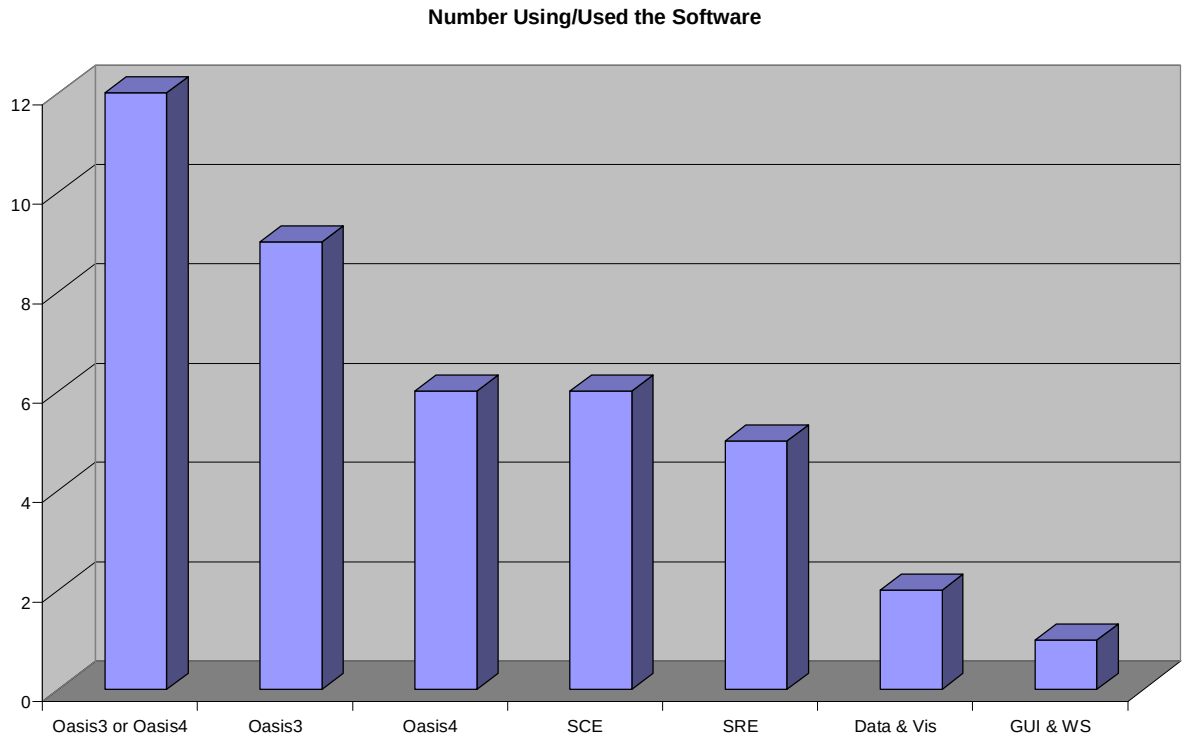
The following chart presents the relative importance that institutions place on infrastructure at their institution. It can be seen that no institutions place infrastructure as being of low priority with the majority placing infrastructure as being of medium importance.



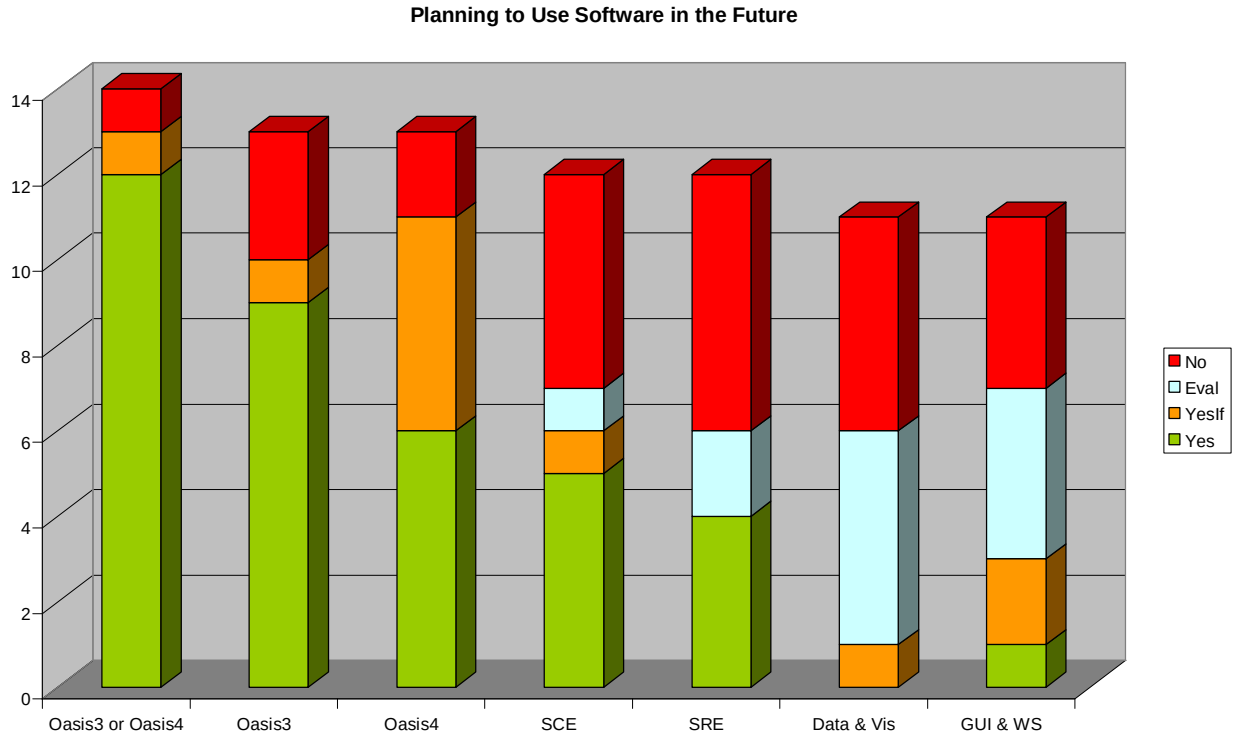
The following chart presents the relative organisation of the sites. It can be seen that few sites consider that they have little organisation, whilst a roughly equal number of sites consider themselves to be somewhat or well organised.



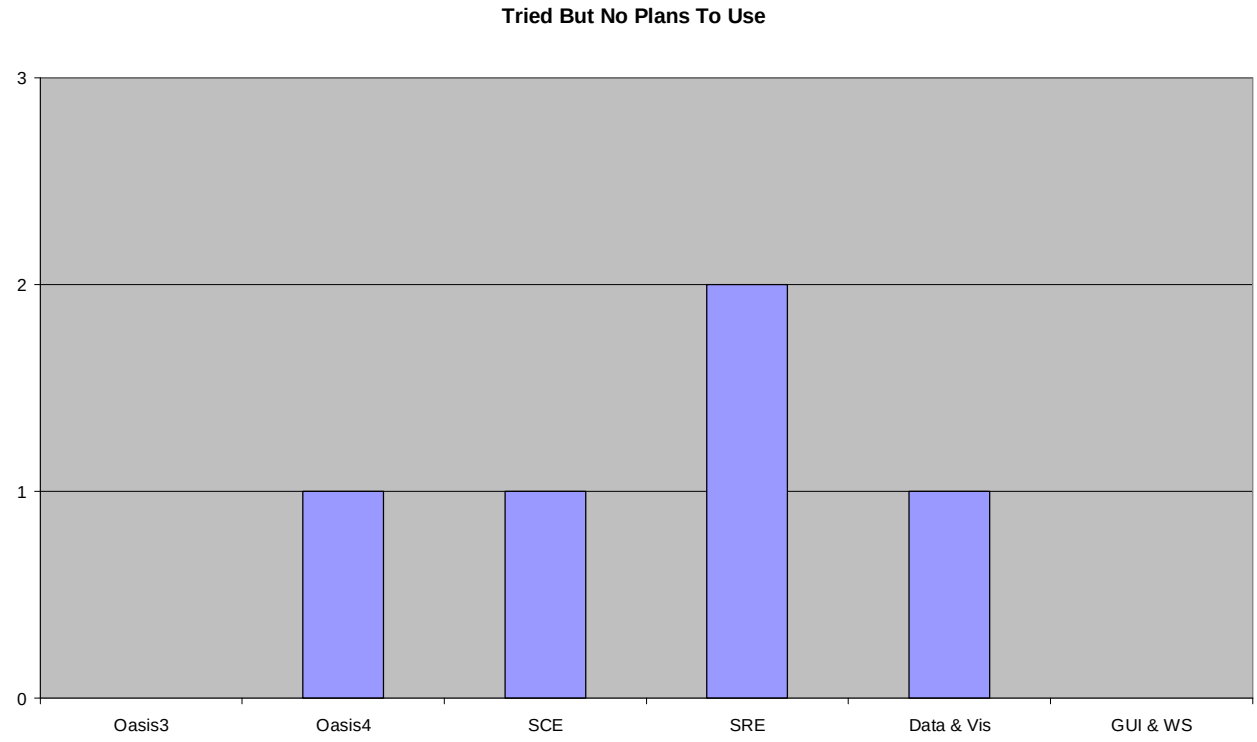
The following graph presents the number of sites that are using and/or evaluating or have used/have evaluated the Prism tools. It can be seen that most sites have used, or are using, at least one of the Oasis couplers. Around half of users have used, or are using the SCE and SRE. Finally, very few sites have used, or are using, the Data and Visualisation tools and the GUI and Web Services tools.



The following diagram presents the sites plans to use, or continue to use the Prism tools in the future. It can be seen that nearly all sites plan to use at least one of the Oasis couplers in the future. This result shows that most of those who said no to using Oasis3 said yes to using Oasis4 and vice versa. At the present time there are also many users that would like to continue to use Oasis3. The SCE and SRE both have approximately half of the sites wanting to use (or evaluate) the software and half not wanting to use the software in the future. The Data and Vis, and the GUI and Web Services tools have between a third and a half of sites planning to evaluate the software and between a third and a half not wanting to use the software in the future.



The following graph presents the number of sites that have evaluated the software but have decided not to use the software in the future.



It is noted that the sections in the questionnaire related to tool integration (between PRISM tools and of PRISM tools with other existing tools) were responded to in only very few returns. More follow-up work would be justified in the future. The initial conclusion is that people working with tools have not yet gained enough experience with integration issues to make useful observations. It is suggested that the issue of integration be evaluated again as experience with a more mature toolset is gained by people in the community.

2.1. Visits

On the recommendation of the PSI core team site visits were made to ECMWF where 3 people were interviewed, IPSL where 8 people were interviewed and MPI-MET where 5 people were interviewed. Fortuitously, at MPI-MET one of the responders from IFM-GEOMAR in Kiel was present and was also interviewed.

3. Survey Sections – Specific Observations

3.1. General

3.1.1. Summary of Questionnaire Responses

This section presents a summary of all points that were raised by at least two different institutions. The number of institutions raising a particular point is considered to be a good indicator of its relative importance to the user community. The points are therefore ordered by number of institutions raising it. For clarity, they are further separated into the categories presented in the Questionnaire. The number of institutions that raised each point is provided inside the square brackets.

Level of infrastructure effort
7 institutions explicitly listed the amount of infrastructure effort in their institution. 6 of these had effort of up to 3 person years/year. One had in excess of 10 person years/year. All institutions implied they had some effort. Effort available is typically made up from a mix of permanent and soft-funded posts.
Plans to improve the organisation of software infrastructure
[5] Improvement is under constant review
[5] Have specific improvements planned or respond only to scientific requirements
[2] Use infrastructure developed elsewhere
[2] No plans to improve in the near future (current infrastructure is adequate)
Difficulties in organising an efficient software infrastructure
[3] Lack of standards
[5] Lack of expertise
[3] Not enough people
[3] Need for platform independence
[3] Soft funding/limited contracts
Standards or standard interfaces that should be common
[12] NetCDF/CF
[4] Standard coupler
[3] GRIB
[3] Standard model interfaces
[3] Standard compile environment
[2] Standard run environment
Role of PSI
[10] Further development of existing tools
[7] Maintenance and support of existing PRISM tools
[4] Maintain an up-to-date website, to promote discussion/dissemination
[4] Define, manage and promote standards
[2] Support for grids (reduced Gaussian/ unstructured) in OASIS4
[2] Improve interpolation in OASIS4
[2] Improve portability and quality control of the software
[2] Promote use of PRISM software on real models in big centres

3.1.2. Collated Responses from Questionnaires

SECTION 1.2: ABOUT YOUR INSTITUTION

Approximate level of infrastructure effort in person-years per year

A

3 py/y on average; 1 person permanent + 1 person (soft money) for development, 1 person full time for support

B

For Earth System Modelling ~ 0.25 person-years per year
For general modeling ~2 person years per year

C

All software is controlled by a common version control system (perforce) available to all users/developers. Several people are employed to ensure an efficient working environment for scientists/developers. A set of GUI's is available to launch and control experiments.

D

F.S.E. has been actively participating to the main PRISM project (2002-2004) [ed. – which provided support effort], however
It is no more formally active in the follow-on project.

E

Please provide brief description:

- Our scientific goals are defined every two years during a general assembly of IGCMG. These goals determine software developments in our institution.
- Technical constraints are discussed frequently and analysed by the executive board
- We have developed software tools (model components and infrastructure tools) that are shared by a large community

F

Ocean-Sea Ice-Atmosphere-River routing-Stratospheric chemistry system. Coupled with OASIS2.
Developers: 1 full-time scientist + 1 engineer (50% of her time)

G

0,2 MM/year

H

3 person-years

J

SGI provide solutions in three areas:
Capability AND capacity computing, storarge and data management and
high-end visualisation.

K

For the atmosphere we are applying the system of our associated weather forecast system. The coupling via OASIS is self made by scientists and a technical engineer

L

The Unified Model is supported by a team in excess of 10 people who also provide technical development, optimisations effort etc. Scientific development is performed through a controlled process. A single installation is required to cater for many different users as well as developers for both climate (global and regional) and weather forecasting for both production, development and research.

M

As a service institute for the German (and wider) scientific modelling community, we aim at having a well-defined, user friendly infrastructure for model based science

N

1 person-year

Plans to improve the organisation of its software infrastructure

A
CNRS should provide to CERFACS 1 engineer permanent position for coupler development and support. We also hope to benefit from the PRISM infrastructure.
B
There is an on-going effort in NCAS to organize modeling centrally to

- keep flexibility and portability
- ensure up to date documentation
- to provide training as well as help and support

This effort is coordinated by CGAM and ACMSU, two units within the distributed organization of NCAS.

C
We have just spend quite some resources converting our version control system from ClearCase to perforce. I am not aware of any plans to make substantial changes to our infrastructure, but it is under continuous development.
D
Yes, we do it on a routine basis
E
[NO] The earth system modelling group at the institute is quite small at present, and does not have enough man power to support such activities, nor do we have much expertise in this area. We are also primarily users of models developed at other centers, and thus we rely on the software infrastructure developed at those centers and within PRISM.
F
Yes.
We are planning the realization of a new software repository for all the models and the transition from CVS to Subversion.
G
We plan to introduce more modularity, flexibility and portability.
Our plan is to be ready for the IPCC AR5 (Assessment Report) which will probably be published in 2013. The ensemble of simulations needed should be carried out in 2010 with a model finalized in 2008 and technically stabilized in 2007. This Earth system model will include new components. Obviously, software infrastructure can always be improved but the high priorities in our planning do not lie there. In any case, changes in software infrastructure must be planned step by step in order that it does not impact scientific work too strongly: incremental evolution is a fundamental need regarding our objectives.
H
Coupling of an aerosol model with the current ESM (thesis - 3 years) [ed. This is model development not really infrastructure].

Making lots of runs (eg IPCC, or for European projects) is of higher priority than development.
I
YES, dependent on the needs of the scientific projects, not planed, no manpower
J
no, there is no economical need for this.
K
We have been using OASIS version 2.2 with modifications on shared memory machines up to know. We know want the system also to run on distributed memory machines and this will require major changes to the system. Therefore are considering upgrading the whole software infrastructure, and PRISM is of course a candidate offering such a solution
L
no resources left for a major change of infrastructure. Also: issues with compatibility of climate and weather models.

M

We plan to replace the current infrastructure with a new infrastructure in the FLUME project. The aims are to be allow new science to be introduced more flexibly into a system that is better designed.
Eventually, FLUME will replace all parts of the system including:

- User Interface
- Version Control
- Compilation
- Coupled model codes and framework
- Post processing and archiving
- Data and file formats

We hope to share as much with PRISM as possible in the project. The first stage of the project will use the PRISM coupler (OASIS4) to couple our Atmospheric model with the NEMO ocean model from IPSL.

N

we recently decided to use the PRISM coupling and infrastructure software; it will be extended in coordination with the PSI group according to our client's needs;

O

Unifiy model infrastructure, improve I/O and parallelization, user interfaces, postprocessing

P

(1) PRISM software just used as a tool (for interpolation), and not for running a whole model environment. (2) no resources for numerical computing

Difficulties in organising an efficient software infrastructure

A

1. there is not one standard
2. we need to adapt to the needs of our partners

B

1. Lack of IT experts specialized in Earth System Modelling at CERFACS.
2. Lack of technical standardization and coordination in the development of the models we use.
3. Lack of time to visit other institutions/laboratories to explore and compare with other software infrastructure.

C

1. the broad range of research topics in NCAS
2. the need for outcomes for short focused projects within the NCAS community
3. rapidly changing modelling environments

D

Our infrastructure meets our present needs efficiently. Because of operational demands, the system is not particularly simple. Incorporating new physical models into our research and operational framework is not always trivial. But given the importance of ease of use for those running experiments, the requirement for computational efficiency and the (always limited) human resources for infrastructure development, we feel we have a reasonably balanced system.

E

1. Code legacy, backward compatibility
2. code complexity, many computer platforms
3. communication between scientists and software developers

F

1. Man power.
2. To Guaranty Platform independence, since we are computing on several different platforms.
3. Lack of expertise in software development/maintance

G

1. Lack of knowledge about software handling and development between users (especially scientists)
2. Lack of staff who can maintain/develop the software infrastructure
3. Willingness of users to adhere to the software infrastructure

H

1. Developing model components and assembling an Earth System Model which provide good scientific results. And for each single component development: readability, ease of use, scientific quality, performance and portability are primordial. The same applies to the Earth System Model with a focus on phasing of model components.
2. Organising the sustainability, portability and performance of the Earth system model and its components.
3. Reliability of the coupler and interfaces between components.

I

1. Lack of manpower
2. Rapid changes of models, grids, etc...

J

1. soft money financing
2. deep valley between software engineerung and earth system scientists
3. mostly bad documentation of earth system software and too specialized earth system software

L

1. Adapting to multiple machine architectures.
2. Make the software infrastructure available to many non-expert end users.
3. Availability of persons with the combination of technical/software developing skills and geophysical skills.

M

1. Standards in data and storage management
2. Amount of storage installed
3. Fast networks and standards in networking. Networks means networks in the terminology of computer sciences.

N

1. Scientific applications are more important the infrastructure improvements as long as the traditional infrastructure can be handled. The level of suffering is not yet high enough

O

1. Meeting the demanding requirement in terms of flexibility,
performance and
robustness.
2. Designing and keeping clean code that can achieve the main aim.
3. Doing this as well as dealing with operational issues outside our
control.

P

1. limited contracts of people; they therefore do not have the time to really understand the software design
2. concerning our clients:scientists don't have infrastructure high on their priority list; they would like to ... but later
3. concerning our clients: rapid model development which requires frequent adaptation to infrastructures; the benefit is therefore not always obvious;

Q

1. manpower
2. qualification

R

1. no (long-term) human resources for numerical computing (either scientists of computer technicians)
2. heterogenic architecture and spread of resources over different supercomputing centres
3. complicated handling

CONVENTIONS AND STANDARDS

What standards or standard interfaces should be common between different Earth System models?

A

1. CF convention for describing model data
2. IO file format for data exchange
3. General code structure, calendar for model coupling

B

1. Standard data format (input and output) or efficient converter; NetCDF with CF convention if possible.
2. Interface with standard coupler; PSMILe for OASIS
3. Standard compiling environment

C

1. the CF convention for describing model data
2. Unix for running models
3. Models using MPI for parallelisation

PRISM develop the 3 tools/standards described in Q2.1 (but maybe too many others tools too).

D

1. Common data formats.
2. Common compile environment (between institutions).
3. The same coupler ideas.
4. The same software management system.
5. Code writing conventions.
6. Script writing conventions.

Remark with respect to data formats: Since ECMWF is an operational weather centre our main data format is the WMO GRIB format for fields rather than the netCDF format typically used for climate research. Our requirements are different from other PRISM users in this respect.

E

1 .CF convention is a good start
2 .standards should work on different platforms (IBM, SHI, NEC, …)
3. standards should allow using models from different institutions

F

1. CF convention for describing model data
2. Several supported physical interfaces, to simplify coupling with different components.
3. Standardised and simplified runtime and compile time environments

G

1. Conventions for input and output datafile (e.g. NetCDF CF)
2. Communication libraries (e.g. PSMILE)

H

1. NetCDF/CF
2. OASIS (Europe)
3. That's all …

Remark : we believe that definition of boundaries and interface between components is a scientific problem (physics, mathematics, numerical). Technical normalisation of coupler interfaces is relevant to achieve reliability between forced versus and configurations.
On the other hand, the target of plug-in plug-out components is not a good question as it is related to fundamental scientific choices.
For example : no best solution exists yet for a 3D interface between biogeochemistry and atmosphere.

I

1. Data: IPCC Netcdf CMOR standard

J

1. standards for data exchange (GRIB, CF, netCFD)
2. standards for documentation (e.g. doxygen) and coding (e.g. FORTRAN95)
3. standards for earth system component interfaces

K

1. CF convention

L

1. Use conventions that are consistent with the requirements of model data deliverables to the IPCC's assessments reports as most recently were enforced by PCMDI.

L

1. Portable file formats. Netcdf using the CF convention is important
2. Standards in accessing and finding data of experiments. DOD is a good approach.
3. Physical interfaces of models. Important for plug-and-play of different components.

M

1. netcdf/cf
2. the technical interfaces of the models to OASIS4

N

1. Common file formats and meta-data standards
2. Common ways of developing models so that they can be more easily exchanged (Single Model Conformance)
3. Common "patterns" to solve common problems, such as the need for service models, like advection.

O

1. for cooperative data analyses: CF conventions, file format (NetCDF)
2. for flexible coupled model setup and running: portable SCE/SRE
3. for scientific cooperation between institutes: coupling software

P

1. CF convention
2. File formats (NetCDF, GRIB, HDF,...)
3. MPI, OpenMP, C, FORTRAN

R

1. NetCDF for all I/O

To what extent does the current PRISM tool set fulfill the requirements for standards?

A

It does not fulfill our requirement for GRIB, but since most users are not operational weather centres maybe it should not do so. [ed. Note also the requirement at another institution for GRIB to be used because it produces smaller volumes of output and the cost of storage is increasingly becoming a factor in decisions]

B

PRISM has not really fulfilled
1 (CF), since various model components still output data in their own binary formats.

2 (supported physical interfaces), since it still remains non-trivial to couple two different components
PRISM has fulfilled to quite some extent 3 (SCE/SRE), although it is often hard to modify the environment, and track errors when they occur.

C

It fulfills them completely.

D

In our opinion not very well

E

Yes. It is feasible but should take time and our priorities are not these as long as our coupling works. Priorities are 1/ Running models 2/ Developping model physics 3/ If possible work on interfaces

F

98% fulfilled

G

Very well.

H

PRISM is using NetCDF/CF. Visualisation software is partially DOD capable. Physical interfaces were not standardized.

I

PRISM is fulfilling the above requirements, but PRISM is much more that just the above.

J

PRISM does the first (common file formats and metadata standards)

K

The PRISM tools meet the standards above. However, it is important that the tools are flexible enough to deal with models/data that do not exactly meet the standards. E.g. It should be possible to visualise a netcdf data file that is not CF compliant using prism tools (and it is).

L

Mostly with the exception that GRIB, HDF are not supported and quite some non-standard features are still used (binary machine dependend I/O)

M

fulfills completely

Is it practical to implement these standards in your existing infrastructure?

A

As we do not develop the models we use, it would be required that the developers of these models (ARPEGE and OPA) release their models in a standard environement.

B

Other communities developing or using these models would have to agree.

C

It is not easy to answer YES/NO to this question for us. We are using the prepIFS tool quite extensively (in our own version) and plan to make extensive use of the OASIS4 coupler for various coupled configurations. Other tools like the SCE and the SRE would require a large investment in man hours to implement within our setup and that might not be feasible

D

[YES] for NetCDF/CF and Oasis.

E

OASIS: yes, SCE/SRE/GUI: no
If your answer to Q2.3 was "No", what would be required to make it practical?

Widespread use of a mature PRISM system by big centers (such as MPI, Hadley, MeteoFrance, etc) for daily operations would raise acceptance within our small climate modelling group. This would enable better collaboration.

F

Yes for CF, although we would have to consider the performance implications. NetCDF files are also very large compared with our equivalents.

ROLE OF THE PRISM SUSTAINED INITIATIVE (PSI)

A

1. (Short term): maintenance and support of PRISM tools
2. (Short term): further development of existing tools
3. (Longer term): Definition of standards

B

1. (Short term): Development, maintenance, and support of OASIS3 and OASIS4
2. (Short term): More support for coupled model assemblage (very important)
3. (Longer term): Provide a Standard Compiling Environment, flexible and efficient

One important role of PRISM should be to be a discussion forum for the tool developers but also for the scientists assembling and using coupled models.

C

1. (Short term): simplicity and ease of use
2. (Short term): flexibility to be able to implement changes
3. (Longer term): sustainability

D

1. (Short term): Full support for reduced Gaussian grids in OASIS4.
2. (Short term): Standalone interpolation for OASIS4.
3. (Longer term): New configuration management system for compilation and running.

E

1. (Short term):interpolation algorithm should return results independent from number of processors
2. (Short term): customer support

F

1. (Short term): Standardisation of model output formats
2. (Short term): Data management. ESM produce vast and ever increasing amounts of data. Effort has to go into methods for writing the data (parallel mode), accessing the data (data structures), and for compressing and saving data. The method for determining the data to be stored should also be standardised
3. (Longer term): Separation of physical and technical interfaces used for coupling, the specification of several possible physical interfaces in any given model, and the specification of several different technical interfaces, and in particular allowing for different parallelization standards/methods.(the last point refers to problems encountered when porting coupled models to different platforms.)

G

1. Maintain a website with updated information on the available tools and how to obtain them;
2. Support the development and optimization of communication interfaces (e.g. PSMILE).

H

During PRISM project, each group developed its part and provided the result as a PRISM tool. For example :

- SCE and SRE came from MPI

- GUI came from ECMWF

OASIS came from CERFACS with large European expertise and use. [Summary: OASIS has been the most useful result from PRISM because of the interaction with the community before and during its development. PSI would operate best following this example.]

I

1. (Short term): 3D coupling (eg 3D aerosols coupled with atmosphere or ocean biogeochemistry coupled with ocean)

J

1. (Short term): portal to open earth system standards and tools
2. (Short term): administrator of standard developments in the earth system area

K

1. (Short term): Improvement of graphical tools
2. (Short term): Adaptation of OASIS4 to SCE

L

1. (Short term): Make sure the software tools compiles and works as inended on a wide range of operating systems and compilers.
2. (Short term): Keep PRISM web sites up to date.

M

1. (Short term): Stabilizing the software written.
2. (Short term): Stabilizing the software and adaptation to the standards set by PRISM , not vice versa.
3. (Longer term): Collection of missing features and a respin of the development effort undertaken by PRISM to implement new feature.

As a support team improve your bug tracking system, get a bugzilla server.

N

1. (Short term): the completion of the OASIS 4 coupler development
2. (Short term):
3. (Longer term): Real applications of the complete PRISM system by big centers

O

1. (Short term): Review and potential adoption of FCM
2. (Short term): Review of descriptive meta-data
3. (Longer term): Developments to support an institutional installation, like the UM (but better architected). This will allow us to replace the UM framework will a PRISM framework.

P

1. (Short term): Support for the existing PRISM software. Bug corrections.
2. (Short term): Further development of the tools. Further improving the interconnection of SRE, post-processing, visualisation and archiving.
3. (Longer term): Wide spread of PRISM software in Europe and the world. Earth System Modelling network.

Q

1. (Short term): more extensive quality control of codes
2. (Short term): more support for unstructured grids and support for GRIB
3. (Longer term): oasis4

R

1. (Short term): a more user-friendly handling of the software (e.g. graphical editor of the xml-files)
2. (Short term): optimization (already started by moving from OASIS3 to OASIS4)

3.2. OASIS3 Coupler

3.2.1. Summary of Questionnaire Responses

This section presents a summary of all points that were raised by at least two different institutions. The number of institutions raising a particular point is considered to be a good indicator of its relative importance to the user community. The points are therefore ordered by number of institutions raising it. For clarity, they are further separated into the categories presented in the Questionnaire. The number of institutions that raised each point is provided inside the square brackets.

Tools used previously or currently in-use
[12] OASIS2 (2.2 or 2.4)
[2] OASIS4 (for test purposes)
[2] Own solution (e.g. STASH at the Met Office)
Reasons to use or not use
[6] Planning to migrate to Oasis4 (either from Oasis3 or earlier versions)
[2] Oasis3 is currently in use
[2] Oasis3 is used purely for interpolation
Benefits
[4] Portable
[4] Targeted at ESM needs
[4] netcdf support
[4] Well used standard
[2] Good documentation
[2] robust
[2] interpolations
Drawbacks
[4] too slow
[2] only 2d exchange
[2] difficult to debug
Improvements
[5] make more efficient/parallel
[3] support 3d exchange
[2] no further improvements should be made
Benefits: (of using with other PRISM tools)
[3] accepted standard (with existing model implementations)
[2] Oasis3 integrated with SCE and SRE
Benefits: (of using with other non-PRISM tools)
[2] Oasis3 is usable independently of other PRISM tools
Drawbacks: (of using with other non-PRISM tools)
[2] requires modification to integrate with other environments

3.2.2. Collated Responses from Questionnaires

SECTION 1.1 OASIS3 AND YOUR SYSTEM – Reasons for use OR non-use in the future

- We just used OASIS3 for interpolating data sets between grids. In general OASIS3 was too slow (e.g. calculating the metrics), and we have moved to OASIS4.
- OASIS is essential for our ESM development. The reliability and stability of the coupler cannot be overstated.
    - OASIS 3 new features and improvements with respect to OASIS 2.4 are useful for our ESM. OASIS 3 is now used in our reference versions.
- If oasis3 is part of a coupled model, we will use it.
- Instead of OASIS3 we will consider using OASIS4.
- I am continuing development on OASIS 4.
- We take a bigger step from OASIS2.4 to OASIS4, to save resources in learning and implementing Oasis3
- We will use OASIS 4
- We want to use OASIS4 in the future

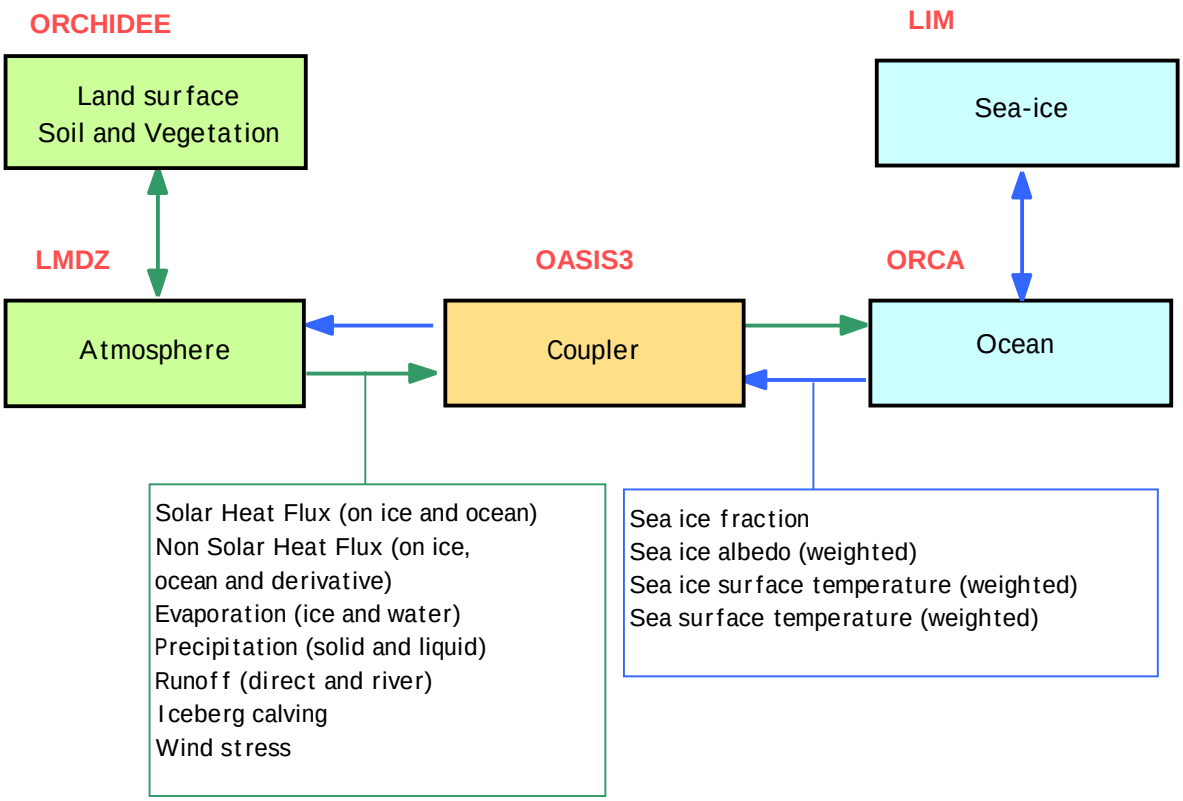
OTHER TOOLS IN USE (Previously or Now)

A
OASIS2.4, MpCCI
B
There was no life before OASIS.
C
OASIS 2.4
D
OASIS2.
E
Oasis4 for tests of Prism system
F
OASIS2.4
G
OASIS 2.4
H
Oasis 2.4
I
OASIS2 - still in use for the older version of our ESM.
J
oasis2.4 as part of the ECHOG-Model
K
Toymodels, ECHAM5-MPI-OM
L
OASIS version 2.2
M
None
N
Oasis2.4 and Oasis4
O
The UM has a hand written coupler through the STASH diagnostic system
P
OASIS0-2.4
Q
OASIS2

SECTION 1.2 EXAMPLE USE

- Use of OASIS3 in the ocean-atmosphere coupled model assembled in the framework of an on-going European project (e.g. DYNAMITE).
- OASIS 3 is being used to couple the Unified Model version 4.5 at various horizontal and vertical resolutions to OPA at various horizontal resolutions and at different coupling frequencies. Many model decades of the coupled model have been run on the Hamburg NEC SX6.
- Interpolation of surface fields from our AGCM (IFS) for use as input data to ocean models.
- Interest in OASIS is linked to its use by Fujitsu Systems Europe customers. Currently this involves only Meteo France users, as long as the VPP5000 is in use on this site.
- Using OASIS3 for performing exchanges between different components of a couple model.
- OASIS3 was used to interpolate data from a Mercator grid to a curvilinear grid. This was done prior to the model run, e.g. for initial and forcing data, topography. SO, OASIS3 was only used as in stand-alone mode with a self-developed wrapper around (called pseudo-model).
- A typical scenario at our institution is a coupled model run with OASIS3 exchanging fields between the atmosphere and the ocean. Another typical scenario is the use of OASIS3 in interpolator only mode to test different interpolation routines.

- 1 – OASIS 3 is also used as a standalone interpolator to produce fields for the IPCC database and to produce forcing fields for ocean standalone simulations.
    - 2 – OASIS 3 in our ESM



- Seasonal prediction model (Ocean-Atm-Sea ice)
- Any example which a customer would like to see on our computer solutions.
- We are running several coupled models using OASIS3, e.g. the ocean-atmosphere coupled model ECHO for IPCC scenario runs;
    - since OASIS3 is flexible enough to be used with different combinations of component models and different physical interfaces, we use it for hierachy of coupled model going from standalone versions of components to comprehensive multi-component Earth system models
- Coupled integrations Atmosphere/Ocean/carbon cycle (IPCC AR4)

SECTION 1.3 TECHNICAL EXPERIENCES

Benefits:

A

1. open source
2. directly supports earth system model requirements
3. well accepted

B

1. Good model interface (PSMILe) significantly improved compared to OASIS2
2. Use of a standard tool widely used in our scientific community.
3. Tool containing specific functionalities required in our type of applications (climate modelling).
4. OASIS3 comes with a toy coupled model, which gives a practical illustration of a coupled model, and is easy to modify to test other OASIS3 interpolations.

C

1. It has a more consistent interface (PRISM puts and gets)
2. Building OASIS 3 was easier as there were tools and libraries to do this
3. Input output is now in netcdf

The documentation is good

D

1.Simple to use
2. Ported on many platforms (good portability)
3. most useful features for interpoling exchanged fields between components

E

1. Provide a good technical and relatively easy to implement framework for developing a coupled model
2. Relatively widely used software, and so quite some support and experience exists

G

1. dealing with all sorts of grids
2. complete NetCDF support

H

1. Enhanced support for NetCDF files
2. Several useful diagnostics (e.g.: expout)
3. Choice of different interpolation libraries

I

1. Portability improved (MPI1)
2. I/O of coupling fields
3. Parallel communication between coupler and components
4. Easier to use psmile library compared to CLIM

J

1. Possibility to have grids, areas masks and coupling files in Netcdf
2. Thanks to f90 rewriting, it is now easier to use the software (in particular, interpolation info is now read essentially in the namcouple, not in the namcouple + the parameter.h !

K

1. Portable and flexible

L

1. Robust approach
2. Clearly designed for climatologists.
3. Supports shared memory and MPI programming paradigm

L

1. It is possible to couple existing models with a minimum impact on the model's source code. OASIS3 includes time and unit transformations, a number of interpolations fullfilling ESM needs (e.g. conservation, smoothness) and time management of exchange.
2. The software is very mature and well tested. We have a lot of in-house experience.

3. OASIS3 is well documented, the support from CERFACS is very good: Questions are answered right away, suggestions from our side are taken into account.

M

1. easy change of models
2. portability
3. standard conformance

Drawbacks:

A

1. Need of special input files for grids, masks, etc.

CBD: In OASIS3, there is no need to create those files before hand. There are PSMILe routines that can be called in the model which lead to the creation of those files on-line.

2. Direct support of 2D interpolation only

CBD: 3D interpolation is supported in OASIS4.

3. Search algorithm is too slow for large problems

CBD: OASIS4 is much more efficient.

B

1. Suffers from its long history (weight) (e.g. ASCII configuring file, some binary input files, etc.).

CBD: We agree. This is why OASIS4 was rewritten from scratch.

2. Some rigidity in the exchanges (one to one only, one to many not possible).

CBD: OASIS4 is more flexible.

3. Some transformations are limited (e.g. the extrapolation works only for logically-regualr grids).

C

1. SCRIP doesn't work with the OPA grid

CBD: This is not true. SCRIP has been used many times to interpolate to or from the ORCA grid. Please contact us directly (oasishelp@cerfacs.fr) for any particular problem.

2. It doesn't work with 64 reals and 64 integers, which needed fixing before it would work with the Unified Model

CBD: A user compiled OASIS3 and the toymodels on a NEC SX-6 at BMRC

with the "-ew" option (so to declare 8 byte INTEGER as well as 8 byte REAL data size), and had to set some special environment variables ("F_UFMTADJUST" and "F_UFMTIEEE") to enable the files "at31topa" and "runoff31" to be read properly (because they contain 8 byte REAL, but only 4 byte INTEGER data).
3. Debugging messages are not clear or consistent
CBD: We agree. We will improve what we can but effort will be concentrated on OASIS4.
4. Students found using OASIS was a very steep learning curve.
D
1. Specific to Climate codes
2. Need to adapt each component to it
3. restricted description of grids, only 2D fields exchanges.
CBD: 3D interpolation is supported in OASIS4.
E
1. It can be a cause for poor model performance, since OASIS3 is not fully parallel
CBD: OASIS4 is much more efficient.
2. It can be difficult to track bugs down, especially when they reside in OASIS
CBD: We agree. We will improve what we can but effort will be concentrated on OASIS4.
F
1. complicated handling
2. too slow
CBD: OASIS4 is much more efficient.
G
1. Nothing special that cannot be improved in future versions
H
1. Portable and flexible
I
1. API
J
1. Usage of extra computer resources as memory and CPU time
CBD: We will improve what we can but effort will be concentrated on OASIS4.
K
1. portability bugs
CBD: Please report portability bugs to us oasishelp@cerfacs.fr.
2.standard conformance bugs
3.performance
CBD: OASIS4 is much more efficient.

Improvements:
A
1. Avoid special input files for grids, masks, etc.
CBD: In OASIS3, there is no need to create those files before hand. There are PSMILe routines that can be called in the model which lead to the creation of those files on-line.
2. Support of 3D interpolation
CBD: 3D interpolation is supported in OASIS4
3. Efficient search algorithm
CBD: OASIS4 is much more efficient
B
Rq: There are many improvements that could be brought to OASIS3 but we think it should be frozen and efforts should be put on OASIS4.
CBD: We agree
C

1. Fixing the drawbacks!
2. Making OASIS parallel as for frequent coupling scenarios the coupler takes longer than the model
CBD: OASIS4 is fully parallel

D

I ported OASIS3 on the Cray.
A single comment: in case you need to port for the first time on a new platform I would appreciate to have a short description of the rules to follow like

1- do that
2- update that file
3…
etc….

or even better to have a true Makefile with dependencies and not scripts.
CBD: This should be done with the next release

All that in order to save time.

E

1. suppress "obsolete" coupling methods (PVM, pipe, mem-segments ?)
2. introduce more grids descriptions
CBD: This is addressed in OASIS4
3. introduce 3D fields exchanges
CBD: This is addressed in OASIS4

Still quite inteesting for for reserach projects, teaching coupling methods, etc...

F

1. Provide a fully parallel version
CBD: OASIS4 is fully parallel
2. Provide a data base of users, if it doesn't already exist, so that gained experience can benefit new users
CBD: We added this item on our to-do list

G

1. a more easy handling
2. optimization
CBD: OASIS4 is much more efficient

H

1. Handling of fields defined on time-varying ocean-sea ice masks;
CBD: This will be addressed in OASIS4
2. Extended diagnostics for conservation properties of the exchanged fields;
3. Computational efficiency (if possible)
CBD: OASIS4 is much more efficient
4. More detailed documentation about the handling and construction of specific files (weights, masks, etc.)
CBD: This is described in detail in the documentation. We do not really understand what we could improve.

I

1. Parallelization of the coupler
CBD: OASIS4 is fully parallel

It is of paramount importance for us that OASIS developers maintain their concerns on portability.

We just wonder what the reference CVS server for OASIS 3 is: Cerfacs, Bedano, … ?
CBD: We took great care that the versions on those two servers are exactly in phase, but we understand that this can be confusing.
J
1. Possibility for 3D coupling
CBD: This is addressed in OASIS4
2. Really portable and simple coupling protocols - they seem to be increasingly complicated to use.
CBD: This was our main concern when we designed the OASIS3 PSMILe API based on MPI. Do not hesitate to contact us (oasishelp@cerfacs.fr) for specific problems
K
1. Everything which went into the OASIS 4 design
L
1. Development of flexible multi-purpose pseudo models e.g.
   - to manage IO of a regional model;
   - as a test-bed model for decision taking
CBD: We understand the need but this is beyond our possibilities for now (lack of manpower)

2. OASIS3 will not be developed any further. It will be replaced by OASIS4.
M
1. intrinsic load balance measurement tool
CBD: This is on OASIS4 to-do list.
2. remove binary machine dependend I/O
CBD: This will be done in the next release (beginning 2006)

SECTION 2.4a TOOL INTEGRATION: BETWEEN PRISM TOOLS

Benefits: (of using with other PRISM tools)
A
1. OASIS3 uses PRISM SCE for compiling
2. OASIS3 toy models give an example on how to use the SRE

Both the Unified Model and OPA already have interfaces to OASIS.
B
1. OASIS3 is a fundamental component of PRISM. To our concern, we cannot see any other use of the other PRISM-tools without OASIS.
C
1. Robust and proven technology
2. One standard way to couple climate modells.
D
1. OASIS3 source code is adapted to the SCE. The software is configured to be compiled within the SCE and to run within the SRE.

2. It is possible to visualise the exchange fields within the SRE using PRISM low end visualisation (LE-Graphics).

3. Many PRISM and non-PRISM models have an interface to the OASIS3 coupler

Drawbacks: (of using with other PRISM tools)
A

1. No real interface to modern WEB technologies like XML configuration files.
CBD: This is addressed in OASIS4

B

1. The PRISM tools have been developed to be used together. One of the most important benefits of the PRISM system is, that it provides a consistent solution for all aspects of Earth system modelling. There are no drawback with using OASIS3 in conjunction with other PRISM tools. The only reason to use alternative tools is, that one prefers the other tool.

Improvements: (using with other PRISM tools)

A

1. API
2. Configurability
3. Advanced grids.

SECTION 2.4b TOOL INTEGRATION: BETWEEN PRISM TOOLS AND OTHER TOOLS

Benefits: (of using with other non-PRISM tools)

A

1. For us, OASIS3 is a nice coupler developed elsewhere.

B

1. Robust tool to perform coupled simulations.

C

1. OASIS3 gives a benefit to the model user by its flexibility with regards to components and exchange algorithms without using other PRISM tools.

Drawbacks: (of using with other non-PRISM tools)

A

1. Importance of independence between tools and especially for OASIS.

CBD: We do not understand this remark. OASIS3 can be used with or without other PRISM tools.

B

1. Is lacking different I/O or file formats.

CBD: We do not intend to add any new format; only netCDF is and will be supported.

2. Usage of other tools may depend on the I/O capabilities of the models.

CBD: We do not understand this remark.

C

1. You would need to adapt OASIS3 to meet the needs of an alternative environment.
E.g.: if a component model comes with a different compile and run environment you would have to either adapt the component to the SCE/SRE environment, or to adapt OASIS to the infrastructure provided with the component. Both ways may be equivalent with regard to the effort, however, one would not benefit from the fact that the PRISM software is standadized and used by many other person.

OASIS3 is based on the coupler OASIS2 which was developed well before the PRISM project. Thus it is no problem to use OASIS3 independently.

Improvements: (using with other non-PRISM tools)

Comments:

At IPSL, we are currently using MODIPSL which is an IPSL tool for extraction, compilation and execution of model configuration. In order to implement OASIS 3 in MODIPSL, we had to use and

modify parts of the SCE tool. These modifications are not implemented in the SCE reference version stored on bedano.

3.3. OASIS4 Coupler

3.3.1. Summary of Questionnaire Responses

This section presents a summary of all points that were raised by at least two different institutions. The number of institutions raising a particular point is considered to be a good indicator of its relative importance to the user community. The points are therefore ordered by number of institutions raising it. For clarity, they are further separated into the categories presented in the Questionnaire. The number of institutions that raised each point is provided inside the square brackets.

Tools used previously or currently in-use
[9] OASIS3
[6] OASIS2 (2.2 and 2.4)
[2] Own solutions (e.g. STASH at the Met Office)
Reasons to use or not use
[3] All Oasis3 features (interpolations) must be available in Oasis4 before switching
CBD: This is a high priority in our developments.
[3] Oasis4 not yet (proven to be) stable enough for use
CBD: We are currently working with a reduced number of beta testers. A stable fully public release should be available within a year.
[2] The new features that Oasis4 provides are required
[2] Oasis4 currently used in development not production
[2] Happy with Oasis2/3 at the moment
CBD: We will support OASIS3 as long as it is used in the community, but we strongly recommend switching to OASIS4, once the stable fully public release is available.
Benefits
[6] More efficient (parallel coupler and interpolation)
[4] Improved api
[2] Portability
Drawbacks
[3] Poorer interpolation than Oasis3
CBD: This is a high priority in our developments.
[3] Not yet mature enough
CBD: We are currently working with a reduced number of beta testers. A stable fully public release should be available within a year.
[2] Increased complexity c.f. Oasis3
CBD: We understand this point. The increased complexity is due to the increased number of functionality.
[2] Dependence on other packages
CBD: We try to rely on portable standard packages only.
Improvements
[2] Provide clearer error messages
[2] Complete interpolation development
[2] Simplify xml structure
CBD: Those 3 items have a high priority in our developments
Benefits: (of using with other PRISM tools)
[2] Works directly with SCE, SRE, GUI and Data Visualisation tools

CBD: OASIS4 is not adapted to those other PRISM tools yet!
Drawbacks: (of using with other PRISM tools)
[2] Dependency on SCE and SRE to compile
CBD: OASIS4 is not adapted to the SCE yet and its toymodels are not adapted to the SRE yet!

3.3.2. Collated Responses from Questionnaires

SECTION 1.1 SCE AND YOUR SYSTEM – Reasons for use OR non-use in the future

- All OASIS3 interpolations need to be included in OASIS4 before we can really think of switching from OASIS3 to OASIS4 for our real coupled models.
  CBD: This is a high priority in our developments.
- No plans yet in place for the move from OASIS3 to OASIS4
- No. The results change if I change the number of processors, this is due to interpolation algorithm
  CBD: This is a high priority in our developments.
- I don't know, maybe only a part of OASIS 4
- Only on customer codes, on VPP5000 at meteo France, and eventually at Mercator site (FSE Opteron cluster)
  - May help customer to debug / improve coupled codes at Meteo France or Mercator.
  - If Meteo France, or Mercator development team, uses OASIS4 for code coupling (for climate studies for example), I should also test these codes on VPP and Opteron Clusters.
- Yes. I was not aware it was available. Further delays will be caused by coding the new interface
  CBD: We are currently working with a reduced number of beta testers. A stable fully public release should be available within a year.
- We need a parallel coupler for our future developments. We will use OASIS 4 when the interpolation library of OASIS 3 is implemented and OASIS 4 is fully validated. We are also interested in the new functionality allowing a single interface for forced (off-line) or coupled configurations.
  CBD: The implementation of all OASIS3 interpolation schemes in OASIS4 is a high priority in our developments. The "new functionality" you describe is currently available in both OASIS3 and OASIS4.
- We would like to use OASIS4 in the future, as it is always best to have the latest and most efficient version of a software. However, our priority is currently to run our models and to develop them. And we are satisfied with OASIS2 and OASIS3 versions.
- OASIS version 2.2 has worked satisfactory up until now. We plan to use OASIS4 for earth system modeling, binding together modules of atmosphere, ocean, sea-ice, and vegetation.
- OASIS 4 is currently implemented for the RCA-RCO (RCAO) coupling at SMHI, But is not yet operational.
- Yes in development, not production. We expect to be pushing for (and helping with) various developments:
  New transformers
  IO performance improvements. Data compression for archive purposes.
  Development of the meta-data for better integration with FLUME design
  Improved documentation
  Proposed simplification of the interface, as per FLUME
  But details are not available yet

CBD: We are keen on interacting with any group proposing improvements (but only after the beta-testing phase is over i.e. within a year)

- We will know more when our experience grows. We look forward to finding ways to keep FLUME and PRISM as close as possible and these may through up other requests of both FLUME and PRISM
  CBD: It is in our interest too to keep FLUME and PRISM as close as possible.

- We have concrete plans to use OASIS4 in the very near future, as we need some of the OASIS4 features OASIS3 does not provide (i.e. 3D-coupling, improved efficiency for high frequency exchange of large files, parallel coupler)
  - We will not adapt our running coupled models to OASIS4 unless OASIS4 includes all OASIS3 features (i.e. diverse interpolation and remapping methods) and is proven to be debugged and stable and efficient and easy to use and portable
    CBD: This is a high priority in our developments.
- Insufficient software quality when last tested.

CBD: We think that OASIS4 software quality of the first fully public release will be satisfying (i.e. after the beat testing phase within a year).

OTHER TOOLS IN USE (Previously or Now)

A
OASIS2.4, MpCCI, OASIS3
B
OASIS3
C
OASIS3
D
OASIS version 2 with significant local modifications.
E
Many things
F
OASIS3
G
OASIS 3
H
OASIS2 and OASIS3
I
oasis2.4 a part of ECHOG-model
J
Coupling through UM STASH
K
OASIS3
L
OASIS3
M
Example toymodels (proto_ex …)
N
OASIS version 2.2
O
OASIS 3
P
Oasis2.4

SECTION 1.2 EXAMPLE USE

- For now, we use it only with toy coupled models to test its functionality.
- ECMWF are currently in the process of implementing two sets of coupled system by use of OASIS4: 1) coupling of IFS (our NWP model) and CTMS (chemical transport models) and 2) coupling of IFS and OPA (an ocean model). The work is ongoing.
    - It has to be demonstrated within the next (6) month, that OASIS4 is able to couple IFS and the CTM with sufficient efficiency and accuracy. This implies that OASIS4 can handle the IFS reduced gaussian grid and its domain decomposition.
    - Coupling of IFS with chemical transport models.
    - Coupling of IFS with the OPA ocean model.
- I have one ocean model and one atmospheric model. I run two models together using oasis4 as the main coupler. Both ocean and atmospheric model are simple
- Depends on the examples our customers would like to run on our solutions, presumably atmosphere/ocean/ice as well regional climate models.
- Ocean atmosphere coupled regional climate model runs (development, control and scenario runs)
- UM Atmos to NEMO ocean, under development. This will be integrated enough to support a small development team, later for more general use.
- Coupling of an atmosphere and a chemistry model in a model intercomparison project (GEMS)

SECTION 1.3 TECHNICAL EXPERIENCES

Benefits:

A

1. Open Source
2. Portability
3. Performance

B

1. More efficient than OASIS3
2. More flexible than OASIS3

C

1. Use of MPI rather than files/pipes for coupling.
2. Parallel interpolation.
3. Standardized interface for models to be coupled

D

1. better interface with model components (better layering of the code)
2. configuration files with XML description
3. simple, more general description of grids + operations on coupled fields

E

1. Parallel Coupler
2. New Interpolation Library
3. Portability

F

1. Designed by climatologists for climatologists
2. Better I/O capabilties
3. Better API

G

1. the potential to run faster (parallel) at some later point in the development
2. clearer interface to be implemented in the model code
3. hope to get rid of MPI-related trouble with OASIS2.4 (when passing many fields)

H

1. efficient 3D-, high-frequency exchange of large fields possible
2. can be used on platforms that require that the applications are highly parallel

Drawbacks:

A

1. Use of XML files

CBD: Note that this was also mentioned as a benefit. We think that using XML because of the increased complexity of OASIS4 configuration (related to increased functionality).

2. Debugging the use of the API is difficult

CBD: Improving the documentation and the debugging is on our list of developments.

B

1. All OASIS3 interpolations are not yet in OASIS4.

CBD: This is a high priority in our developments.

2. The model interface is slightly more complex than the one from OASIS3.

CBD: We agree but this is a direct consequence of increased functionality

3. The current PMIOD structure is too complex; it should include only descriptive elements.

CBD: We agree; this is done.

C

1. Lack of support for reduced Gaussian grids.

CBD: This is done now.

2. Poorer standalone interpolation than OASIS3.

CBD: This is a high priority in our developments.

3. More complicated computer resources scheduling.

CBD: We do not understand this remark.

D

1. more complex than Oasis3.... more config files to fill correctly (but more internal checks) : PMIOD, SMIOC, etc...

CBD: We agree but this is a direct consequence of increased functionality

2.now depend on several "ancillary libraries", like libxml, and parallel I/O, so more components to install before use.

3.Need to be used in conjunction with SCE and SRE, so need to have these 3 components well adapted to any hardware/software platform.

CBD: This is not true. OASIS4 is not adapted (yet) to SCE and SRE, and even if it would be, it would still be usable independently.

E

1. Needs detailed knowledge about the software to use it

CBD: Improving the documentation and debugging is on our list of developments.

F

1. We need help to stabilize the package.

G

1. It is not yet really ready

CBD: We are currently working with a reduced number of beta testers. A stable fully public release should be available within a year.

H

1. Lack of some interpolation/remapping methods

CBD: This is a high priority in our developments.
2. Not very mature, not much experience
3. Not used in the community
CBD: We currently have about 4 beta testers: GFDL for a coupled model including MOM ocean model, SMHI for regional coupling, IFM-GEOMAR for interpolation of high-resolution fields, ECMWF for coupling atmosphere dynamics and atmosphere chemistry within the GEMS project.

Improvements:
A
1. Clearer error messages and debugging information
CBD: This is a high priority in our developments.
2. A **small** configure script to create a make.inc
B
1. Include at least all OASIS3 interpolations and other transformations.
2. Support Reduced Gaussian grids
3. Simplify PMIOD XML structure.

The PMIOD structure needs to be reviewed. There are other on-going efforts at the international level (GO-ESSP for example) to define model descriptive XML files. Collaboration with those groups need to be reinforced.
CBD: We agree with those 3 points; they are on our list of developments.
C
1. Support for reduced Gaussian grids.
2. Better documentation.
3. Clearer error messages in case of failure or grid inconsistencies
CBD: We agree with those 3 points; they are on our list of developments.

A clear documentation of the interpolation scheme and its properties (conservation of averages, shape etc.) is needed. Additionally, it might be good if the user can not only chose between different methods but could easily implement its own scheme.

OASIS4 capability to couple on-line and off line (by i/o of netCDF files) is a/ would be a good feature.
CBD: This functionality is included in OASIS4 (and in OASIS4)

There might be a tendency that the ambitious xml control structure imposes a burden on the implementation for a specific coupled system. Although generality is a nice feature, it seems sometimes difficult to extract your specific case from the many options provided.
CBD: We agree, the use of the XML files should be simplified; this is on our list of developments.
D
1. to be more independent of the low level Unix utilities (like m4)
2. May be use ONLY Perl scripts for generation of the SCE and SRE (avoid m4 !)
3. Generates one file with all Fortran parameters using Perl for interpreting the initial XML files: PMIOD, etc. This would simplifiy and accelerate the deployment of the Oasis4 coupler. (no need for linking with libxml). These "parameter files" could be either Namelists (but this may be too constraining and not so easy to evolve in time) or even just "include files" with fortran PARAMETERS (so no need for formatting), that could be automatically included in the sources at compile time.
Even simpler, since most scientists use a PC to work, it exists already several tools in JAVA for interpreting any XML file, (or it is trivial to do). Such a tool should generate, for each coupling experiment with OASIS4, the suitable parameter files used for the experiment (either at compile

time or at execution time). This completely avoids the necessity for a libxml that is specific to each architecture.

Be sure that the install/ test procedures are well "self-contained" and do not do any "pre-requisite" installations. Specifically : Oasis4 need to have "embedded" in a given release ALL the tools, libraries, procedures to make it work on any simple Linux platform (or other systems). Prism should evolve so that the number of such ancillary tools and libraries are minimum.
CBD: We will consider those remarks (besides the ones that relate to the SCE/SRE like the use of m4)
E
1. Implement parallel IO
CBD: This is done
1. After stabilizing performance tuning.
F
1. complete the development of interpolation schemes
CBD: This is a high priority in our developments.

SECTION 2.4a TOOL INTEGRATION: BETWEEN PRISM TOOLS

Benefits: (of using with other PRISM tools)
A
1. The GUI gives a friendly way to build the SMIOCs files
2. Adapting OASIS4 to the SCE will help the users to switch from OASIS3 to OASIS4
CBD: This is in our list of developments.
B
1.OASIS4 needs to be used in conjunction with SCE and SRE
2. compilation / installation is simple , AS SOON AS, SCE and SRE
are fully installed and operational
CBD: OASIS4 is not adapted yet to the SCE/SRE and even when it will be, we will make sure that it remains usable independently.
3. The simple "toys" or "simple models" allow to solve most installation
or usage problems
C
1. PRISM is more compliant with NetCDF/CF.
2. Configurable via XML files.
D
1. assuming that the high level design, i.p. the user aspects are similar to that of OASIS3, we expect that it will fit into the SCE/SRE and data visualisation similar to oasis3.
CBD: Yes, it should be the case.

Drawbacks: (of using with other PRISM tools)
A
1. The SCE platform dependent include files might be a drawback, we would prefer to have OS dependent include files (see PQSCE.txt)
CBD: OASIS4 is not adapted yet to the SCE/SRE and even when it will be, we will make sure that it remains usable independently.
B
1. These 3 Prism tools : Oasis4 + SCE + SRE are interdependents.
They must be installed/ deployed in the right order.

CBD: OASIS4 is not adapted yet to the SCE/SRE and even when it will be, we will make sure that it remains usable independently.

C

1. None.

D

1. we do not anticipate any problems with interfacing OASIS4 with any other other PRISM tool, at least to additional problem compared to OASIS3

Improvements: (using with other PRISM tools)

A

1. Modify the SCE to have OS dependent include files.

CBD: This is a remark for the SCE

2. See Q1.3.3 of PQGuiAndWS.txt

B

The more components, the more complexity and source of potential installation Problems. With the GUI, then Java libraries, Client –Server interactions, PHP, Applets, etc come into play, making the full installation longer and potentially more difficult.

CBD: We understand this concern. It has to be understood that the OASIS4 part of the coupler requires only Java and is not hard to install/use. And it is always possible to use the coupler without the GUI (but then the user has to edit the files by hands).

C

1. Test integration between OASI4 and the GUI.

SECTION 2.4b TOOL INTEGRATION: BETWEEN PRISM TOOLS AND OTHER TOOLS

Benefits: (of using with other non-PRISM tools)

A

1. You can use the compile scripts coming with the package.

Drawbacks: (of using with other non-PRISM tools)

A

1. You do not profit from the PRISM tools

Improvements: (using with other non-PRISM tools)

A

1. Adaptation to the SCE/SRE

CBD: This is in our list of developments.

3.4. SCE: Standard Compiling Environment

3.4.1. Summary of Questionnaire Responses

This section presents a summary of all points that were raised by at least two different institutions. The number of institutions raising a particular point is considered to be a good indicator of its relative importance to the user community. The points are therefore ordered by number of institutions raising it. For clarity, they are further separated into the categories presented in the Questionnaire. The number of institutions that raised each point is provided inside the square brackets.

Tools used previously or currently in-use
[10] Home grown scripts
[5] Systems developed by others (e.g. for Hirlam, UM, Modipsl, PrepIFS)
Reasons to use or not use
[2] Not suitable/feasible for institutions requirements
Benefits
[4] Portability (easy to port to new platforms)
[3] Ease of use
[3] Consistent/standard environment
[2] Only modified files are compiled
Drawbacks
[3] Change of model source directory structure required
[2] Difficult to adapt/debug
Improvements
[2] Support models not adhering to standards
[2] Improve documentation

3.4.2. Collated Responses from Questionnaires

SECTION 1.1 SCE AND YOUR SYSTEM – Reasons for use OR non-use in the future

- The "compiling environment" we use has functioned satisfactory up to now. If we decide to use other components of PRISM, like OASIS4, it seems natural to consider the SCE.
- The "building scripts with other scripts" method is probably difficult to be maintained. One set of templates that can be adjusted for specific needs would probably suffice in our case. During the development phase, it would be more helpful to have one single Makefile and, at the end, to modify the scripts that build it.
  CBD: The "building scripts with other scripts" method was actually chosen in order to minimize the maintenance by the PRISM system developers. Another reason was the need for a seamless interfacing with the PRISM GUI.
  Besides we recommend to generate any compile script at the start of the development work and use that as a template that can be adjusted for specific needs (platforms or models). Writing of new header files and modification of the script that builds the compile script(s) can be done 'at the end'. The latter task can be transferred to the PSI support team.

- The IPSL demonstrations for PRISM were implemented within SCE on the previous PRISM CVS server. We explored the idea of moving to the PRISM SCE for all IPSL

components, starting with NEMO in autumn 2004. These trials were made by engineers of IGCMG (IPSL Global Climate Modeling Group) with the help of Josephine Ghattas, and we came to the conclusion that using this option was not feasible. NEMO is neither distributed nor used with SCE and neither are the others IPSL components
CBD: The main reason for this is the problem with the source code orgaisation as far as I know. This was not the case with the previous version of the models but emerged since the component modularity was reduced instead of extended which is against the PRISM philosophy. Maybe the philosophy has to be changed.

- We want to have a compile environment suitable for a single installation for multiple users. It also needs to be able to reduce the compile time and overheads to improve productivity and fully protect our users from make. The tool we chose should be used by other teams in Research outside the UM to allow staff to work with their colleagues more easily. We can provide details why we think the SCE falls short of our requirement and why we would propose our own tool, FCM. … but we would like to offer FCM to PRISM and think this would suit some PRISM sites.
  CBD: "Instead of trying to foresee the future, we have added features as required and will continue to do so. " [1] was the motto under which the NetCDF/CF conventions were developed and became popular. The SCE developers also had this motto in their mind. So we would be interested to see details why you think the SCE falls short of your requirements. Some of the details will be touched below I guess. In any case will the 'reconciliation' of the different high-level requirements for operational systems at a single centre and infrastructures for research adressing component model modularity be a non-trivial task.

OTHER TOOLS IN USE (Previously or Now)

A
Hand made Makefiles
B
Unified Model scripts
C
We are using system based on make which is integrated into our prepIFS/SMS system.
D
System previously used at MPI otherwise none
E
Custom makefiles and shell scripts.
F
Modipsl , see http://www.ipsl.jussieu.fr/poles/Modelisation/cours_modipsl_2004.blanc.pdf
G
We are using a simple compilation script.
H
standard tools like 'make', 'gmake'
I
UM compile environment, soon FCM
J
- Compile scripts and Makefiles coming with the packages
- home made scripts, not standardised

[1] *J. Gregory (2003)*

K
Standard Makefiles
L
OASIS-toymodels, ECHAM5-MPI-OM
M
The directory structure is based on the directory structure of the atmosphere model ARPEGE. Otherwise the “compile environment” is developed specifically for our model system.
N
GNU configure/autoconf and cmake.
However, those tools were used for standalone models or non-climate applications or packages like VTK.
O
The HIRLAM compilation system

SECTION 1.2 EXAMPLE USE

- We use it to compile OASIS3 in coupling and interpolator-only mode.
- For compiling Climate codes, SCE is much simpler to use than any other system based on Makefiles. It implies internal rules and restrictions that ALL climate codes should follow.... This is unfortunately not yet the case.
- Recompile a coupled model, all or one its components, to perform a new experiment.
- A typical scenario at our institution is the build of a stand-alone model or of a coupled ESM.
- SCE was developed for coupled model systems and therefore it is very complicated and fault-prone. There exists no scenario at our institution, which can be a candiate for using SCE as development tool.
- Almost all models we are using are adapted to the SCE. This includes PRISM and non PRISM models. It makes life much easier for us.
- NEC HPCE is not a research institution, therefore we do not define such scenarios. We are just supporting our customers in running the PRISM-System
- Presumably one of the coupled models integrated in the SCE. Depends on what a customer would like to see on our solutions.

SECTION 1.3 TECHNICAL EXPERIENCES

Benefits:
A
1. Easy generation and use of Makefiles
2. Recompilation of modified files only
B
1. when adapted to a given architecture, SCE is very simple to use and also "safe".
C
1. Simplification and standardization of the compile environment
2. More easily ported to a new platform, due to clear separation of complier options
D
1. It allows you to compile other models without going too much into details
2. It is very useful in the production phase
E
1. specialized for earth system models
F

1. Having the same look&feel for all models we are using.
2. Flexibility to accommodate new models easily. Portability. Fast, parallel compiling of only the minimum number of routines.
3. Relatively small package. Easy to use. Easy to install. Good documentation and support from the developers.

G

1.easy adaptation for different platforms
2.consistent compiler options
3.consistent library use

H

1. Portability
2. Ease of use

I

1. A validated representative selection of models.
2. Clear instructions to compile those models.

Drawbacks:

A

1. Very complex to adapt/debug when it does not work.

CBD: This should not be the case. We have to improve documentation and try to make the steps required for adaptation of a new component/platform more 'logical'.

2. Too rigid, not flexible enough, complex to modify to add options. E.g. I modified the SCE to be able to compile oasis3 and its libraries in single precision; I had to change the following files:
   -> in prism/util/compile/frames/include:
   Guispecif_all.h, Print_par_all.h, Cppflags_toyatm.h, Cppflags_toyoce.h, Cppflags_toyche.h, Cppflags_oasis3.h, Cppflags_libs.h, Comments_libs_frm.h, Command_par_libs_frm.h, Guispecif_libs.h, Check_libs.h, Print_par_all.h
   -> in prism/util/compile/frames/include_<machine>:
   OSspecific_<machine>.h

CBD: As was said above, one should first try to modify existing compile scripts according to the new needs.
The single precision business indeed adds functionality to the system and therefore is somehow more complex than simply including new platforms or models. Changes in 11 header files for the modification of 4 models and a number of libraries does not seem to me to be too much of an effort. This should be implemented now however as a permanent option which requires just the change of 1 parameter. It should be possible to do that through command line parameters. The 'user interface' should allow that. Remains to be discussed whether single and double precision libraries may exist simultaneously or whether it is assumed that a PRISM installation works either in single or in double precision. Note that the existence of more than one installation on the same platform is not a problem. Note also that on many platforms dozens of versions for the same libraries exist (compilers, versions, precisions).

3. Uses a complex directory tree (e.g. the script to compile oasis generated in prism/src/mod/oasis3/ uses a script to compile the libraries generated in prism/util)

CBD: Since we have only one compile script for all libraries whereas for the models we have one compile script for each model, the place of the library compilation script and its output may not be specified as in the models source code branch. This place is now printed to the screen for the convenience of the user.

4. Uses tools not installed everywhere (e.g. m4)
CBD: I don't know of any WS which does not have m4 installed or installable. On compute hosts this has in some cases been a problem. That is why it is possible to generate the compile scripts for a specific compute host on any WS.

B

The main objections to the SCE as it is now are as follows:
1) No support for pre-built libraries. For a large code as IFS we really do not want to recompile any source files not needed.
CBD: The PRISM system knows more than 10 general-purpose libaries. These are non-configurable and can be compiled by the system administrator at installation time. There is a way (already coded in the system) that allows to define each of the libraries to be a non-changeable system library or a library changeable by the user. The use of both local and system versions of the same library is also possible.
2) Too strict requirement for directory layout.
CBD: Yes … but this was the PRISM philosophy: component modularity shall be enforced. And also a common look&feel with all aspects of the system, i.e. also with the source code organisation. Also allowing any directory tree structure makes the scripts more complex (to write).
3) Insufficient support for non fortran codes.

We are following the work do by the Met. Office to adopt their configuration management system to PRISM with great interest. If it proves efficient we might consider using it.

Any change to the compile environment for IFS must be justified in context of data assimilation runs since this is our main activity.
CBD: If the requirements that come with the data assimilation activity are communicated to us we can try to take it into account. I see however difficulties to accommodate the earth system community and operational forecast community with the same tools
There is a potential for using a common infrastructure for compilation (and running) of components for coupling of models to make import of new models easier. Especially for shared developments (in our case for instance ocean models) it is clearly of interest if a common compilation (and running) can be made sufficient easy and efficient to use.

C

1. depend on a specific level of m4
CBD: see comment to A 4., Section 1.3

2. need constant updates in order to follow architectures, compilers and libraries
CBD: There is no need to follow updates for architectures, compilers and libraries as long as one is happy with what is there. Bug fixes in libraries should indeed be followed.

D

1. It can be time consuming to detect errors in the scripts when they occur, for example when modifying the script or porting it to another machine where some required software is not available
CBD: see comment to Section 1.1, first bullet

E

1. Too cumbersome during the developmental phases of model components

CBD: During the developmental phases of model components people can work as usual: use any SCE compile script and change it until satisfied. Only if the component is supposed to become part of the PRISM system has the 'creation of compile scripts from header files' business to be done.

2. Reduced flexibility
CBD: The idea was to give flexibility. Depends perhaps on the definition.
F
SCE was used once to implement the IPSL IGCMG demonstration runs (i.e. a limited set of fixed configurations). Its technical options to generate the Makefiles has been used successfully to increase the speed of parallel compilation for one component.

It has appeared not to be useable for the distribution of the reference versions and configurations of IPSL component models (see J. Ghattas reports). A lot of work remains if we want to use SCE for the list of our coupled and off-line configurations.

For example, the need to change the directory tree structure of source files names to adapt to SCE is not acceptable to us: the organisation of source code is not a technical point but is strongly related to the choices of scientific representation of the phenomenon under study.
CBD: It does not mean to give up 'scientific representation' if a directory package/src is renamed into package_src or the like. The only 'real' problem that occurs is the seamless interaction between the component's 'native' repository and the PRISM software (repository). It was decided in the PRISM project to ask the model owners to provide scripts to do the transformation of directory structures. The hope at that time was that the scientific community would then adapt the structure supported by PRISM.
The PRISM source code organisation was thought to stress component modularity, i.e. help to replace component models in coupled models and exchange components between centres/scientists.
We would like the shared tools to be "components and ESM compliant", whereas PRISM asks component models to be "PRISM compliant".
CBD: Indeed was PRISM about unifying component models (with the exception of the science) in order to facilitate the exchange of ideas and models. This does not seem to be (and has never been) popular in the community.
G
1. not well tested in all possible development environments
CBD: I hope the developers of the SCE are not supposed to test the tool 'in all possible development environements'. It was tested on a quite large number of platforms during the PRISM project phase however. Now it has to be tested by the users.

2. not robust enough
H
1. a little overhead, if you are only using one or two models on the ever same machine.
CBD: The overhead will always remain, however there may be room for improvement.
I
1.insufficient split of site and system dependent parts
CBD: Indeed are the system dependent parts duplicated for all sites. This allows script generation on machines the script will not run on.

J
1. To include new models into the SCE structure

Improvements:

A

1. To be able to easily compile a model and its library with a new
   CPP flag without having to modify too many files

CBD: Each model has its own set of CPP flags. So this has to be changed once for each model involved unless the flag is supposed to be used by all components. In that case it can be set in either the header files used for all models or the platform header files.

2. In general, not try to cover all possible cases but provide a simpler
   tool easier to modify

CBD: We try indeed to cover all cases if they make sense and have the potential to be useful for more than one user. Maybe it is possible to hide more of the options in the 'script generation part' and make them visible to the user only if really needed. This is done e.g. with the platform dependent script code.

3. Have OS dependent include files and not platform dependent include files
   (even if we know that generic OS dependent include files will have to be
   somewhat adapted for a specific platform); this is especially needed for
   Linux station and we know it is in development.

CBD: It is done but not yet officially released. A SUN WS version is in work.

B

1. suppress m4 dependence, replace by Perl ?

CBD: m4 is used in only one place. It can be replaced by perl however this would not add anything to the system. It is not high up in the priority list.

2. Use only Perl as "ancillary" utility.

CBD: If the above will be done perl, perl is indeed the only "ancillary" utility used by the SCE.

3. if possible, reduce the number of files in the installation directories

CBD: The number of files is high due to the large number of models that are adapted.
It is planned to have subdirectories for the model dependent header files similar to what is done for the platform dependent header files. This will hide more files to the user. Another reason for the large number of header files is that we do not want to have any redundant script code. This is a maintenance requirement.

Avoid to depend on many Linux utilities : m4, perl, configure, etc...
Use only : perl and make if possible.

CBD: See above

C

1. Support and better documentation on how to modify the compiler environment

CBD: yes, a step-by-step guide is needed.

2. Support on porting to other platforms, including documentation on useful compiler settings etc.*CBD: Support has simply to be asked for. 'Useful compiler settings' can be found in the system dependent header files. For the Linux WS setup we support the Lahey and Portland compiler.*

D

1. Some parts of the structure layout should be "suggested", and users should not be forced to adhere to them.

CBD: see above

E

1. further increased automatisation.

CBD: It is planned to fully generate the Makefiles, not only the prerequisite lists of the targets.

2. making the use possible even for model which do not comply to the standards

CBD: There is work in progress to account for models have have code that shall not be compiled or whose compile prerequisites may not be automatically detected since they do not obey the coding rules. Any relief of standards makes automatisation more difficult however.
3. improve the user 'interface'
CBD: This is planned. However we do not intend to go so far as to simulate a configuration GUI in the scripting system.
F
1. split of site and system dependent parts
CBD: They are split.
G
1. Documentation needs improvement (README files should be sent with the SCE)
CBD: OK
2. Extend the SCE for new components
CBD: This is done on request
H
1. In certain areas one has to re-spin the way of doing certain steps (using cmake?)
CBD: I don't know what re-spining and cmake is. In any case don't we start to work until a modification is really needed (see citation at the end of Section 1.1) .

SECTION 2.4a TOOL INTEGRATION: BETWEEN PRISM TOOLS

Benefits: (of using with other PRISM tools)

A
1. The PRISM models are adapted to the SCE
2. There is a well defined interface between the SCE and the SRE
3. PRISM source code management is based on the SCE
B
1. Independence from other PRISM tools

Other PRISM tools like visualization are currently not part of the SCE.
Is there an intention to integrate those tools?
CBD: No, the visualization tools come with their own make tools which we recommend to use.
Drawbacks: (of using with other PRISM tools)

A
1. There are no drawbacks since the SCE was developed with regard to the other tools

Improvements: (using with other PRISM tools)

A
1. an installation package which installs all PRISM components (SCM/SCE/SRE/VIS/ARCH) on a specified platform may make sense
CBD: Each part of the PRISM system shall be usuable independently of the others. Also the installation shall be possible for each part separately. So I don't see the big benefit from just one package.
In any case will we not provide anything to install the visualisation tools.

SECTION 2.4b TOOL INTEGRATION: BETWEEN PRISM TOOLS AND OTHER TOOLS

Benefits: (of using with other non-PRISM tools)

A

1. the SCE gives the user a common look&feel when compiling any model;
    - it is set up to be used with standalone-models and then does not require
    - the availability of any PRISM software (coupler or libraries);
    - it is easily extended to other platforms and is documented;
    - it uses the commonly accepted GNU software e.g. requires recompilation of the minimum only;
    - it can be used with multiple processors if available;
    - it does not requires any change of a model code, just the source code organisation,
    - therefore the adaptation of a model to the SCE is without any risk;

B

1. It collects a representative selection of models into one compile environment.

Drawbacks: (of using with other non-PRISM tools)

A

1. You do not profit from the already realised interfacing with other PRISM tools

B

1. Presumably other tools are not integrated into the SCE.

CBD: If only a subset of components is compiled in the SCE, this does not prohibit the use of the other tool e.g. the SRE. The default place and names of executables are set in the SRE as they are set in the SCE. However this can be changed in the setup phase of the experiments.
It is then in the responsibility of the user however, that all components are compiled with consistent options.

Improvements: (using with other non-PRISM tools)

A

PRISM was focused on standards. What does integration of other, non-PRISM mean? Extending the PRISM SCE with tools that do not obey, say, NetCDF/CF standards but special binary file formats or HDF5? This leads to Nirvana.

CBD: Maybe, however this is the ultimate goal of many people …

3.5. SRE: Standard Running Environment

3.5.1. Summary of Questionnaire Responses

This section presents a summary of all points that were raised by at least two different institutions. The number of institutions raising a particular point is considered to be a good indicator of its relative importance to the user community. The points are therefore ordered by number of institutions raising it. For clarity, they are further separated into the categories presented in the Questionnaire. The number of institutions that raised each point is provided inside the square brackets.

Tools used previously or currently in-use
[7] Home grown scripts
[5] Systems developed by others (e.g. for Hirlam, UM, Modipsl, SMS)
Reasons to use or not use
[5] Existing System fits requirements
Benefits
[3] Ease of use
[3] Portability
[2] Consistent/standard environment
[2] Standard directory structure
[2] Easy to incorporate new models
[2] Easy to install
Drawbacks
[2] Difficult to adapt to new platforms
Improvements
[2] Interface with a GUI
[2] Further integrate with post-processing and visualisation

3.5.2. Collated Responses from Questionnaires

SECTION 1.1 SRE AND YOUR SYSTEM – Reasons for use OR non-use in the future

- Yes, for the OASIS3 toy coupled model only. We think it would be too complex to use the PRISM SRE for our other coupled models as we currently have a running environment that fits our needs.
- Not unless the Unified Model uses this.
- ECMWF developed SMS (Supervisor Monitor Scheduler) system for monitoring and running and prepIFS for launching.
    - Our current system has been adapted to our requirements through several years of development. We do not believe that we will benefit from using the PRISM SRE.
    - There is a potential for using a common infrastructure for running (and compilation) of components for coupling of models to make import of new models easier. Especially for shared developments (in our case for instance ocean models) it is clearly of interest if a common running (and compilation) can be made sufficient easy and efficient to use.
- As with SCE : at Meteo France or Mercator Ocean.

- Similar considerations as for SCE. Particularly, it is difficult to run and organise several experiments without duplicating all the necessary files. The entire SRE is rebuilt all the time, and this is not always needed.
  CBD: I am not sure I understand your point. For each experiment you are running a new directory is generated, named as the experiment. This experiment directory contains all input and output needed for the specific experiment. If you are running several similar experiments using the same initial data this leads to some data duplication. However, the data is duplicated only in the medium term file system ($data) not in the archive. Input data no longer needed on this file system will be removed.
  The output data and restart files are archived for each experiment you are running. We think this is mandatory and should happen automatically (unless you specify in the setup that no archiving is wanted).

- SRE was used only once for the PRISM demonstration runs (work of Marie-Estelle Demory). It appears that SRE is able to chain and control the timing of ESM experiments and to store output files.

    - With MODIPSL, we are presently able to perform the same tasks as well as being able to drive the post-processing, generate the graphics and monitor the simulation. For us, using SRE, in its present form, will mean taking some steps back in a lot of areas.
      CBD: You are right that maybe the SRE is not mature enough to replace your system. However, the SRE also contains post-processing tasks. And we are working on automatically generated graphics to monitor running experiments. So maybe at one point the SRE will become interesting for you. Please let us know.
- Yes, in the case of PRIM models
- Our current SRE has worked satisfactory until now. If we decide to use other components of PRISM, like OASIS4, it seems natural to consider the PRISM SRE.
- SRE is my first contact with a runtime environment for **climatologists.** However, SRE looks very much like the scripting environment for structural mechanics or ab-initio codes.
- SRE is not used at SMHI, because compatibility of the RCA climate model with the related HIRLAM weather forecast model has been given priority. Initial intentions to use both proved to be unrealistic. We do not have resources to operate both.
- What we have fits our needs very well, not immediate benefit of change.

OTHER TOOLS IN USE (Previously or Now)

A
Hand made scripts.
B
Standard Unified Model scripts
C
ECMWF developed SMS (Supervisor Monitor Scheduler) system for monitoring and running and prepIFS for launching.
D
The environment/scripts developed at MPI.
E
Custom shell scripts
F
MODIPSL
G

standard scripting in unix-shell environment

H

UM scripts

I

- Run-scripts coming with the packages
- home made scripts, not standardised

J

Own scripts based on old PRISM version.

K

ECHAM5/MPI-OM

L

The simultaneous running of several components in our coupled system is controlled by OASIS version 2.2. The scripts for data integration and post-processing is based on those for running the standalone atmosphere model ARPEGE.

M

A home-made environment related to HIRLAM

SECTION 1.2 EXAMPLE USE

- We use it to test new releases of Oasis3 and its toy coupled models on different platforms with different communication techniques.
- To perform a set of ensemble simulations, either started from varying initial conditions or by apply intial perturbations
    - To set a IPCC scenario. i.e., Define the model components, and then the forcing and initial data, and the length of the run
    - Perform sensitivity simulations in which the models are slightly changed, for example in the method of coupling.
- We used SRE to produce the simulations during the PRISM project. We are still using it for ESM simulations with several components.
- The combinations collected currently in the SRE.
- Almost all models we are using are adapted to the SRE. This includes PRISM and non PRISM models.
    - The brick box design makes it a good tool for use in model intercomparison projects

SECTION 1.3 TECHNICAL EXPERIENCES

Benefits:

A

1. Have an easy-to-install running environment for the toymodels on different platforms.
2. Benefit form a standard directory tree.

B

1. Well adapted to interactive or batch systems
2. When installed, simple to use
3. Oriented to production

C

1.Standardisation of the run time environment

2.Directory structure is improved and more logical

3. Simplification of the procedure for setting up new model simulations

D

1. Useful and complete logging capabilities.

E
1. Portability
2. Ease of use
F
1 A collection of representative models which are validated with a couple of input decks.
2. Reflects to a certain extend the workflow of a climatologist
G
1. Having the same look&feel for all models used with the SRE.
2. Flexibility to accommodate new models easily. Portability.
3. Relatively small package. Easy to use. Easy to install. Good documentation and support from the developers.
H
1.portability
2.single scripts

Drawbacks:
A
1. The directory tree is too complex, e.g. it is difficult to identify the files used to run oasis3 toyclim coupled model (for example, the grid and restart files are in prism/data/toyclim/input_toyclim_standard_standard_prism_2-2.tar.gz and the configuring file namcouple is in prism/util/running/adjunct_files/oasis3)
CBD: We tried to keep the directory structure as clear as possible. There are two different places for input files: the data directory for binary input data and the adjunct_files directory for namelist-like ASCII input files. Like this, the namelist-like files profit from the version control system as the source code and the scripts. We are not happy with the way the data files are kept at the moment (as gzipped tar-files in the cvs repository). We are working on a better solution.
By the way: The clear directory structure was mentioned several times as benefit of the SRE.

B
1. Not so easy to change
CBD: What would you like to change? The way how to include a new model to the SRE and how to port the SRE to a new platform is described in detail in the prism report on the SRE (http://prism.enes.org/Publications/Reports/index.php). Also included in the report is a step-by-step instruction on how to proceed.
2. Environment was too close to DKRZ environment.
Other sites may have less steps (for example no cross compiling) or very different ways of doing archiving. So the generation step must be versatile enough to define a specific "workflow" for a given site.
CBD: Some people at the DKRZ find it too close to IPSL or to other centers. In fact, we developed an environment that can be run as well on a linux PC with just one file-system as in a super computing environment with different file-systems and/or machines for compilation, calculation, pre-/post-processing and archiving.
We tried to hide this complexity from the user as much as possible. The next release of the SRE will contain site-dependant functions. So the run-scripts will no longer contain any features that are irrelevant at the specific site.
C
1. Can be time consuming to adapt to another platform or add another model
CBD: We tried to design the SRE in a way, that adapting new models and porting the system to another platform is as easy as possible. All model and site dependant features are clearly separated and labeled. Creating the model dependant include files for a new model should be much less work than writing a whole run-script for this model.

Step-by-step instructions on what needs to be done to include new models or to port the system to another platform are given in the PRISM report on the SRE (http://prism.enes.org/Publications/Reports/index.php).

2. Can be difficult to track bugs in script generation when errors occur
CBD: Please, do not hesitate to contact us if you experience problems with script generation. Usually, they are resolved easily.

D
1. Limitations when running several sensitivity experiments.
CBD: I do not really understand what limitations you are thinking of. To run several similar experiments you should copy the setup-file you created for the first experiment. Then, modify the setup according to the new experiment, change the experiment-ID and create the new tasks. The different experiments will be running in independent directories and will not disturb one another.
2. Sometimes difficult to integrate with local environments.
CBD: What do you mean with the local environment? Do you think it is difficult to port the SRE to another platform? Then the PRISM report on the SRE (http://prism.enes.org/Publications/Reports/index.php) might be helpful. It includes a section on porting and step-by-step instructions on how to proceed.

E
1. None
F
1. PRISM graphics and postprocessing need some integration
CBD: I agree. We are working on that.
G
1. A little overhead, if you are only using one or two models on the ever same machine.
CBD: This is true. However, we try to hide the complexity as much as possible. The next version of the SRE will bring further improvements in this respect.

H
1.SRE defined functions expanded in runscripts.
CBD: This will be changed with the next version of the SRE.
2.difficult handling of non-default experiments
CBD: Why is that difficult? The setup-file contains all configurable variables of an experiment. Just change these variables according to your experiment and create the tasks.
If you are thinking of non-default source code, you should provide the executable with a suffix of your choice and define the setup variable $atmvers / $ocevers accordingly. Then the experiment will be run with the non-default executable.
If you want to change the namelist for one of the models, you should invent a new variable that is defined in the include file config_model.h and is used for the generation of the namelist.

3. site dependency
CBD: What exactly do you mean? Of corse the SRE contains site-dependant features. We tried to separate all site-dependant and non site-dependant features to make the tasks as clear as possible.

Improvements:
A

We think that running a coupled model is a very complex task; trying to standardize a whole running chain that would work for all coupled models on all platforms and in all configurations is not desirable because it would necessarily be either too rigid or too complex.
CBD: This is opposite to the PRISM philosophy. Running a coupled model is indeed a complex task. Having an environment in which all models you are using can be run in a standardised way on a variety of different platforms, makes this task a lot easier.
To avoid misunderstandings: We do not want to urge the whole earth system modelling community to use the SRE. The SRE is an offer for all those, working with several models on several machines and who have experienced the limitations of the environment they have been used so far.

B
1. concentrate in a few variables all site-dependent settings, at the beginning of one script, so all parts are then generated correctly, without the need to check all scripts.
CBD: This is our design. The setup contains all configurable variables, including the site-dependant variables. It is included in all scripts.

C
1. Support and documentation of how to modify the run environment
CBD: Please read the PRISM report on the SRE (http://prism.enes.org/Publications/Reports/index.php). If you need further support, please do not hesitate to contact us.
2. Support on the porting of the run environment to different platforms
CBD: The PRISM report on the SRE includes a chapter on how to port the SRE to another platform. A step-by-step instruction is included.
D
1. None
F
1. GUI integration needs some stabilizing
CBD: The SRE is designed in such a way that the GUI can be integrated easily. However, the installation of SMS is not a local task but has to be realised by the super-computing centers. For the moment, the interest in using the SRE in connection with the GUI seems not large enough at any center.
2. PRISM graphics and postprocessing need some integration.
CBD: We are working on that.
G
1. further increase interconnection to pre- and post-processing as well as visualisation tools.
CBD: We do!
2. interfacing with a GUI would be helpful
CBD: See F 1. just above

SECTION 2.4a TOOL INTEGRATION: BETWEEN PRISM TOOLS

Benefits: (of using with other PRISM tools)
A
1. The SRE works fine for Oasis3 toyclim coupled model.
B
1. SCE is part of PRISM and, forcibly, has to be integrated. [? Ed.]
C
1 .Representative set of models together with input decks and scripts which reflect the work flow to run those models
D

1. The PRISM models are adapted to the SRE
2. There is a well defined interface between the SCE, the low-end visualisation (in work) and the SRE

E

1.independent of other tools

Drawbacks: (of using with other PRISM tools)

A

1. Finding the correct answer of a coupled run.

CBD: I am not sure I understand this.

B

1. no drawbacks

Improvements: (using with other PRISM tools)

A

1. Use a less complex directory tree.

CBD: The clear directory structure was mentioned also as a benefit of the SRE. However, we are searching for a better solution for the input data.

B

1. Improve the way of validation of a coupled run.

CBD: We developed a test suite to automatically setup and run the different coupled models and to compare the output with reference output files. It is checked only, whether or not the output is exactly the same.
It would be useful, to find a standardised method to test whether or not the output generated on different machines (or with different compilers) stays as similar as one can expect.This would be an interesting task for the PSI.

SECTION 2.4b TOOL INTEGRATION: BETWEEN PRISM TOOLS AND OTHER TOOLS

Benefits: (of using with other non-PRISM tools)

A

1. For other tools might be a way of adaptation to a kind of standard.

B

1. it is not needed to use the SRE in connection with the PRISM coupler or the SCE. E.g. the SRE is enabled to be use with standalone models without OASIS. Also it can be used with executables created with other than the SCE. The SRE is ported to a number of platforms (see tables in http://prism.dkrz.de/Workpackages/WP3i/ -> 'Assembled Models') and easily adapted to new platforms, which is well documented.The benefit comes from using the SRE with a number of models and on different platforms by the common loo&feel it provides.

Drawbacks: (of using with other non-PRISM tools)

A

1. .Not yet integrated

B

1. You do not profit from the PRISM tools; if it is used with only one model on only one platform, the benefit becomes less obvious then the drawback of the overhead.

Improvements: (using with other non-PRISM tools)

A

1. Maybe, a document like a quick reference describing how to adapt to the SRE.

CBD: This document exists. The PRISM report on the SRE contains the sections “Adding a new Coupled Model” and “Adaptation of the SRE to a new Site”.

3.6. Data Processing and Visualisation Tools

3.6.1. Summary of Questionnaire Responses

This section presents a summary of all points that were raised by at least two different institutions. The number of institutions raising a particular point is considered to be a good indicator of its relative importance to the user community. The points are therefore ordered by number of institutions raising it. For clarity, they are further separated into the categories presented in the Questionnaire. The number of institutions that raised each point is provided inside the square brackets.

Tools used previously or currently in-use (There are many!)
Reasons to use or not use
[3] Existing Tools fit requirements
Benefits
[2] Scripting support (as well as GUI)
[2] Netcdf compliance
Drawbacks
[2] Lack of portability
[2] Currently unstable
Improvements
[2] Improved documentation
Benefits: (of using with other non-PRISM tools)
[2] Support for netcdf
Reasons to use or not use
[3] Existing Tools fit requirements
Benefits
[2] Scripting support (as well as GUI)
[2] Netcdf compliance
Drawbacks
[2] Lack of portability
[2] Currently unstable
Improvements
[2] Improved documentation
Benefits: (of using with other non-PRISM tools)
[2] Support for netcdf

3.6.2. Collated Responses from Questionnaires

SECTION 1.1 DP and VIS AND YOUR SYSTEM – Reasons for use OR non-use in the future

- No. Because we already have satisfying tools that we use and know well.
- The Unified Model doesn't output data in CF compliant netcdf although there are tools to convert the data. CDAT is only just being introduced into NCAS and so there is little or no experience with this package.
- Metview and magics are developed within ECMWF and closely integrated within our infrastructure and adopted to our needs. We see no reason to learn a new tool.

- During the PRISM project, we had investigated and proposed an alternative solution for the high-end graphics needs (based on VTK). We feel it has more potential than the OpenDX solution and that it was not properly assessed by the working group. The OpenDX demonstration remains unconvincing to us
- If I am convinced it is an advantage to use it (for the time being I don't know the softwares + I don't know about anyone using it).
- I could not find the download site in the web!
- User unfriendly, no scientific need for this.
- We are quite happy with the data processing and visualization tools we are using, but will consider the many interesting approaches incorporated in PRISM, especially if we decide to use other components of PRISM.
- Running examples and acceptance by big centres would increase the acceptance in our small climate modelling group (Rossby Centre)
- This is a rather difficult question to answer. We will soon have a new member of staff to look at strategy for data processing and visualisation and we will no more when they have analysed the situation. We have a lot of investment in user code and it will be a large effort to move to any new system.
- the CDAT software is since ever difficult to install; on each platform I know of, one or more aspects do not work; e.g. on our data processing platform (Linux ia4) no file or hardcopy output can be done;
- We plan to test it.

OTHER TOOLS IN USE (Previously or Now)

A
Ferret and IDL for visualisation
STATPACK (from LOCEAN, Paris) for statistical analysis
NCO for data post-processing
B
A plethora of different tools eg VCS, IDL, Ferret, Grads, Matlab
C
Metview and magics (developed at ECMWF).
D
NetCDF/CF, Ferret and FAST/ATLAS (IPSL tool).
E
Matlab + VTK (visualization); Unix + Fortran + NCO (processing)
F
for visualisation: grads, with self designed xgrads GUI, gnuplot, xmgrace, vis5d, ... for postprocessing: pingo, cdo
G
CDAT4, COCO
H
For data processing we mainly use Fortran programs combined with scripts, nco, and Matlab. For data visualization we mainly use Matlab, Tecplot, Ferret, and ncview.
I
AVS, IRIX Explorer, ncview, SGI OpenGL tools
J
Home-made tools of various kind, and cdo, ferret, matlab, IDL, ncview
K
we use coco a bit, PV-Wave
L

GraDS
M
Afterburner, grads, CDOs, Ferret, IDL, NCAR-Graphics...

SECTION 1.2 EXAMPLE USE
A
Visualization of the coupled runs which are currently integrated in the SCE/SRE
B
Coco is being used to do means such as "mean January surface temperatures from a 200 year run" etc for all months and for a number of field types. It is built into scripts that find the data, restore from archive and do the meaning process and re-archive the data. Coco was used as a framework to convert our own data format (pp-format) in NetCDF/CF/CMOR files.
C
- the use of the software in model inter-comparison projects in order to arrive at the 'same' plots;
- mass production of plots e.g. for the IPCC scenarios; (this requires that the use is interfaced with the SRE)
- visualisation of Cera (oracle data base) downloads at download time for clients;

SECTION 1.3 TECHNICAL EXPERIENCES

Benefits:
A
1. Open software without license
B
1. NetCDF/CF compliant
2. Integrated in CDMS/CDAT
3.Scripting level as well as high-end interactive look-and-feel
C
1. Scripting languages are fast to develop
2. A small move towards standard file formats
D
1. irregular grids are possible
2. GUI or script based control is possible
3. can be used together with a number of NetCDF file processing tools(e.g. cdms, cdutil,cdtime)

Drawbacks:
A
1. Not very transparent
2. Portability difficult
B
1. Needs to stabilize and better documented
C
1. Developing software around CDAT is not easy for some (many) people
2. OO may be hard for scientists to use well, not yet sure.
D
1.installation is difficult; maybe it is not portable no portable installation package is provided to my knowledge
2. there are some 'unstable' routines which make the tool crash from time to time

Improvements:
A

1.Detailed documentation of the tools

B

1. More examples
2. An release notes mentioning the versions of packages mandatory to run those data processing and visualization tools

During verification of coupled runs the tool suite for data and visualization suffered from regressions within the packages used to build the tools. Packages having a badly needed feature for on PRISM tool did not work with other PRISM tools due to regressions in parts of the source suspected to run stable.
There was even a dependency on the Linux version

C

1. Better documentation
2. Immature code, likely to need generally improving and debugging

SECTION 2.4a TOOL INTEGRATION: BETWEEN PRISM TOOLS

Benefits: (of using with other PRISM tools)

A

1. At least low-level tools are using the same frame work (Python, CDMS/CDAT).

B

1. Common file format between oasis IO and coco

C

1. installation is difficult; maybe it is not portable no portable installation package is provided to my knowledge
2. there are some 'unstable' routines which make the tool crash from time to time

Drawbacks: (of using with other PRISM tools)

A

1. One has to stabilize the tools

B

1. There was little attempt to develop coco interfaces and documentation
in line with the rest of PRISM, I guess this could have been better.

C

1. portability (lack of)

Improvements: (using with other PRISM tools)

A

1. Stabilize and pay persons to test the interaction with other PRISM tools.

B

1. Common documentation standards

C

1. ease of installation; portability

SECTION 2.4b TOOL INTEGRATION: BETWEEN PRISM TOOLS AND OTHER TOOLS

Benefits: (of using with other non-PRISM tools)

A

1. NetCDF/CF
2. Python language can interface with other languages and packages, (See Python binding of VTK!)
3. Even High-End graphics with OpenDX can be interface with Python.

B
1. It only relies on a common file format
C
1. is able to use and interprete NetCDF files
2. interfacing to the above mentioned processing tools

Drawbacks: (of using with other non-PRISM tools)
A
1. I suspect the CF standard is too open to ensure one bit of software can cope with all CF files. Hence CMOR
B
1. always installation and portability

Improvements: (using with other non-PRISM tools)
A
1. First find a reasonable selection of other tools, means it does not make sense to integrate tools which rely on file formats wich are at their end of live cycle.

3.7. Graphical User Interface and Web Services

3.7.1. Summary of Questionnaire Responses

This section presents a summary of all points that were raised by at least two different institutions. The number of institutions raising a particular point is considered to be a good indicator of its relative importance to the user community. The points are therefore ordered by number of institutions raising it. For clarity, they are further separated into the categories presented in the Questionnaire. The number of institutions that raised each point is provided inside the square brackets.

Tools used previously or currently in-use
[8] Home grown solutions
[2] UMUI
Reasons to use or not use
[2] The GUI creating a SMIOC is/would be useful
Benefits
[2] Usable locally and remotely
[2] Ease of use

The above summary is limited due to the relatively small number of responses for this Section. It may therefore be as (or more) insightful to examine the collated responses directly, which are presented in the following Section.

3.7.2. Collated Responses from Questionnaires

CBD: There is some confusion in the responses to this survey. Several users have mixed up visualisation tools with the GUI/Web services tools. Another comment was that the tools were incompatible with a batch queuing environment, which is simply not true (most of the jobs submitted and monitored at ECMWF with these tools runs in batch). In general it seems that few users has any real knowledge about what the tools in this work package can (and can not) do.

From the result of the survey it is clear that there are very few users outside ECMWF actually using or planning to use the prepIFS/SMS system. Some users had been trying to install it without success whereas for other users there was little or no interest in using these tools. Lack of time seemed to be the main reason for the lack of effort/interest. Several users mentioned that they are quite interested in being able to control the coupled setup (OASIS4 and the involved models interface to OASIS4) via a GUI.

The conclusion from the survey (and the discussions at the PRISM community meeting) has to be, that the GUI tool (prepOASIS) to configure OASIS4 is for most users of most importance within the next year or so. There was some interest among the users to use prepIFS/SMS to setup, launch and control experiments sometime in the future but not at this time..

SECTION 1.1 GUI and WS AND YOUR SYSTEM – Reasons for use OR non-use in the future

- First, the tool has not been available for quite a while. Then installation of the GUI on our local system failed.

    - For my purposes a GUI is not required to launch and monitor applications. A GUI to generate the OASIS4 SMIOC input files will be helpful and will be used in future, when available.
- No, we do not really use it but we participated only in the development of the extension of the GUI to support the OASIS4 coupler and have a general idea on the functions of the other parts of the GUI.
    - We plan to use the GUI to create the XML configuring files for OASIS4 (SMIOCs). We would like to have a Graphical User Interface to build also the XML description files (PMIODs); this is not currently not covered. We would like to use the GUI and web services to follow our simulations locally and/or remotely on different sites (ECMWF, Meteo-France, IDRIS, ...) if the tool is installed on those sites. In that case, we are ready to put the effort to install the tool here at CERFACS.
- NCAS would not use this tool until the UK Met Office use this tool for their model.
- Nothing for this kind of task. This kind of tool is certainly not our priority for the next five years.
    - We have a large number of configurations with new ones being developped regularly. A number of them use stand alone components. We are interested in a single how to use procedure.
    - We use a large number of computer centers (from Supercomputers to PC) and we have to be able to install tools without root permission in a distributed environment.
    - We have to keep the possibility for each user to modify text file before launching a new simulation.
- we will test it.
- No. The tools are not public available
- Our current system is not well developed, so the graphical user interface and web services of PRISM will be considered.
- Depending on support for our customers and licenses we could install the GUI on our inhouse platforms. However, if there is no business case we are not undertake the porting effort.
- This would mean much implementation work for our small group. Currently, the majority of the group is not favoring a GUI. We wait and see if the bigger centers really start using this in their production runs.
- Hope so, to be evaluated. We would like to have a common tool. The PRISM GUI is very well designed but is maybe not quite as flexible in some ways as our own user interface. Our review of PrepIFS is now quite old and should be refreshed. We would like to cooperate on a common tool in FLUME.
- We do not have enough persons to support the GUI. It still needs development or improvement, i.p. protability and installation. We might come back to the GUI software uasge in the longer run, i.p. we are looking forward for the OASIS4 supporting GUI.
- No. High batch production rates (for which a GUI is unsuitable)

OTHER TOOLS IN USE (Previously or Now)

A

My own scripts

B

vi and emacs

C

Unified Model User Interface

D
We have been using various versions of prepIFS for many years.
E
Paper printed user's guide.
F
grads, vis5d, gnuplot, xmgrace
G
we are using NEC Tools for submitting jobs
H
We have used a locally developed web interface for analyzing and displaying data.
I
Scripts of the SRE are okay.
J
UMUI GUI
K
Run time environment, batch production

SECTION 1.2 EXAMPLE USE
A
Setting up and launching of research experiments with IFS (either coupled or uncoupled).
B
We used the tool to setup, run and monitor example experiments within the PRISM project phase. It has the potential, once being installed, to speed up the setting up and execution of experiments by unexperienced persons through preventing misoperating.

SECTION 1.3 TECHNICAL EXPERIENCES

Benefits:
A
1. Help the user build his/her XML files
2. Follow a simulation on a remote or local site
B
1. Ease of setting up experiments.
2. Ease of constructing new experiments based on previous (*e.g.* control/test situations)
3. It can be used both locally and remotely.
C
1. Ease of use for newcomers compared to the unsage of Unix scripts.

Drawbacks:

A
1. There is no interface to build the AD and SCC XML files
B
1. It can be a bit slow sometimes.
C
1. It is an extra layer in complexity of the system. This makes debugging and testing the system more complicated
2. The software needs to be maintained. Web server must be provided.
3. World wide access to the supercomputer is connected with security problems

Improvements:

A
1. Include a tool to help the user create the PMIOD XML files
2. Include a tool to help the user create the AD XML files
3. Include a tool to help the user create the SCC XML files based on the ADs.

We consider that a GUI can help us to build error-free XML files, and help us to submit and follow a job. We therefore consider that a GUI can help describing, configuring, and following a run of an EXISTING coupled model, but cannot really be used to assemble a coupled model, as this task requires fine and specific developments in the model codes themselves.
B
1. Faster response when loading experiments.
C
1. Increase automatisation of installation
2. Improve the design of the GUI; dont show windows where nothing should be put in; interface with some sort of help functions

SECTION 2.4a TOOL INTEGRATION: BETWEEN PRISM TOOLS

Benefits: (of using with other PRISM tools)
A
1. Help build the OASIS4 XML files
2. Submit and follow a coupled run using the OASIS3 or OASIS4 coupler.

Drawbacks: (of using with other PRISM tools)
A
1. additional complexity; requires many software to be installed

Improvements: (using with other PRISM tools)
A
1. enable installation locally on WSs

SECTION 2.4b TOOL INTEGRATION: BETWEEN PRISM TOOLS AND OTHER TOOLS

Benefits: (of using with other non-PRISM tools)
A
1. if the GUI is to be used with compile and run infrastructures different from the SCE/SRE it would require more time to adapt it to that infrastructure then using the SCE/SRE since that is already done for a number of models. This relates to the use of OASIS3 in the SCE/SRE only. We dont know how the GUI for OASIS4 works.

Drawbacks: (of using with other non-PRISM tools)

No responses.

Improvements: (using with other non-PRISM tools)

No responses.

4. Points Made During Discussions

This section contains additional points that were raised during the visits to institutions as part of the survey. Points are generally anonymous and provided in no particular order. The points are organised in three subsections: General Points, Points related to how PSI should operate and Practical issues PSI should address, and then further categorised where appropriate for clarity.

4.1. General Points

Comments on PRISM

- The PRISM project presented a positive external message as it supported model diversity through a common infrastructure.
- The idea of the PRISM project was good and the result of PRISM was positive overall.
- Some of the PRISM tools would have benefited from improved software engineering practices.
- Perhaps the word PRISM could be dropped from the PSI to allow a “fresh start”.
- The PRISM project was too ambitious in its goals and made decisions too rapidly.

Tool Specific

- OASIS is extremely well regarded in the community.
- Tools should be small, well focussed on what they do and they should not require large “software stacks” to be installed to enable their operation.
- It is difficult to set up optimisation flags that are specific to a subroutine in SCE.
- OASIS4 should provide support for consistency between code.
- OASIS4 should support parallel models which are written using OpenMP as well as MPI.
- It should be possible to use “home grown” interpolations in OASIS4.

Standards and Compliance

- Once adopted, community standards (and shared tools) are beneficial and make it easier to do science. For example, the SCE allows model versions to be simply changed, and standard interfaces make the coupling of new models simpler.
- A standard set of tools is not necessarily what is required. Standardisation is generally a good thing where exchange (of models, or code, or data for example) occurs between groups. However, where exchange is not required, institutions often wish to have their own specialised tools (for build and run, data processing, visualisation etc.). Thus, tools in these areas need to be independent so as to allow groups to use whatever is appropriate for them.
- Not all groups will wish to use all tools and solutions should not be imposed on groups which do not really require them.
- The ESM effort is still relatively immature in terms of the “science of coupling” models together. Therefore, it is too early to consider standardisation for model interfaces or even model names, (although standardisation of model interfaces is an interesting research area).
- PRISM should not be a “thing to adapt to”. The PRISM project has been expensive in terms of its requirement to “adapt to PRISM”. Rather PRISM should attempt to adapt to models where possible.
- There should be no (or very few) mandatory requirements, which may conflict with existing in-house standards, from any side in a potential collaboration. For example, it should not be the case that all models MUST output using the same standard format. Conversions can always be applied afterwards.

4.2. Points relating to how PSI should operate

Structure

- The proposed PSI structure looks too complicated (technocratic).

Interaction with users/sites

- "Bottom-up" collaboration where shared interests/problems are identified. Promote collaboration to solve these problems and disseminate the activities and results to the rest of the community.
- The requirements of scientists in the community are key to identifying what needs to be done (PSI should be science driven).
- The PSI should actively engage with scientists and infrastructure developers in their own environment in order to identify requirements and disseminate good practice.
- Each institute will have a mechanism to manage their internal developments and the institute will have external links within the community. PSI should couple to these existing mechanisms.
- PSI team members should visit institutions regularly, for example to discuss requirements, determine requirements and expertise, disseminate good practice and undertake specific technology demonstrations, rather than there being many meetings to which scientists and developers in institutions have to travel.
- In PRISM, the OASIS coupler was a success (whereas not all tools were) because, early in PRISM, the requirements from users and developers were sought and got. This is a good technique and should be followed in PSI.

Coordination

- PSI should aim to co-ordinate and influence activities in institutions rather than to direct them.
- PSI should play a role in co-ordinating responses to funding calls, such as EU proposals.
- Where sharing between groups (i.e. exchange) is occurring, seek and promote good solutions, where they exist, and only build a new solution if absolutely necessary.
- PSI should encourage networks of specialists (e.g. in graphics, compilation etc.)
- PSI should seek to promote collaboration where shared interests exist and supplement effort in groups to make the results more widely available. PSI should aim to provide support in the longer term. This would enable local effort to be focussed primarily on local problems, as it should be. An example of such collaboration might be FCM.
- PSI could be a forum for discussion with a view to members of the community working in a co-ordinated way where possible.
- PSI could begin discussions with other projects/communities such as COSMOS.
- There needs to be international collaboration as well as European. In particular with the US and Japan (Earth simulator).

Effort

- Institutes will not allow their infrastructure staff to be controlled/managed by an outside body like PSI.

- Groups will only be willing to put a small amount of effort into adopting a new tool, though this is relative to the perceived pay-off to scientists. New tools should have minimum impact on the scientists.
- Support provided in an institution has to be focussed on the institutions priorities. So offering software (tools, models) to the community is ok, but only with NO guaranteed support. The provision and funding of support effort is a key issue for PSI.
- ECMWF are aware of the benefits of having their models used by others (but are concerned about the support issue). However, they are willing to have to do some work in-house to incorporate models from outside into their system.

Standards

- PSI should promote good practice in the field, rather than impose standard solutions.

Provision of software

- Tools should be easy to adopt, i.e. easy to install, configure to local requirements and to use. Good documentation (with hard copy option) is key. Tools should also be portable and proper checks over compilers, OS's, libraries etc. should be made.
- PSI should maintain tools if required. There is an issue as to whether tools should be labelled as PRISM tools or not. If they are then perhaps it should only be when a significant proportion of the community is using them.
- PSI should not just develop Euro tools or "PSI tools". It should rather promote best practice with current tools. PSI could act as a "kite mark" for good quality tools which it promotes.
- What level of support should be provided with tools (quality control, bug fixing, …)?
- A mechanism should be set up to allow any modifications to PSI tools to be fed back to the maintainers.

Priorities

- Solving the immediate requirements of the community should be the priority and these problems will be focussed on the next-step science.
- PSI should aim to tackle issues one at a time in priority order (and complete each in around a year), rather than attempt to solve all problems at the same time. Priority should be, at least partly, based on the common issues between institutions.
- The next set of IPCC runs are a key driver for a number of sites. The priorities for infrastructure development should support preparation for these runs as a priority. This sets a clear timescale.

Dissemination of information

- PSI should promote "user group-" and "special interest group"-based community activities (e.g. email lists, webpages, and occasional meetings).
- PSI should host a community meeting (two days?) at most once a year to discuss strategy for the next phase of infrastructure developments (with at most a two year lookahead). Special interest group meetings could be organised at the same meeting.
- There is a clear desire for there to be "user groups" (or email lists/wiki pages etc.) for each tool where users can share their experiences and problems. Further, each tool should have a dev-list and/or bug tracking software such as bugzilla.

Remit

- PSI appears to be doing things now. Members of PSI should be more proactive in discussing/explaining the proposed role to the community.
- PSI should have a role as a technology watch group (i.e. review tools as they are developed and recommend them to the community). PSI should therefore be open to new tools and ideas if they are considered useful and should also consider removing tools if there are not (no longer) beneficial.
- PSI should not engage in research.
- PSI should engage in technical rather than scientific support. Scientific models should not be provided by PSI, they should be provided by the institutions developing them.
- Should PSI address the NWP community explicitly or just focus on support for Climate studies?
    - For example, should PSI support/promote the use of GRIB?
    - Note that GRIB is in use at MPI-MET because (and only because – they would prefer to use netCDF) it produces significantly smaller output files. This matters because the cost of (long term) storage is becoming increasingly significant.
- A long term aim of PSI should be the interoperability of data. As such it would be a mistake to ignore the requirements of the NWP community.

4.3. Practical issues PSI could tackle

- Some people have commented that it is difficult to find out what tools are available at any time and from where they may be obtained. This suggests that the current set of stable releases of tools and download instructions etc. could be better advertised (on the PSI website?).
- A major issue for coupling is that I/O should be able to be performed from a model in a way that is independent of whether the model is operating in a coupled or forced mode.
- The management by PSI of a FAQ for each tool would be helpful. In particular for OASIS4 as it continues to develop. Some example questions such a FAQ might address are:

  1) The OASIS4 user guide (1st edition) contains some passages which are difficult to understand or terminology which is not clear, particularly for a user without in-depth knowledge of PRISM concepts.
  The pages at http://www.ccrl-nece.de/~redler/PRISM/WP3A/PSMILe/ are useful in clarifying some of the prism calls. e.g. documentation of the prism_get_nb_ranklists call is not particularly enlightening. Similarly, prism_set_scalefactor seems like it might be relevent to NEMO grids, but documentation is not complete and the purpose of the call isn't really explained. Examples of usage of each call would be helpful - i.e. not just the syntax, but an explanation of the context where a user might want to implement a particular call. The user guide says the neighbourhood search works in parallel "as much as possible". Which seems to imply some limitations - it would be good to know what these are.

  2) A truly conservative coupler will be needed. What are the likely timescales for that being available? Presumably this is an issue for other users? Rene Redler also mentions difficulties with handling the ORCA grid. It would be useful to know if these problems are likely to be addressed within PRISM or if this is something which will require special treatment by the component side. (We could, for instance, regrid within the NEMO

component from an ORCA grid to a standard lat-long grid, before sending data to the coupler, but that's not something which is likely to be desirable or efficient).

3) The recommendation to implement a separate call to PRISM_init seems redundant given that PRISM_init_comp has the capability of detecting whether prism_init has already been called and can then call it if necessary.

5. Thanks

A very large thank you is due to all the people who have committed time and effort to participate in this survey. A special thanks is due from the authors to the people who hosted them during the site visits which took place during the survey.