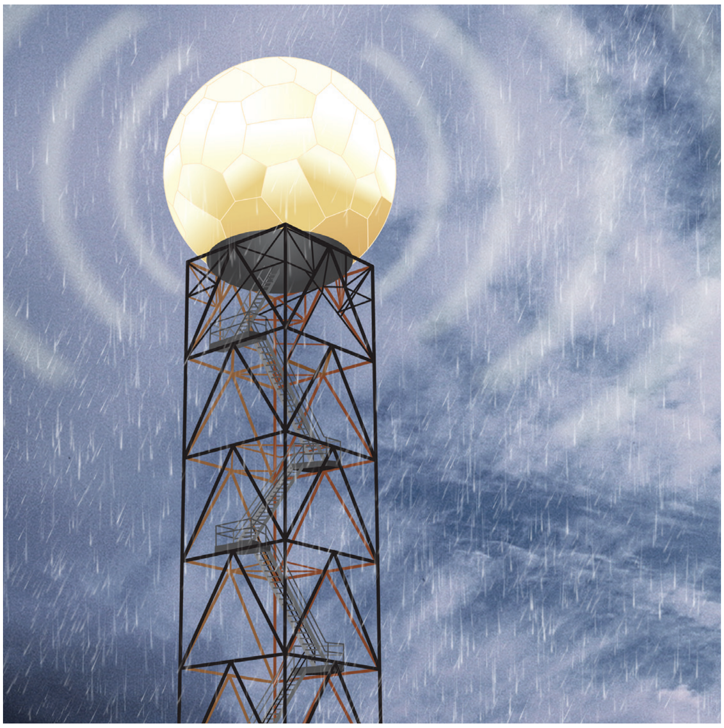
ECMWF Feature article

from Newsletter Number 117 – Autumn 2008

COMPUTING

The EU-funded BRIDGE project



www.ecmwf.int/en/about/news-centre/media-resources

doi:10.21957/t8axr71gg0

This article appeared in the Computing section of ECMWF Newsletter No. 117 – Autumn 2008, pp. 29-32.

The EU-funded BRIDGE project

Baudouin Raoult, Cristian Codorean

BRIDGE is a project funded by the European Commission under the sixth Framework Programme - Information Society Technologies (FP6-IST). The project started in January 2007 and was scheduled to last for two years. The aim of the project is to demonstrate the benefits of GRID technologies for international co-operation, in particular between Europe and China. The BRIDGE project covers three application areas: pharmaceuticals, aeronautics and meteorology.

The BRIDGE project provides the partners of the meteorological activity with an opportunity to address the development of the next phase of the THORPEX Interactive Grand Global Ensemble programme, otherwise known as TIGGE (see *ECMWF Newsletter No. 116*). In this next phase the data archives are distributed over a number of repositories, instead of all being held centrally (as in the first phase), but efficient and transparent access to users is maintained.

The aim of the meteorological activity in BRIDGE, which involves ECMWF, DWD and the China Meteorological Administration (CMA), is to build an infrastructure to provide distributed access to the TIGGE databases at ECMWF and CMA. This infrastructure will be used to explore ways of creating probabilistic forecast products in a distributed fashion. The main challenge will come from the sheer volume of data involved: methods will have to be found to use the data efficiently with minimal data transfers.

Design and concepts

The aim of the BRIDGE application is to implement distributed processing on distributed data while minimising data transfers. The distributed computational and archiving facilities may reside on different continents. It is expected that:

- Each site will host only part of the data.
- Each site will offer basic data manipulation services (e.g. computing an average).

During the analysis phase of the project, the concepts of *'products'* and *'operations'* were defined. A product (e.g. an ensemble mean) can be defined in terms of basic operations (e.g. data retrievals and averages).

Depending on their nature, some operations can be *'decomposed'* into elementary operations, which can be run independently.

A typical example of an operation that can be decomposed is the computation of an ensemble mean: assuming that ECMWF holds 50 members and CMA 30 members, we could either transfer the 30 fields from CMA to ECMWF and compute the mean of 80 fields there, or the mean could be computed by summing the 50 fields at ECMWF and summing the 30 fields at CMA, then adding the two partial sums and finally dividing the result by 80. In the first case, we would have to transfer 30 fields across the Internet, while in the second case we would have to transfer only one field.

As illustrated in the example above, it is assumed that intermediate results are usually much smaller than the original data. The adopted strategy to minimize data transfers is to:

- Decompose operations into a set of simpler ones.
- Deploy services that can perform these operations at each site, so that the operations are executed at the data location.
- Transfer the data only when necessary, otherwise exchange data references (URLs).

 doi:10.21957/t8axr71gg0

Using the Metview macro language

Most of the derived products generated twice daily at ECMWF from the Ensemble Prediction System, such as probabilities and distributions, are created using the Metview macro language.

Metview is an interactive meteorological application which enables operational and research meteorologists to access, manipulate and visualise meteorological data on UNIX workstations. It is used at ECMWF and many other meteorological centres to create derived products and plots by means of a powerful macro language.

The Metview Macro language provides researchers with an easy, powerful and comprehensive way to manipulate and display meteorological data. It extends the use of Metview into an operational environment as it enables a user to write complex scripts that may be run with any desired periodicity.

The language supports all the usual flow control statements: `if`, `while`, `repeat`, `for`, ..., it also supports numbers, strings, dates, lists and associative arrays as types. In addition it implements all the mathematical functions such as `sin()` or `exp()`, following the Fortran convention, so a researcher can easily port existing code into the macro language.

As the aim of the BRIDGE meteorological activity is to create such products in a distributed fashion, it seems natural to reuse an existing production system, i.e. express data access using the MARS query language and adapt the Metview macro language so it can invoke operations on the GRID instead of performing them locally. This will then allow us to easily port existing macros in the BRIDGE environment.

Implementation

In the context of the BRIDGE project, the Metview language has been extended to support functions which are invocations of GRID services and to support the notion of “remote data”. A variable can contain a reference to a piece of data that resides on another site.

The application makes use of the GRIA GRID middleware in Europe and the GOS GRID middleware in China. The interoperability between the two types of middleware is addressed by the project’s technology partners.

The main components of the applications are:

- ***script parser –*** parses the Metview scripts.
- ***grid executor –*** maps Metview function calls to GRID service invocations.
- ***cost estimator –*** estimates the cost of invoking a services.

The cost estimator is the central component of the application. Its role is to decide on which site an operation must be performed so that the amount of data transferred is minimal. To illustrate the algorithm, let us consider the following code snippet:

```
plot(retrieve(param:’tp’,step:48) + retrieve(param:’tp’,step:24))
```

The script parser will transform it into the expression tree given in Figure 1.

For illustration purposes, it is assumed that one of the retrievals was performed at ECMWF and returned a reference (URL) to a 50 Megabyte file. Similarly, the second retrieval was performed at CMA and returned a URL to a 25 Megabyte file.

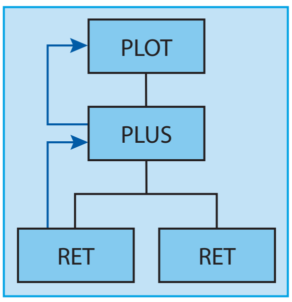


Figure 1 Expression tree.

doi:10.21957/t8axr71gg0

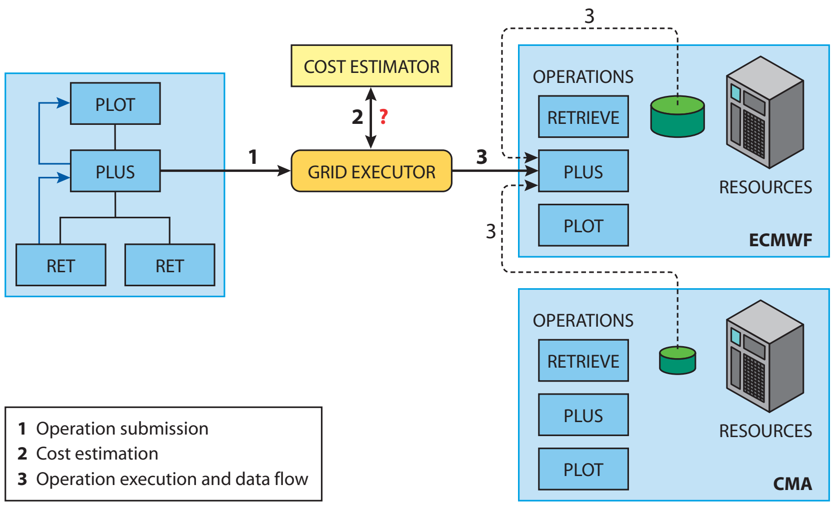


Figure 2 Example of an operation execution.

The next step is to perform the plus operation. Figure 2 shows how the sequence of calls is carried out.

- The plus(grib,grib) operation needs to be dispatched on the GRID. The operation needs to sum two pieces of data, one available at ECMWF and one at CMA (depicted in green in the figure).
- The plus(grib,grib) operation is available at both ECMWF and CMA, so we need a way of deciding where to execute the operation.
- The *cost estimator* is the component that will make the decision. The choice is based on selecting the execution scenario with the lowest cost. In the above case, the decision is to dispatch the *plus (grib,grib)* operation at ECMWF because moving the piece of data required by the operation from CMA to ECMWF costs less than moving the data from ECMWF to CMA as the data at CMA is smaller than that at ECMWF.

As we can see from this example, the more sites that offer the same operations, the more possible execution paths there are to choose from. For a medium to long script, the number of choices will grow exponentially with the number of operations performed and the number of services available on the GRID.

The exponential nature of the decision making prevents any realistic *a priori* evaluation of the cheapest execution path. On the other hand, the chosen solution, where the choices are made dynamically, is scalable and manageable.

Deployment

At this point in the project, the GRID infrastructure has been deployed between ECMWF and CMA, providing access to the TIGGE data held by each site. The GRID has been extended to include services hosted by DWD, with the objective of assessing any DWD-specific requirements and considering their impact on the application prototype, making the system more portable to other sites.

GRID services have been installed at ECMWF, DWD and CMA, using the GRIA and GOS GRID middleware. A Perl package has been written that isolates the implementation of a service from the specific middleware, thus allowing us to deploy the same operations at any of the sites. Currently, the operations that have been deployed provide:

- Access to the TIGGE data from MARS at ECMWF and CMA.
- The creation of plots using Magics^{++}. Some of the following plotting formats are supported: png, gif, jpeg, svg, ps, pdf, kml.
- Manipulation of GRIB data using the grib_api tools.
- All Metview operators and functions on fields (+, /, sin(), cos() sqrt(), stdev(), mean() etc.).
- Csobank: Access to the GME global model from the meteorological database (DWD).
- Climate Data Operators Software (CDO) from the Max-Planck-Institut für Meteorologie for calculations on GRIB fields (DWD).

Deploying the GRID infrastructure has been a challenge, particularly with regard to the security infrastructure of each of the sites.

doi:10.21957/t8axr71gg0

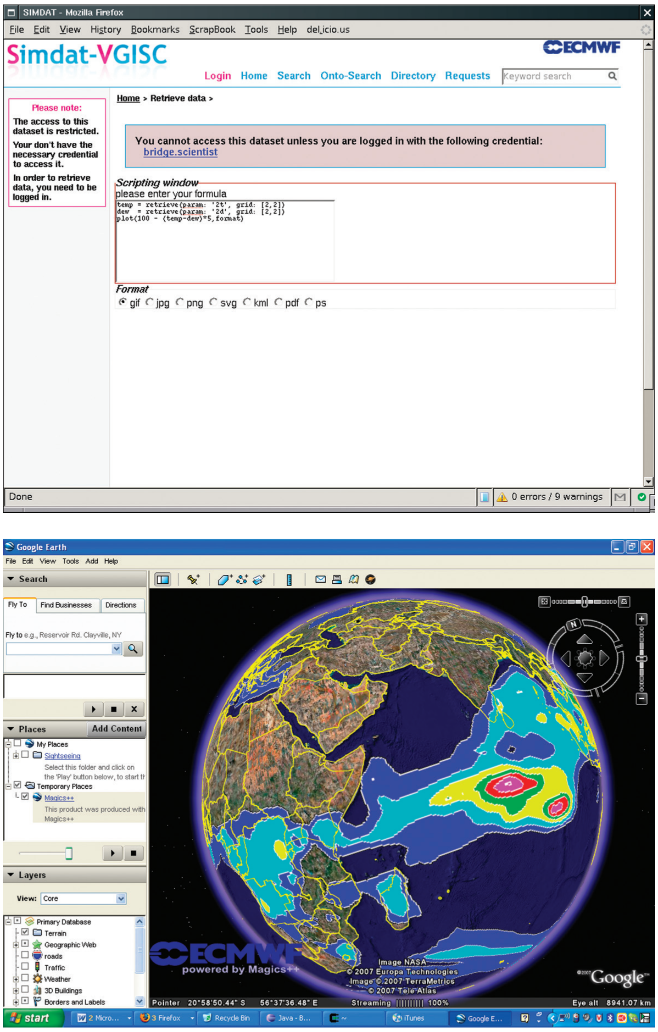


Figure 3 (a) Calculation of the ensemble mean precipitation by accessing BRIDGE from the SIMDAT portal and (b) the result viewed in Google Earth.

Integration with SIMDAT

The BRIDGE software has been integrated into the infrastructure of the SIMDAT project. SIMDAT is a four-year EU-funded project aimed at developing generic GRID technology for the solution of complex data-centric problems (see *ECMWF Newsletter No. 104*). A specific data repository has been developed that handles BRIDGE scripts. Consequently a user can now submit, monitor and retrieve the results of scripts submitted through a SIMDAT portal.

Figure 3 shows how to access the ensemble mean precipitation computed with BRIDGE from the SIMDAT portal and the result viewed in Google Earth.

Achievements and provision of further information

The work done as part of the BRIDGE project has demonstrated that the technology exists to perform distributed operations on distributed meteorological data. It has also highlighted some of the difficulties, such as network latency, security and the management of intermediate results.

Several deliverables have been produced that provide detailed information on the architecture and implementation of the BRIDGE software. The source code is available under the Apache 2.0 licence. For more information on the project, please contact the authors at *Baudouin.Raoult@ecmwf.int or Cristian.Codorean@ecmwf.int.*