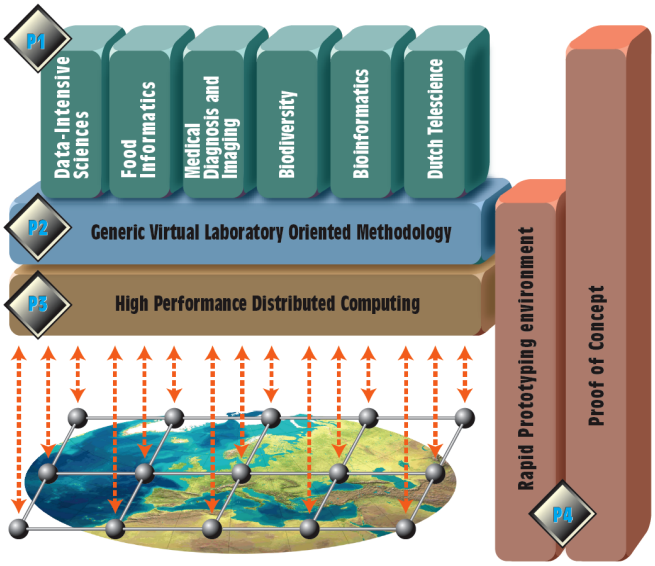
vl·e virtual laboratory for e·science

information

VL-e enables new approaches to traditional sciences

Information has become the fuel of our knowledge society, and our ability to digest, understand and share it will determine our scientific, economic and social progress.

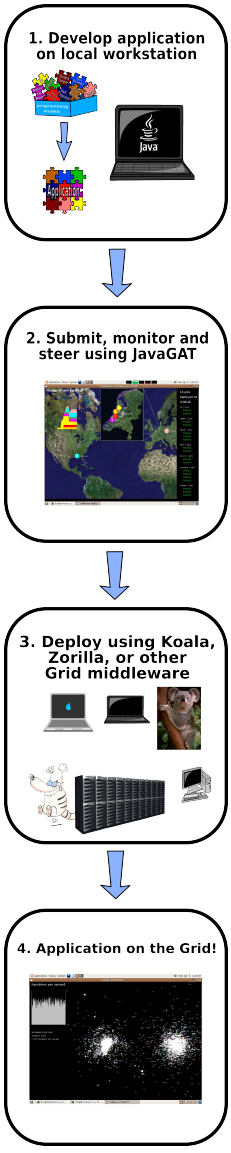
The exceptional increase in computing power, storage capacity and network bandwidth over the past decades forms the basis of a digital revolution which has only just started. Also the changing scale and scope of experimental sciences require a new research paradigm: (digitally) enhanced science or e-Science. The aim of the 'Virtual Laboratory for e-Science' (VL-e) project is to bridge the gap between the technology push of the high performance networking plus the Grid and the application pull of a wide range of scientific experimental application domains. A typical example of this is the life sciences, where VL-e offers solutions for combining laboratory research with computational experiments and simulations, making use of the knowledge and experience gained from dealing with large data sets in high energy physics. At the same time, however, it is recognised that data sets in the life sciences are far more complex than in high energy physics.



More specifically, VL-e is developing a Proof-of-Concept (PoC) infrastructure (both hard- and software) to enhance location-independent access to scientific information and stimulate global and multidisciplinary collaboration, thereby enabling new approaches to traditional sciences. The VL-e software (both for rapid prototyping and in the PoC) provides generic functionalities that support a wide range of e-Science applications. This PoC infrastructure will boost the knowledge economy of the Netherlands.

Currently, six application domains are involved: Data-Intensive Sciences, Food Informatics, Medical Diagnosis & Imaging, Biodiversity, Bioinformatics and Telescience. Several Dutch universities, academic hospitals and industries in the life sciences and ICT domain participate in this project. There is also strong collaboration with NBIC (Netherlands Bioinformatics Centre). The recently started Dutch BIG GRID project will build a nationwide production grid, making use of methodology still being developed within VL-e.

Developing, Scheduling, Deploying, and Managing High Performance Distributed Grid Applications



Researchers who try to use grid systems are currently faced with a multitude of problems. We develop software and techniques for creating and using grid applications quickly, efficiently, and reliably.

When developing an application for a grid environment, it is important that the programming model used is suitable for the application while hiding as many of the complexities of a grid as possible. We offer developers several programming models.

Satin, our simplest and most powerful programming model hides the grid completely from the programmer. A Satin application is written as a sequential divide-and- conquer style Java application. Satin then automatically distributes this application across a grid. We also offer MPJ (the Java language bindings for MPI) and RMI (the standard Java RPC like model) as alternative programming models.

All these programming models make use of Ibis, our specialized grid communication software. Ibis is capable of handling machines leaving and joining a running computation and is tolerant against failures. Through our Smartsockets library, applications can handle difficulties encountered while communicating in a grid environment, such as NATs, multi homed machines, and Firewalls.

Running a grid application is as hard as developing it. Numerous middleware systems are in use on grids, some with limited capabilities.

To be able to deploy and manage grid applications in a simple, reliable and middleware independent manner, we developed the JavaGAT. The JavaGAT is capable of running any application, Java or otherwise, on all common grid platforms in use today. This includes the DAS-3, the system we use to develop our software. If no grid is present, Zorilla, our Peer-to-Peer middleware, can be deployed to form an ad-hoc grid.

When an application is deployed on a grid, the resources available at a single site may not suffice. Our KOALA grid scheduler addresses this problem by co-allocating resources from multiple sites for use by a single application. KOALA runs alongside the local schedulers of the grid sites, supports applications developed with different programming models, and provides load balancing across the sites of a grid when co-allocation is not used.

vl-e facts

budget 40 M, period 2004-2008
more than 20 consortium partners from industry and academia
director: prof. dr. L.O. Hertzberger
website: http://www.vl-e.nl

consortiumpartners

A&F Wageningen, AMC, CWI, DSM, Friesland Foods, FEI, FOM AMOLF, NBIC, Nikhef, IBM, LogicaCMG, Philips Research, Philips Medical, SARA, Top Institute Food and Nutrition, TNO Kwaliteit van Leven, TU Delft, Unilever, UvA-IBED, UvA-IvI, UvA-SILS, VU, VUmc, WTCW

VL-e Program line: Large Scale Distributed Systems

Subprogram: High Performance Distributed Computing

Leader: Prof. dr. Henri E. Bal
bal@cs.vu.nl

VU University Amsterdam
http://www.cs.vu.nl/ibis

Delft University of Technology
http://www.st.ewi.tudelft.nl/koala

ibis

The Koala Grid Scheduler