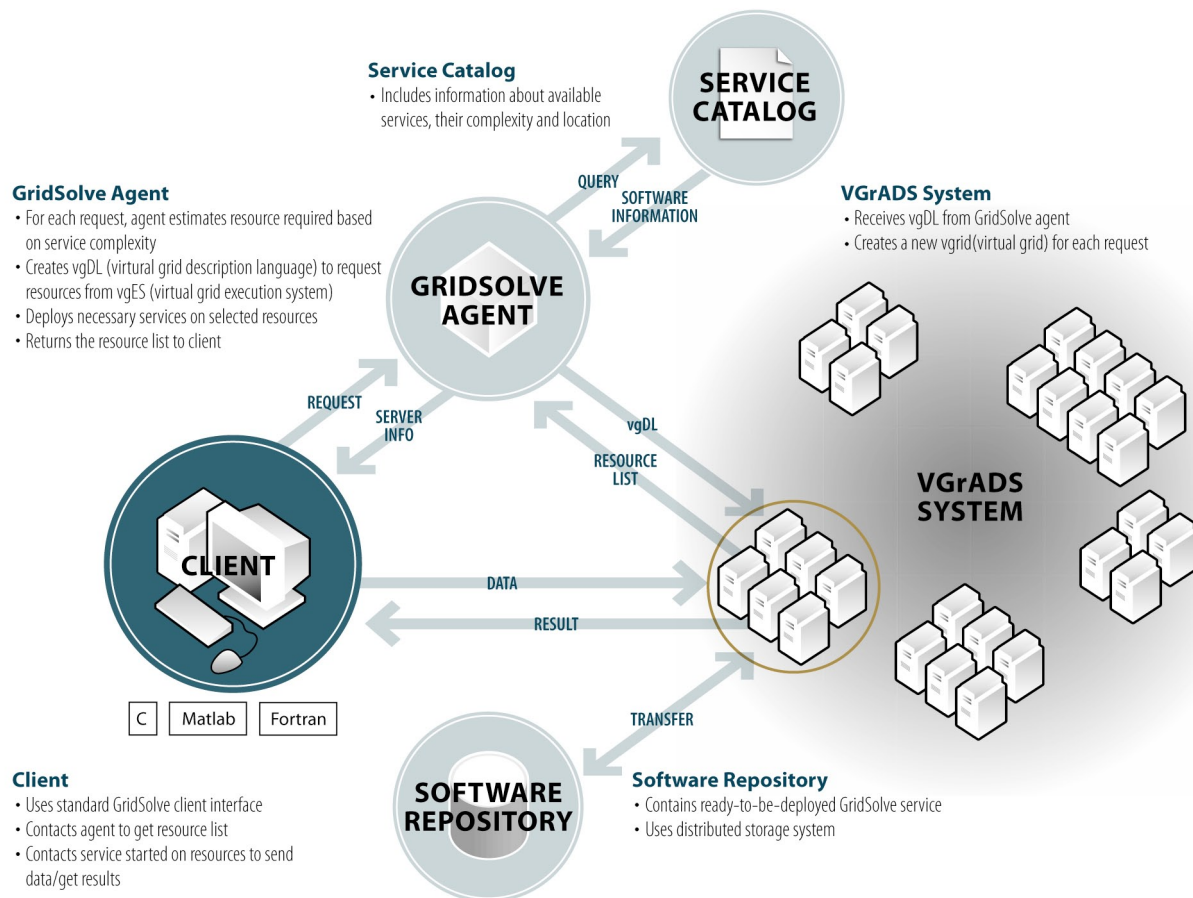


VGrADS and GridSolve

The overall goal of integrating GridSolve with VGrADS is to provide end user multiple interfaces to the grid. GridSolve has implemented several interfaces such as Matlab, Mathematica, Octave, Fortran, and C. By seamlessly integrating GridSolve agent with VGrADS execution system, users are allowed to access and utilize Grid resource with interface of his choice. The integrated system can also be used to extend the capabilities of problem solving environments, such as Matlab, by increasing the number and types of implemented algorithms available and solving them on Grid resources.



VGrADS

The complexity, unreliability, and overhead of low-level operations in today's systems obscure the Grid's potential. Virtual Grid Application Development Software (VGrADS) project aims to attack a fundamental part of this problem - how to more effectively program these highly complex and dynamic systems. To address this, VGrADS will adopt the concept of virtual grid (vgrid) architecture as a fundamental organizing principle. In this architecture, VGrADS will abstract the Grid into:

- Physical resources (e.g. data archives, computing systems, and instruments)
- Abstract resource classes with specified attributes, formed by aggregating and conditioning physical resource
- Vgrids, composed of components from abstract classes
- A set of abstract programming models and development tools that target vgrids as application and service-visible execution interfaces.

Vgrids cleanly separate high-level programming tools, applications, and services from the complexity of dynamic Grid scheduling and resource management.

GridSolve

GridSolve is a project that investigates the usage of distributed computational resources connected by computer networks to solve complex scientific problems efficiently. It is a remote procedure call (RPC)-based client/agent/server system that allows users to discover, access, and utilize remotely housed software modules, as well as the computational hardware needed to run these modules. The resources to be leveraged can be distributed by geographic location and/or ownership, and heterogeneous operating environments are supported. The motivation for GridSolve is to create a grid-based software computing environment used routinely by a large user base to enhance scientific computing capabilities. Fundamental characteristics include:

- Ease-of-use for both the user and administrator
- Efficient utilization of resources
- Ease-of-integration of new software modules
- High levels of quality assurance (in the accuracy and performance of both the GridSolve system and the underlying software services)



<http://vgrads.rice.edu/>



Sponsored by



VGrADS and GridSolve

The complexity, unreliability, and overhead of low-level operations in today's systems obscure the Grid's potential. Virtual Grid Application Development Software (VGrADS) project aims to attack a fundamental part of this problem – how to more effectively program these highly complex and dynamic systems. To address this, VGrADS will adopt the concept of virtual grid (vgrid) architecture as a fundamental organizing principle. In this architecture, VGrADS will abstract the Grid into:

- Physical resources (e.g. data archives, computing systems, and instruments)
 - Abstract resource classes with specified attributes, formed by aggregating and conditioning physical resource
 - Vgrids, composed of components from abstract classes
 - A set of abstract programming models and development tools that target vgrids as application and service-visible execution interfaces.

Vgrids cleanly separate high-level programming tools, applications, and services from the complexity of dynamic Grid scheduling and resource management.

GridSolve is a project that investigates the usage of distributed computational resources connected by computer networks to solve complex scientific problems efficiently. It is a remote procedure call (RPC)-based client/agent/server system that allows users to discover, access, and utilize remotely housed software modules, as well as the computational hardware needed to run these modules. The resources to be leveraged can be distributed by geographic location and/or ownership, and heterogeneous operating environments are supported. The motivation for GridSolve is to create a grid-based software computing environment used routinely by a large user base to enhance scientific computing capabilities. Fundamental characteristics include:

- Ease-of-use for both the user and administrator
- Efficient utilization of resources
- Ease-of-integration of new software modules
- High levels of quality assurance (in the accuracy and performance of both the GridSolve system and the underlying software services)

The overall goal of integrating GridSolve with VGrADS is to provide end user multiple interfaces to the vgrids. GridSolve has implemented several interfaces such as Fortran, C, Matlab, Mathematica, and Octave. By seamlessly integrating GridSolve agent with VGrADS execution system, user is allowed to access and utilize Grid resource with interface of his choice. The integrated system can also be used to extend the capabilities of problem solving environments (PSE), such as Matlab, by increasing the number and types of implemented algorithms available and solving them on Grid resources.

We have adapted a fault tolerant version of parallel conjugate gradient equation solver (PCG) to use in our framework. The client first requests resource within a cluster from vgES according to the problem size and complexity. After the service has been deployed on the selected resource, GridSolve servers start and register to the agent. A process is dedicated in the PCG application to serve as an in-memory checkpoint server. Every fixed number of iterations, all processes calculate checkpoint of each relevant vector, which is then stored on the dedicated checkpoint process. In case one of the resources fails, the fault tolerant FT-MPI replaces the failed resource, rebuilds a communicator and notifies the PCG application. The application can then proceed using the stored checkpoint without requiring a restart.



<http://vgrads.rice.edu/>



Sponsored by

