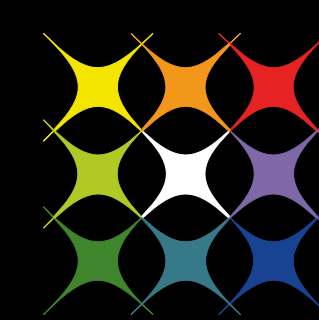# Ibis: Java-Centric Grid Computing

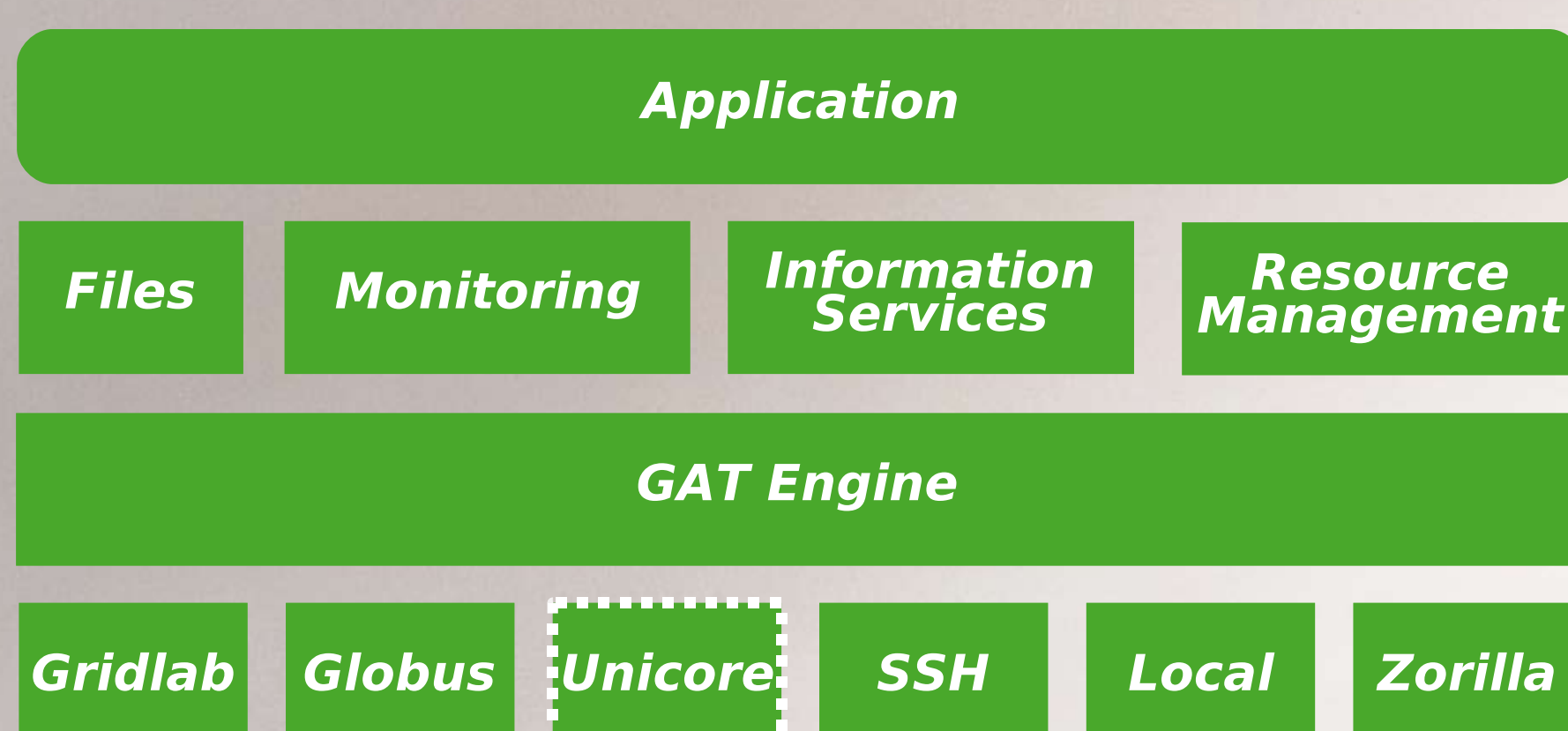vl·e virtual laboratory for e·science

Introduction

The main goal of the Ibis project is to create an efficient Java-based platform for grid computing. The Ibis project currently consists of the **Ibis communication library**, a variety of **programming models**, the **Java Grid Application Toolkit**, and the **Zorilla** peer-to-peer grid middleware. All components can be deployed on any grid platform, due to the use of Java. Each of the components will be described in detail below. On the right there is a selection of the projects currently using Ibis software.

Java Grid Application Toolkit

The Java Grid Application Toolkit (JavaGAT) offers a set of coordinated, generic and flexible APIs for accessing grid services from applications, portals, and data managements systems.

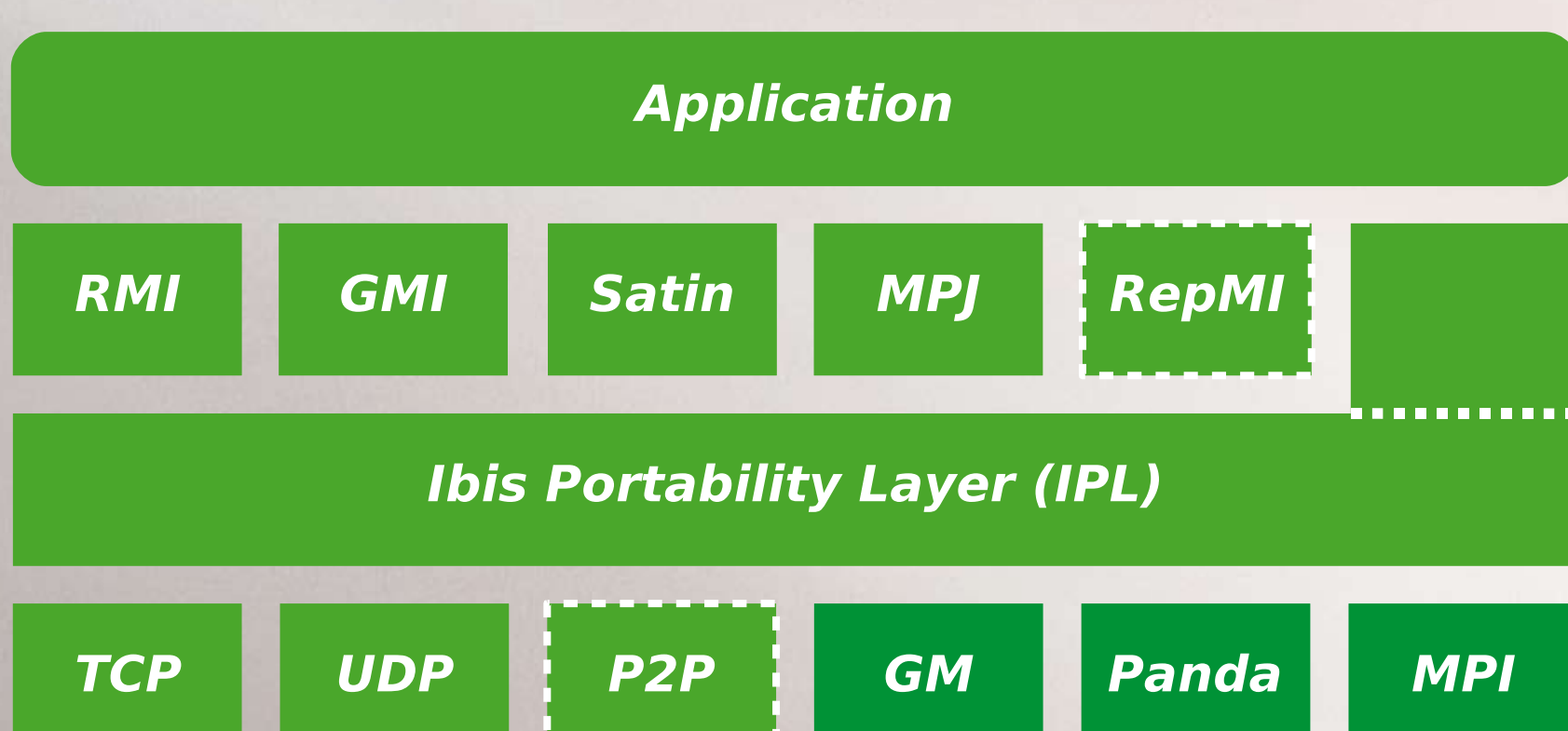| Application			
Files	Monitoring	Information Services	Resource Management
GAT Engine			
Gridlab	Globus	Unicore	SSH	Local	Zorilla

As shown in the above figure, the JavaGAT sits between grid applications and numerous types of grid middleware, such as Globus, Unicore, SSH or Zorilla. JavaGAT lifts the burden of grid application programmers by providing them with a uniform interface that provides file access, job submission, monitoring, and access to information services. As a result, grid application programmers need only learn a single API to obtain access to the entire grid. Due to its modular design, the JavaGAT can easily be extended with support for other grid middleware layers.

Zorilla

Zorilla is peer-to-peer grid middleware. It implements all functionality needed to run applications on a grid in a fully distributed manner, such as scheduling, file transfer and security. Deployment of Zorilla is easy since machines automatically organize themselves into a grid. Due to its peer-to-peer design, Zorilla scales to large numbers of machines. Jobs can be submitted to Zorilla directly or using the JavaGAT. Although Zorilla was explicitly designed to support supercomputing applications, it is possible to run any type of application.

Ibis Communication Library

The Ibis communication library is specifically designed for usage in a grid environment. Its run-everywhere property and support for fault-tolerance, malleability and high-speed networks, make it an easy to use and reliable grid communication infrastructure. This modular design of Ibis is shown below:

Application				
RMI	GMI	Satin	MPJ	RepMI
Ibis Portability Layer (IPL)				
TCP	UDP	P2P	GM	Panda	MPI

Ibis Programming Models

Ibis offers a wide range of programming models, suitable for different classes of applications: RMI is best suited for client-server applications, Satin offers a master-worker/divide and conquer programming model, GMI implements object-oriented group communication, while MPJ offers pure Java MPI.

MultimediaN

The MultimediaN institute performs various types of multimedia research, such as feature learning, object recognition and multimedia databases. One of their demonstrations, which is used to illustrate the effectiveness of using a grid in Multimedia computing, shows object recognition performed by a robot dog. The dog is connected to a wide-area Grid system, which potentially consists of hundreds of computers distributed world-wide. The JavaGAT is used to start computations on the grid.

ProActive

ProActive is a Java grid library for parallel, distributed, and concurrent computing, featuring mobility and security in a uniform framework. With a set of simple primitives, ProActive provides a comprehensive API allowing to simplify the programming of applications that are distributed on local networks, clusters of workstations or grids. ProActive uses Ibis for high-performance communication.

MEG Data Analysis

The Vrije Universiteit Medical Center has a magneto-encephalography (MEG) scanner which produces large amounts of data which must be processed. MEG is a tool to study the function of the human brain. The size of a data set from each session is typically hundreds of megabytes. The medical center uses a task framing application written on top of the JavaGAT to process MEG data, performing important research on multiple sclerosis and phantom pain.

Parallel simulation of soil-structure-interaction

The Institute of Numerical Methods and Informatics in Civil Engineering is implementing a Grid-System for the parallel simulation of soil-structure-interaction for geotechnical problems. Parallelization is based on mobile Java-Agents, taking advantage of agent features such as portability, mobility, reactivity etc. In the time critical parts of the parallel simulation, mainly solving a linear equation system, Ibis is used for a more efficient communication.

Triana

Triana is a workflow-based problem solving environment developed at Cardiff University that combines a simple and intuitive visual interface with powerful data analysis tools. Triana is used by scientists for a range of tasks, such as signal, text and image processing. Triana is capable of using the JavaGAT to run the task graph on the grid.

AstroGrid

JavaGAT is used intensively within in the AstroGrid community. The Max-Planck-Institute for Astrophysics (MPA) in Garching uses JavaGAT to grid-enable the work-flow engine ProC. This work-flow engine will also be used within in the international Planck project to analyse data sent back by the Planck spacecraft, studying relic radiation from the Big Bang.

Holland.

ibis

GridLab

Vrije Universiteit Amsterdam

R. van Nieuwpoort, J. Maassen, C. Jacobs
Vrije Universiteit, Amsterdam, The Netherlands