

Abstract

The falling costs and steady increases in processing capacity and data storage of personal computer, as well as throughput improvements on local networks have stimulated the idea of using such non-specialized computers to execute high performance parallel and distributed applications. Furthermore, the Internet has seen dramatic increases in available bandwidth and reductions in latency thanks to fibre optics which makes distance no longer an obstacle to harnessing the power of significant numbers of geographically dispersed computers. The field of Grid Computing distinguishes itself from distributed computing by its focus on large scales collaboration. Through this collaboration, scientist can obtain access to, for example, computing power, databases and instruments not available locally and carry out research which was not possible previously.

Being both shared and composed of heterogeneous resources, computational grids are complex environments to use effectively. Numerous softwares, known as *service middleware*, have and are being developed to make these difficulties transparent to the user. Unfortunately, both the variety of middleware available and their often restricted application (specialization) means that scientists and even computing specialists require an indepth understanding of the workings of, and how to operate, each chosen middleware in order to fully take advantage of the functionality they offer.

One solution that might significantly facilitate the use of computational grids is the adoption of a common standardized interface for access and operation. Portals are generally Web-based and thus only require that the user has a basic knowledge of how to access WWW pages. Furthermore, these pages can be presented in the form of a graphical interface to facilitate interactivity with the user and hide the specific details of operating computational grids. Features offered by portals make them powerful tools to further the use of grids in scientific communities.

This work contributes to facilitate the use of computational grids through the implementation of a portal, based on the recently proposed *portlet* standard. The design approach adopted permits the interoperability between different middlewares and multiple computational grids simultaneous and was developed to allow functionalities to be incorporated independently of the programming language in which they were written. Additionally, the portal supports personalization to allow users to chose the degree of specialization to which they would like to be exposed to.

Keywords: grid portal, gridsphere, portlets, JSR168, easygrid