Towards a Monitoring Framework for Worldwide Grid Information Services

Serafeim Zanikolas and Rizos Sakellariou

Department of Computer Science, The University of Manchester, Manchester, UK.
{zanikolas,rizos}@cs.man.ac.uk

Abstract. Despite important advances in Grid computing, scalability and interoperability issues obstruct the integration of the existing isolated grids into *the* Grid. We propose a worldwide monitoring framework which forms the basis for building worldwide grid information services that can have a global notion of the Grid. This work motivates the monitoring framework in terms of applications and challenges, sets its architecture and design space, and offers indicative performance results in a variety of hypothetical Grid instances.

1 Introduction

Grid computing is emerging as a new paradigm for on-line resource sharing and collaboration; numerous yet small and isolated grids exist as of early 2004. The existing grids, however, are missing the properties that are required for scaling and integration into *the* Grid (i.e., the capitalisation denotes a worldwide Grid where millions of nodes can collaborate to form "virtual organisations" as needed), in the same way that the Internet bridges together dispersed networks around the globe. The technical issues of this integration include the restructuring or extension of basic grid services in order (i) to satisfy the *scalability* and *robustness* required for operation into the Grid, and (ii) to support *interoperability* between grids with potentially different middleware implementations. Among the basic grid services, information services (e.g., [3]) include the collection of data about resources (monitoring) and the use of those data to support resource discovery and enquiry. However, existing information services are not interoperable, and suffer in terms of scalability because they are intended for small to medium size deployments. In this respect, the execution of a large scope resource discovery query (e.g., search throughout the Grid for the cheapest resource with given characteristics) would need a significant amount of time.

This paper proposes a worldwide monitoring framework that operates on top and hides the diversity of existing monitoring and information services, and thus supports the visibility and eventually utilisation of resources across grids. The suggested framework can cope with the scalability requirement of a Grid because resource information is collected in advance of queries, as opposed to on demand. Consequently, the overhead is proportional to the total number of grid resources, in contrast to the total number of query requests, which is the case for existing information services. The framework provides a uniform and general-purpose interface for worldwide grid information services, and allows analysis of the Grid in a similar way Netcraft and Alexa (`netcraft.com`, `alexa.com`) analyse the World Wide Web. Besides motivating a worldwide monitoring framework and setting the design issues that need to be addressed for a scalable, flexible, and robust realisation, an additional contribution of the paper is a feasibility experiment with simulation results that indicate the performance of a minimal version of the proposed framework in a variety of hypothetical Grid instances. This paper considers architectural and performance issues of the framework but does not contribute towards interoperability.

The next section outlines potential applications of the suggested monitoring framework and identifies the involved challenges. Section 3 proposes a general architecture and highlights the relevant design space. Section 4 presents simulation results to illustrate the framework's scalability and Section 5 concludes.

2 Applications, Challenges & Related Work

The proposed monitoring framework can be used two-fold: (i) to build efficient, Grid-wide resource discovery end-user applications or services, such as search engines or directories of grid resources, or (ii) to provide a variety of "global view" services to enhance the understanding of the Grid.

Grid search engines, as envisaged in [4], may provide an efficient means of resolving complex, large-scope queries (i.e., involving the combination of information from many organisations.) However, end-users may not be able to specify exactly their needs or be unaware of the resources that are available throughout the Grid. In this respect, *resource directories* could support end-users to find the appropriate matches through a browsable list of resources, offered in a variety of classifications. The latter can be based on criteria such as discipline (in case of collaborations, data repositories, and on-line instruments), sharing policy, physical location (e.g., continent, country, city), administration hierarchy (e.g., virtual organisation, organisation, department), supported quality of service guarantees, and hardware and software specifications.

On the other hand, *global view services* will provide a comprehensive quantitative and qualitative characterisation of the Grid throughout time. This will help to understand the dynamics of the Grid, support long-term capacity planning and contribute to its realistic modelling and simulation by answering questions such as: how many are the grid resources in the public (commercial) Grid; what is their distribution among sites; which resource types are rare or popular; which grid sites are most reliable; how these numbers evolve throughout time; and how does the Grid's evolution relates to that of the Internet.

Challenges A realisation of the proposed monitoring framework needs to address a set of contradicting requirements; namely scalability, extensibility, adaptivity, flexibility, completeness and freshness of the collected information. The system must be *highly scalable* with respect to the number of resources available throughout the Grid, and to the imposed network intrusiveness. The latter is a crucial concern given the high utilisation costs of wide area network links. The design must also allow for *extensibility* to readily accommodate the new types of resources that are introduced for sharing through the Grid. This concern has motivated the grid and web communities to adopt XML and RDF based representations (`www.w3.org/{XML,RDF}`); their applicability on a worldwide monitoring framework, however, should be investigated with respect to the stringent scalability requirement. The monitoring framework also needs to be *adaptive* to unforeseen conditions, such as a rapid growth of the number of grid resources or an Internet slowdown caused by a worm. In addition, it must be *flexible* with respect to (i) the overhead that resource providers are willing to accept; and (ii) the supported query language for resource-related information retrieval.

Moreover, the framework has to support a high degree of coverage of the Grid. Although this *completeness* may not always be needed, it is what differentiates the proposed framework from the existing, scope-limited grid information services. The completeness requirement is an engineering challenge (e.g., to achieve the required scalability and accommodate the diversity in monitoring and information systems) as well as a political one (e.g., why should resource providers expose detailed information concerning their resources.) Last, the *freshness* of the collected information refers to its correctness with respect to the dynamic nature of resources. Obviously optimal freshness cannot be achieved given the scale and dynamism of the Grid. Highly dynamic features of resources (e.g., CPU load) could be considered as statistically annotated aggregates.

Related Work Many research activities relate to the integration of grids, including middleware interoperation [2] and semantically rich (e.g., domain dependent) resource matching [5]. In the context of large-scale information services, however, the only work we are aware of is that of Balaton *et al* [1]. In contrast to our work which extends the web crawler paradigm, their proposal resembles the architecture of the USENET Internet service. In particular, their system appears to have a more decentralised structure; it would be interesting, however, to see how a USENET-like system could support user-level flexibility of data collection, delivery and scheduling policies (discussed in the next section.)

3 Overall Architecture & Design Space

This section describes the considered framework's architecture and the relevant design space that needs to be explored. The presented architecture is a generalised extension of the one described in [4]. For the needs of the following discussion, resource provider and resource information collector sites are referred as *grid* and *collector sites* respectively. Grid sites host resources that are exposed through the monitoring framework; collector sites systematically collect information about available resources. The exact granularity of grid sites (e.g., departments or organisations) is an issue that needs to be investigated. Generally, higher granularity suggests more potential for aggregating resource information, but also more network links affected by the traffic generated by the monitoring process. The proposed architecture consists of proxies, crawlers, and collectors (Fig. 1). *Proxies* are located on grid sites for continuously collecting data about local resources (i.e., configuration and status properties), either from the actual resources or from the local information services. Also, proxies aggregate and normalise these data according to a universal and compact data representation. On the collection side, multi-threaded *crawlers* traverse the network of previously discovered proxies for retrieving monitoring information, which is later used in application-specific ways. Additionally, proxies actively push a minor subset of the monitoring information to multi-threaded *collectors*, located at the collector site, to achieve timely delivery of previously designated important events (e.g., that an important resource has gone through a hardware upgrade.) Note also that proxies, in addition to the important functionality already described, are necessary to achieve the decoupling, in time and space, between resources on the one hand, and crawlers and collectors on the other.[1]

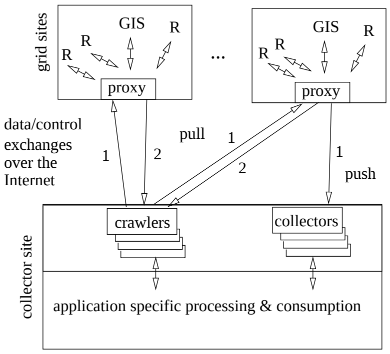

Fig. 1. Overall Architecture of the proposed monitoring framework (R:Resources, GIS:Grid Information Services.)

Design Space The described architecture sets the components and their interactions, but still many design variations need to be evaluated. These relate to four core themes: resource representation and addressing; data collection and reduction; data delivery and scheduling policies; and proxy membership management.

An extensible, expressive, yet highly compact *resource representation* is needed to capture, in addition to standard resource properties, the multi-dimensional relationships between resources (e.g., physical, logical, administrative.) Additional information regarding usage policies and disciplines needs to be supported. Another issue to be tackled is the provision of a universal *resource addressing scheme* that, among others, features high performance filtering; property-, structure-based schemes (e.g., feature vectors and uniform resource identifiers (URIs) respectively) and combinations of them have to be considered.

Exchanges between proxies and crawlers/collectors need to be relatively rare (given the sheer number of proxies.) To this end, crawlers have to set, and communicate to proxies, a *data collection policy* concerning the type, level of detail, and temporal granularity (e.g., fifteen-minute average values of, say, processor load) of the desired information. Proxies systematically collect and keep this information for the period between consecutive crawler "visits." The collection activity may vary significantly depending on the capabilities of local information

[1] Time decoupling means that two parties need not be on-line at the same time in order for them to communicate; space decoupling means that the parties need not know the other end's address/location.

services (e.g., lack of subscriptions requires polling.) In addition, *data reduction* techniques need to be evaluated for minimising the size of the transmitted information, and therefore the imposed network overhead.

A *data delivery policy* determines whether proxies passively provide (pulled from crawlers) or actively send (pushed to collectors) a given type of information. This should be decided on the basis of a given information's lifetime and significance, as well as the importance of the associated resource. Once such a policy is in place, a *scheduling strategy* is needed to determine the frequency and ordering of pull and push exchanges, in order to achieve optimum freshness of the delivered information, taking into account the resource limitations of the collector site (e.g., an inappropriate scheduling of pushes could easily overwhelm a collector site's network or processing capacity.)

An equally important design axis is the effective *management of proxy membership*. That is, the procedure that determines how new proxies join the monitoring framework (e.g., register with a collector site or passively discovered.) Also, collector sites have to set and identify the conditions in which a proxy's membership should be maintained or terminated (e.g., during a temporary network partition or a permanent shutdown of a grid site.)

4 Simulation-Based Preliminary Results

A simulation-based evaluation (based on SimPy, `simpy.sourceforge.net`) of a minimal version of the framework was carried out to illustrate its scalability and estimate its performance in terms of the time needed for a (theoretically) complete Grid crawl. Although performance needs may vary among different applications of the framework, a Grid crawl per 24 hrs is assumed for resource discovery purposes. (Note that the framework *is* suitable for more frequently changing information through the use of push updates, which however are not considered in this simulation.) At this point, data reduction, failures, access control, and proxy management issues are not considered (i.e., proxies are assumed to be known to the collector site.) Proxies are collecting information about the latest configuration of resources and their status since the previous crawler visit; crawlers visit proxies sequentially and retrieve all information for all resources (i.e., no filtering is applied.) No collectors exist since proxies do not push any information.

A collector site has a 16 Mbps full-duplex Internet connection, and adequate crawler hosts to saturate the connection. Connections between grid sites and the collector site are assumed to have an average end-to-end bandwidth of 256 Kbps. The end-to-end latency is sampled from a normal distribution, with parameters $\mu = 0.2, \sigma = 0.1$ (seconds.) Given the network settings and an average resource description size, the total transmission time of crawler and proxy messages is calculated by taking into account packet size, end-to-end bandwidth and latency.

The simulation has been run against three Grid instances (see Tab. 1), each one consisting of 10^6 grid sites, with a different static distribution of resources among sites (10 resources each, uniform in the range [1,100] and upper-bounded

Zipf(a=1.5), respectively.) The number of average crawls per second and the needed time for a complete Grid crawl may be misleading because the three Grid instances do not have the same total number of grid resources and the same number of resources per grid site. A more useful metric, the average crawled resources per sec (see Tab. 1), suggests that the performance is dependent on the average number of resources per site; this is because the cost of network latency dominates the cost of network connections to proxies with only a few associated resources. This further advocates higher granularity of grid sites (e.g., organisation vs department.) The distribution of resources among sites does not seem to affect performance significantly except in the Zipf instance probably due to proxies with numerous associated resources being visited at the end of the Grid crawl (leaving the collector site's bandwidth under-utilised since every connection can have only up to 256 Kbps.) Last, an estimation of the achievable number of crawls of a Grid consisting of 10^7 resources within 24 hrs, ranging from more than 2 or 3 crawls per 24 hrs, is quite encouraging with respect to scalability, especially considering that this is only a minimal version of the framework with no elaborate optimisations (such as not re-retrieving valid data.)

Table 1. Configuration & results of a simulation of the framework in three different Grid instances.

	balanced	uniform	Zipf
total resources	$1e7$	$5e7$	$6.315e6$
avg resources/site	10	50.0089	6.3831
sites with one resource	0	10173	607894
avg crawled resources/sec	362.5294	399.0035	312.1786
Grid crawls per 24 hrs assuming 10^7 resources	3.1323	3.4474	2.6972

5 Summary

This paper introduced and motivated a monitoring framework that forms the basis of (to be developed) worldwide information services, which are important for the bridging of grids into the Grid. The paper proposed a general-purpose architecture, identified the relevant design space, and presented preliminary performance results to illustrate the framework's scalability in a variety of hypothetical Grid instances. Further exploration of the design space is needed to fully define a working framework and improve overall performance.

References

1. Balaton, Z., Gombás, G., Németh, Z.: Information System Architecture for Brokering in Large Scale Grids. In: Proc. of DAPSYS 2002, Linz, Kluwer (2002) 57–65
2. Brooke, J., Fellows, D., Garwood, K., Goble, C.: Semantic Matching of Grid Resource Descriptions. In: Proc. of AxGrids04, (2004)
3. Czajkowski, K., Fitzgerald, S., Foster, I., Kesselman, C.: Grid Information Services for Distributed Resource Sharing. In: Proc. of HPDC-10 (2001)
4. Dikaiakos, M., Ioannidis, Y., Sakellariou, R.: Search Engines for the Grid: A Research Agenda. In: Proc. of AxGrids03, Santiago de Compostela, Spain (2003)
5. Miles, S., Papay, J., Decker, K., Moreau, L.: Towards a Protocol for the Attachment of Semantic Descriptions to Grid Services. In: Proc. of AxGrids04, (2004)