Network Control in Distributed Computing for Scientific Applications

[1]Mihai Lucian Cristea, [1,2]Rudolf Strijkers, [1]Vladimir Korkhov, [1]Adam Belloum, [3]Mark Kettenis, [3]Aard Keimpema, [4]Damien Marchal, [1]Paola Grosso, [1]Cees de Laat, [1,2]Robert Meijer

[1]University of Amsterdam, Science Park 107, 1098 XG, Amsterdam, The Netherlands
{m.l.cristea, v.korkhov, a.s.z.belloum, p.grosso , delaat}@uva.nl; cushingreggie@gmail.com

[2]TNO, Informatie- en Communicatietechnologie, Groningen, The Netherlands; {rudolf.strijkers, robert.meijer}@tno.nl

[3]JIVE: Joint Institute for VLBI in Europe, Dwingeloo, The Netherlands; {kettenis, keimpema}@jive.nl

[4]CNRS: Centre National de la Recherche Scientifique, Lille, France; damien.marchal@lifl.fr

Keywords: distributed computing, network management

Introduction

A recent collaboration of computer scientists with radio astronomers consisted in running world-wide real-time experiments using Grid-based software correlator for radio telescope images [1]. A radio-telescope produces a lot of data. All the data need to be transmitted to a central place for correlation, which combines the individual observation into one single higher-clarity result. The transmission process was done using postal service in the past, but today the tendency is to use optical network technologies to transmit these data from radio-telescopes to the computation centre such a grid [2].

For example, SCARIe is a Software Correlator Architecture Research and Implementation for e-VLBI (Very Long Baseline Interferometry), which needs to perform signal correlation of multiple telescopes in real-time during an astronomic event. Therefore, SCARIe requires good guarantee on both processing power and network communication. In addition to the real-time constraints, SCARIe also requires dynamic resource usage because the Earth rotates and only a part of the telescopes can observe the target on the sky at a moment.

Running SCARIe on Grid is difficult because Grid infrastructures are mostly focused on large batch-like jobs and communication channels are mostly best-effort. In other words, Grids need to provide some resources guarantee, in addition to the CPU and storage, also for communication services. We have analysed the specific requirements of SCARIe application when running in Grids and come to a framework that support an optimal interworking of both computational and network resources.

VLBI on the Grid

Figure 1 shows a basic deployment of SCARIe application on a grid. The application requires grid nodes for different purposes, as follows:

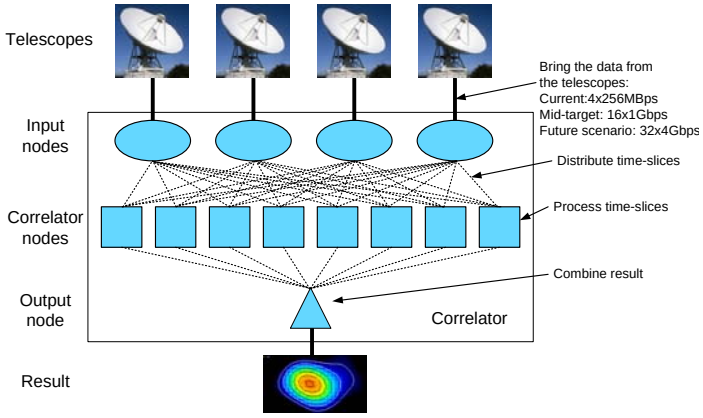

Figure 1. Distribute the SCARIe application on a grid: one input node for each telescope to distribute the workload, a certain amount of correlator nodes for signal processing, and one single output node for merging the results.

- *Input nodes*: one input node for each radio-telescope streaming data in the application. Input nodes extract timestamps and independent frequency bands from received data and send time-slices to the correlator nodes;
- *Correlator nodes*: nodes that perform the signal processing (delay, correlation) and send the correlated result to the output node;
- *Output node*: one single output node to collect and sort the correlated slices into a final output stream (to disk or stream back to astronomer);
- *Manager node*: one single node that initializes the nodes and delegates time-slices to the correlator nodes;

In addition to the mapping of application on the grid nodes, the workflow of the application uses two data flows: (1) the control messages and (2) the effective telescope signal that needs to be processed. The control messages are exchanged between different nodes in order to regulate the correlation. The control messages are a low bandwidth stream and they use message-passing interface standard (MPI). The telescope signal requires a high bandwidth due to the high sample rate used by the telescopes. Currently, SCARIe application handles the telescope signal by using TCP streams between nodes, but other communication protocols such as RTP could be implemented, too.

Although the first experiments done with SCARIe in DAS3 grid [3] worked for the minimal setup (4 telescopes streaming 256Mbps each, see Figure 1), one of the most important problem of SCARIe on grid relates to the networking capabilities, especially when going to higher data rates. Despite there is 1Gbps and 10Gbps networking capabilities in grid, the network does not provide a constant throughput for SCARIe application due to the unpredictable bandwidth usage by other applications running in Grid.

The future demands of SCARIe application already envision using of 32 telescopes, each streaming up to 4Gbps. In order to

allow SCARIe application running on grids with such future demands, we need to provide the following grid characteristics:

- Control the optical networks between radio-telescopes and the grids facilities;
- Constant throughput between the nodes involved in SCARIe application;
- Flexibility in choosing specific network characteristics to be guaranteed (e.g., low-delay, high throughput, etc);
- Ability to add/remove nodes on the fly during the experiment due to the change in the application requirements in terms of both networking and computational resources.

A grid could provide such networking characteristics if the network resources are integrated into the grid middleware. Hence, network resources can be claimed dynamically by any application, similar to the computational resources are in used nowadays.

Network control in distributed computing

We propose to provide control over network resources in distributed computing by (1) enhancing a grid middleware with a network broker and (2) use a programmable network that manipulates the traffic according to the requested network services. We used WS-VLAM [4], a grid workflow execution environment, to support coordinated execution of distributed Grid-enabled components combined in a workflow. Each Grid application is encapsulated in a container, which takes care of state updates to the workflow system and provides an execution environment that allows the manipulation of various aspects of the application. WS-VLAM was extended with support for network resource management with the aid of a network broker. NetBroker uses streamline [5] as a traffic manipulation system installed in every distributed node.

In Figure 2, the architecture we propose adds a Network Broker (NetBroker) component (5) with functions similar to grid brokers (4). The NetBroker acts as a single point of access to network resource management and provides capabilities to query and request network resources. The workflow engine is extended with an additional service to include discovery, allocation and provisioning of network resources and services via the NetBroker.

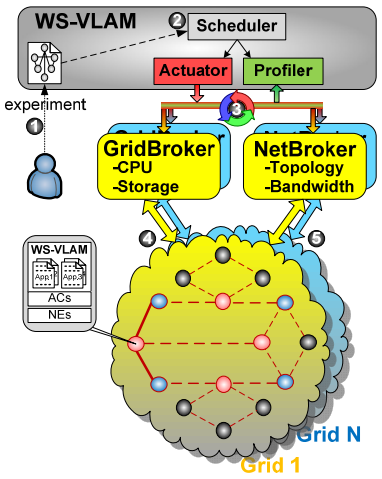

Figure 2. Grid architecture that includes programmable network services.

The framework of the grid middleware extended with network management support works as follows:

1 - ***User*** deploys an experiment: application & basic infrastructure requirements;
2 - ***WS-VLAM*** maps the experiment using *Actuator* onto available Grid resources which were detected by *Profiler*;
3 - **Control loops** may occur in which WS-VLAM is a controller to adjust the resources such as to solve the applications demands regardless of the environment changes;
4 - ***Broker*** manages the computational resources;
5 - ***NetBroker*** programs the networking infrastructure of Grid;

Each grid node supports the applications running under WS-VLAM supervision and provides the application-specific network services through application-components ***ACs*** as supported by network elements ***NEs***.

Experiments and Results

We setup a test bed to experiment with the effectiveness of our approach. All the nodes in the test bed run Globus Toolkit 4 and the programmable network software at the kernel level. The network broker and WS-VLAM run on separate machines over a control network. Figure 3 shows our test bed in which nodes are interconnected through two networks, as follows: the default network uses a shared 100Mbit switch and the second network uses an IDXP2850 network processor unit programmed to route IP packets at 1Gbps.

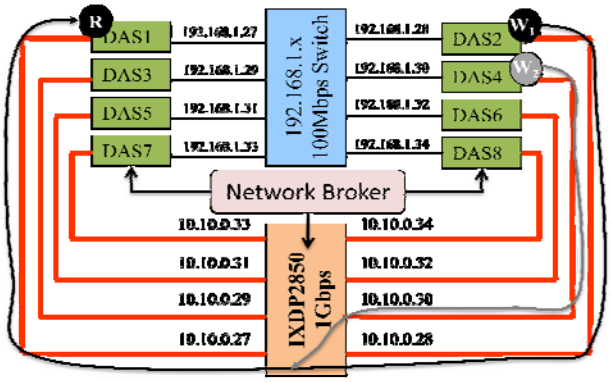

Figure 3. Setup of network connectivity in the test bed and first experiment. The network broker accesses the nodes over a separate control plane.

Figure 4 shows a screenshot from the workflow manager with multiple workflows. The workflow manager starts workflows one by one: 1, 2, 3. When the network performance (throughput) measured by an application decreases below a certain threshold, the application will request better connectivity to WS-VLAM. WS-VLAM will then offload the resources of the requesting application from the 192.168.1.x network onto the 10.10.0.x network (e.g. path 4 moves to the 1Gbps network), yielding improved performance.

The performance of our test bed is illustrated in Figure 5. Due to the shared 100Mbps, the per-path performance decreases while more paths are established and exchange data traffic at maximum. The switch offers one single network service: best effort. The application running in the workflow (*bwMeter* in Figure 4) measures the throughput and when it reaches a programmable *LO* threshold, it sends a request for more resources to WS-VLAM. Next, NOS receives a demand for better paths and decides to create alternative paths over 1Gbps network. Then we see that the throughput as measured by *bwMeter* increases in the second part of Figure 5.

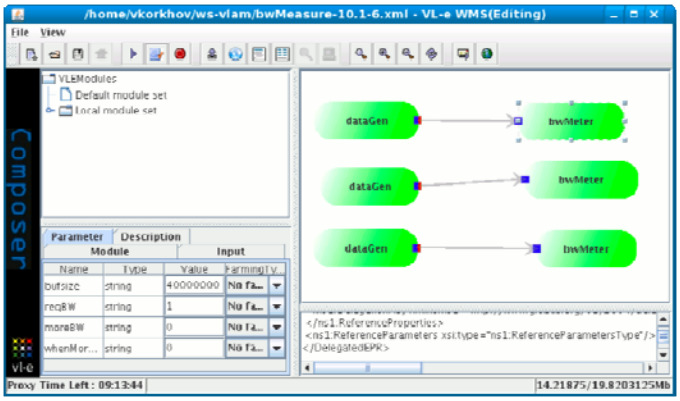

Figure 4. shows a WS-VLAM workflow where multiple producers (dataGen module) and consumers (bwMeter) are connected.

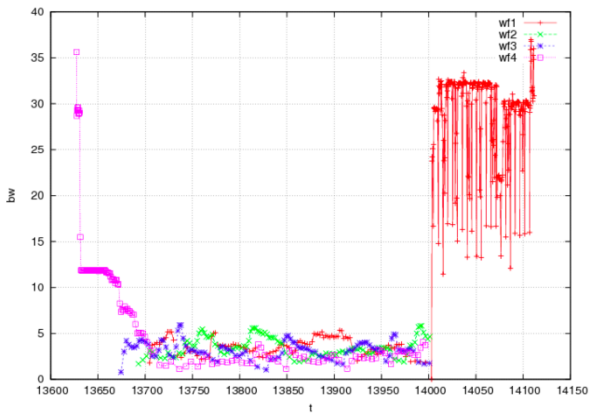

Figure 5. Experimental evaluation of test bed (bw in MB/s and t in seconds).

Conclusions and Future Work

We believe that current and future large-scale, high-throughput, with both computational and networking resource guarantee systems will migrate towards Grids computing. The scale of such systems makes it unfeasible to over dimension supporting infrastructures. At that moment, network resources are not transparent to applications anymore and will need to be managed to achieve an optimal interworking between sensors, networks, and computational resources in the Grid.

While most efforts addressed wired networks between Grids, in this paper we presented a systems approach to solve two major issues in development of scientific applications for Grids with specific network demands as met in SCARIe project: (1) enabling control of network resources in order to support application specific services such as soft real-time and (2) empowering the applications in Grids to control the available resources in order to allow self-tuning for an optimised way of resource usage.

While we believe that applications should have more control over network resources, it also raises fundamental issues. How can Grid applications manage and control potentially tens of thousands of nodes? What are the characteristics and limitations of applications that include management and control of networks? In other words, what is the balance between network services and application control?

References

[1] N. Kruithof and D. Marchal, "Real-time Software Correlation", in INGRID Workshop, 2008.

[2] A. Szomoru, A. Biggs, M. A. Garrett, et al., "From truck to optical fibre: the coming-of-age of eVLBI", in *7th European VLBI Network Symposium* Toledo, Spain, 2004

[3] Das-3 grid. http://www.cs.vu.nl/das3/.

[4] D. Vasyunin, A. Wibisono V. Guevara-Masis, A. Belloum, “WS-VLAM: Towards a Scalable Workflow System on the Grid” Workshop on workflows in Support of Large-Scale Science (WORKS 07) Monterey Bay, 2007

[5] H. Bos, W. d. Bruijn, M. Cristea, et al., "FFPF: Fairly Fast Packet Filters", in *OSDI*, 2004.

VITAE

Mihai Lucian Cristea received his B.Sc. degree in Automation and Computer Science Engineering in 1999 and M.Sc. degree in Artificial Intelligence in Process Control in 2000 from the "Dunarea de Jos" University of Galati, Romania. During his Ph.D. research between 2002 and 2006 at Leiden University, he contributed to the Streamline packet-processing framework at multi-gigabits speeds. Since 2007, at University of Amsterdam, Mihai is involved in the research of programmable networks, design and development of Token Based Networking including the Token Based Switch (TBS-IP) for light path selection at multi-gigabit speeds on behalf of the applications.

Rudolf Strijkers received a BSc and MSc (2007) degree in computer science from the University of Amsterdam. He currently pursues a PhD degree at the same university and works for the Dutch institute for applied scientific research TNO. His scientific interests are programmable and adaptive networks from an application perspective.

Vladimir Korkhov is a postdoctoral researcher at the University of Amsterdam. He received PhD degree from the University of Amsterdam, Faculty of Science after MSc degree in Mathematics and Computer Science from St.Petersburg State Institute of Fine Mechanics and Optics, Russia. His research interests are focused on resource management in Grid computing and distributed software systems, workload balancing in heterogeneous environment, and workflows on the Grid. He co-authored more than 25 research papers in journals and conference proceedings.

Adam Belloum is an Assistant Professor at the computer science department of the University of Amsterdam. He received the M.Sc. and Ph.D. degrees from the Compiegne University of Technology, France. He started his research activities in 1992, he first worked in the area of designing parallel computers (VLSI design), where he participated in the design and the implementation of a prototype of a

parallel computer dedicated to image processing based on DSP processors. In 1997, he moved to The Netherlands where he was leading the research activities in the area of Web caching until the year 1999. During the last 10 years, he was heading a small research group at the IvI, UvA, which has developed the first prototype of the run time system of the virtual laboratory toolkit. He published more than 50 articles in international conferences and journals and helped in managing three research projects: JERA project (1997-1999), the Virtual laboratory (1999-2002) and the VLe (2004-2009).

Mark Kettenis studied Technical Physics at the University of Twente in Enschede, The Netherlands. He did his Ph.D. at the Institute for Theoretical Physics of the University of Amsterdam under supervision of Dr. L.G. Suttorp and Prof. Dr. H.W. Capel. In December 2001 he successfully defended his Thesis titled "On the Inhomogeneous Magnetised Electron Gas". After working at a small IT services provider he joined JIVE to work on long-baseline user software for the RadioNET FP6 project. As a Software Project Scientist, he has been closely involved in the ongoing development of the e-VLBI. He has been leading the software correlator development at JIVE since February 2007.

Aard Keimpema studied physics and computer science at the University of Groningen. He then obtained his Ph.D. also at the University of Groningen in computational physics. He currently works as a scientific programmer on the SCARIe project where he works on the SFXC software correlator.

Damien Marchal received his PhD from the University of Lille (France) in the field of computer graphics and computer animation. He then worked as Post-doc on the SCARIe project focusing on interaction between SCARIe and the Starplane project. He is now working as engineer for the CNRS (National Center for Scientific Research). His research interests include field interpolation techniques and discreet differential geometry as well as logic based semantic reasoning and high performance distributed computing.

Paola Grosso received her Ph.D in Physics from the University of Turin in Italy. She joined the System and Network Engineering (SNE) group of the Universiteit van Amsterdam in 2004. She is lead researcher of all the group activities in the field of optical networking. She is involved in the GigaPort Research on Network project to develop models and control plane for lambda networks. Her research interests are provisioning and design of hybrid networks for lambda services; development of topology models for hybrid multi-domain multi-layer networks, design of network lightpaths architecture for 4K and beyond Digital cinemas.

Cees de Laat is associate professor of the System and Network Engineering Science group at the University of Amsterdam. Research in his group includes optical/switched Internet for data-transport in TeraScale eScience applications, Semantic web to describe e-Science infrastructure, distributed Authorization architectures and Security & privacy of information in distributed environments. With SURFnet he develops and implements projects in SURFnet7 Research on Networks. He serves as chair of GridForum.nl and board member of ISOC.nl. He is co-founder and organizer of several past meetings of the Global Lambda Integrated Facility (GLIF) and CineGrid.org.

Robert Meijer received a PhD degree in experimental nuclear physics of the University of Utrecht. He switched to ICT research at TNO in 1991 and became professor at the University of Amsterdam in 2002. His scientific interests are next generation networks and computer technologies with application-specific dynamic adaptation capabilities.