StarPlane: application-specific management of optical networks

1 Title, short name, investigator

a. Full project title: 'Starplane: application-specific management of optical networks'

b. Project short name: StarPlane

c. Principal investigator: prof.dr.ir. H.E. Bal

2 Summary, Abstract

2.1 a. Summary

As witnessed by the rising popularity of overlay networks, applications increasingly demand more flexibility from their networks. In e-Science the need is especially pressing. As more and more sites collaborate via wide area networks (WANs) in e-Science experiments, latency and bandwidth become major issues. This makes the topology and dimensioning of the network of vital importance (e.g., because queuing in intermediate hops adds to the latency). Moreover, it would be desirable to allow for network partitioning to prevent cross-interference by applications (e.g., to shield TCP-friendly connections from newly developed aggressive protocols on high-speed WANs). Unfortunately, in existing networks the topology and dimensioning of networks is fixed, and the number of hops between two nodes is immutable. Similarly, it is impossible for applications to request partitioning of network resources.

Optical networking will change the way networks are used. New control technology permits an application, in principle, to set up one or more end-to-end 'lightpaths', providing it with hundreds of Gbit/s of aggregate bandwidth fully dedicated to the application. This moves the bandwidth bottleneck, as the network outside the computer becomes much faster than the connections inside. Using lightpaths, applications may allocate networks much like they allocate physical memory. For instance, applications may configure and dimension their own network topology, which would then consist of true end-to-end lightwave connections without intermediate queuing, switching or cross-interference. This is in contrast to current work in overlay networks, where a virtual link often consists of multiple switching/routing nodes connected by different types of links.

Unfortunately, while it is technically feasible to set up lightpaths, there exist neither the management plane to let individual applications exploit this functionality directly, nor the knowledge of how to integrate optical networks with applications. The proposed StarPlane project addresses both of these concerns.

2.2 b. Abstract for laymen (in Dutch)

Applicaties hebben een groeiende behoefte aan flexibiliteit in het netwerk (zoals wordt aangetoond door de populariteit van 'overlay' netwerken). Topologie en dimensionering van een netwerk zijn bepalend voor de latency en bandbreedte die de applicatie ervaart. Daarmee zijn ze voor veel wetenschappelijke applicaties van essentieel belang voor de prestatie. Idealiter zou een applicatie zelf willen kunnen bepalen hoe zijn netwerk er uitziet. Het huidige netwerk is echter star: topologie en dimensionering liggen grotendeels vast.

Optische verbindingen zullen de manier waarop we met computernetwerken omgaan ingrijpend veranderen. De enorme bandbreedte van zulke netwerken betekent bijvoorbeeld dat de verbinding buiten de computer sneller is dan die binnen de computer. Daarnaast bestaat er, in potentie, de mogelijkheid om het netwerk in de toekomst veel flexibeler te gebruiken. Het is bijvoorbeeld mogelijk om een gedistribueerde applicatie een geheel eigen netwerk te geven bestaande uit lichtpaden (rechtstreekse optische verbindingen tussen de verschillende locaties waarop de applicatie draait). Op deze lichtpaden vindt geen routering, switching of buffering plaats. Voor het eerst is het mogelijk applicaties een eigen netwerk te geven zonder gebruik te maken van dure 'leased lines'.

Dit alles heeft grote gevolgen voor de manier waarop applicaties geschreven moeten worden. Helaas zijn er op dit moment twee obstakels die verwijderd moeten worden om applicaties in staat te stellen optimaal gebruik te maken van de nieuwe mogelijkheden. In de eerste plaats ontbreekt het aan de benodigde *management infrastructuur*: hoewel het fysiek mogelijk is om lichtpaden aan te leggen ten behoeve van applicaties, is er nog vrijwel niets gedaan om deze

capaciteit op eenvoudige wijze beschikbaar te stellen aan de applicatieprogrammeurs. In de tweede plaats ontbreekt de kennis over hoe applicaties gebruik kunnen maken van de nieuwe mogelijkheden (hoe moeten we ze aanpassen, wat is voor een applicatie het meest geschikte netwerk, etc.).

In het voorgestelde StarPlane project worden de methoden ontwikkeld om de toegenomen snelheid en potentiele flexibiliteit in optische netwerken toe te kunnen passen in applicaties. Het onderzoek zal de toegenomen flexibiliteit van optische netwerken ontsluiten, door applicaties op eenvoudige wijze hun eigen netwerken-topologieen te laten creëren en dimensioneren. Verder wordt de kennis ontwikkelt over hoe dit gebruikt kan worden in echte applicaties.

3 Classification

The proposal fits in the GLANCE program under the theme Management and Analysis.

4 Composition of the research team

Name	University	Position	Specialization
Prof. dr. ir. H.E. Bal	VU	professor	Parallel programming
Dr. ir. H. Bos	VU	assistant professor	Computer networks
Dr. ir. C.T.A.M. de Laat	UvA	associate professor	Internet and Grids
Prof. dr. P.M.A. Sloot	UvA	professor	Computational science

Table 1: staff members involved in the project

Professor Bal has much experience in parallel programming environments, grid computing, and applications. He is adjunct director of the Virtual Laboratories for e-Science (VL-e) project, which includes many research groups with data-intensive e-Science applications. He also is the main coordinator of the DAS-2 and DAS-3 projects. Dr. de Laat is head of the Advanced Internet Research (AIR) group at the University of Amsterdam (UvA). He has much expertise in computer networks (including ATM and optical networks) and grid computing; de Laat is a member of the steering committee of the Global Grid Forum and also plays a leading role in the GigaPort-NG project, which will build the next generation (optical) academic networking infrastructure in the Netherlands (SURFnet-6). He also participates in the NSF OptIPuter project. Dr. Bos' research interests include high-speed networks (including network monitoring), security (e.g., worm detection), and management. He was part of the switchlets/Tempest team in Cambridge that allows operators to partition ATM networks so that multiple virtual networks can be active simultaneously on the same physical network. He participates in several large European projects (SCAMPI, LOBSTER, NOAH) and heads the NWO/STW DeWorm project. Prof. Sloot brings in much expertise in computational science, interactive problem solving environments, and grid computing. He will act as supervisor for the requested Ph.D. student.

5 Research school

The proposers are members of ASCI, the Advanced School for Computing and Imaging.

6 Description of the proposed research

Optical networking technology promises to change radically the way computer scientists and application researchers use networks. Firstly, it will allow an unprecedented increase in wide-area network bandwidth, in the order of tens of Gbit/s per lightpath (optical link [30]). As a single distributed application may obtain many lightpaths across many physical links, aggregate bandwidths may easily reach up to hundreds of gigabits (or even terabits) per second. This alone completely reverses the location of bandwidth bottlenecks in distributed systems: the network outside the computer is rapidly becoming *much* faster than the network inside the computer (i.e., bus and memory speeds). Secondly, fully optical network infrastructures have the potential for applications to allocate links *on demand*. A lightpath is a direct physical point-to-point connection, not a virtual connection. So, links can be allocated very much like physical memory in a traditional computer system. It will even be possible to let applications modify the *topology* and *dimensioning* of the network at runtime, depending on the (changing) application's needs. Whereas most current wide-area links are still static, fixed low-bandwidth connections, the future infrastructure provides dynamic, flexible, high-bandwidth

connections. Many scientific applications exist that require the bandwidth and flexibility provided by optical networks, ranging from high-energy physics (e.g., CERN's LHC project) to astronomy (e.g., VLBI). With the development of computational grids and virtual laboratories, even more applications will arise that store, process, and visualize huge amounts of distributed data (see Section 6.3).

However, realizing this enormous potential of optical networks is difficult, and many problems need to be solved first. For example, much research is already being done on optimizing protocols like TCP/IP for optical networks [15]. In our view, the most fundamental open problem now is how to integrate optical networks with the scientific applications that need them. This integration is still poorly understood, because current applications (and operating systems and grid middleware) do not look at networks as resources that can be allocated and managed. Hence, mechanisms, interfaces, and policies will be needed that allow applications to manage optical network links, taking into account that the entire optical infrastructure will be shared by many applications.

We therefore propose to develop a management plane, called *StarPlane*, that allows applications to manage optical network links. In our proposed project, we will both develop the StarPlane management infrastructure and study how applications can use it. Thus, future applications are able to make optimal use of optical networks, rather than continue to use them as if they were simply faster versions of the existing inflexible infrastructure.

The project's focus is mostly on e-Science applications, both because these applications are well-positioned to benefit from the added flexibility, and because the e-Science sites (e.g., clusters and supercomputers) are the first to be fitted with the necessary optical infrastructure. However, we argue that the results from this project will be beneficial to other application domains also. Indeed, it is realistic to expect that larger universities and organizations will have full access to advanced optical network technology in the near future.

6.1 Demand and timeliness

We think this proposal is timely, because the Netherlands will obtain an optical network infrastructure (SURFnet6) in 2005 and much research already started here in 2004 on the technical aspects of optical networks in the "GigaPort Next Generation" project and on data-intensive applications in the "Virtual Laboratories for e-Science" (VL-e) project. Both projects have obtained about 20 million euro funding from the Dutch government for their research. Our project aims to develop the necessary glue to connect the raw resources provided by projects like GigaPort-NG with the virtual laboratories of projects like VL-e. It will also use the DAS-3 grid infrastructure that has been requested in a separate NWO equipment proposal (although we do not depend on the acceptance of the DAS-3 proposal). The StarPlane project will thus take optimal advantage of the new infrastructure (SURFnet6 and possibly DAS-3) as well as from ongoing research projects in the Netherlands (GigaPort-NG and VL-e), especially as we participate in all these projects. We also collaborate with the NSF OptIPuter [28] project, but our work is different in that we aim at very short time scales (subseconds) for allocating and modifying links, giving applications much more flexibility than with OptIPuter's long-lived connections.

At the same time, we observe that from the application side there is a clear demand for more network flexibility, both in e-Science and elsewhere. This is illustrated, for instance, by the large number of overlay networks proposed in recent years. An overlay network realizes a 'virtual topology' on top of the physical network. It provides applications on the overlay with a rather weak illusion of a private network that is to some extent isolated from the rest of the network. Unfortunately, overlays are a poor match for isolation, as the underlying network provides absolutely no resource guarantees. Moreover, to the applications overlays often result in unpredictable behavior, because the physical resources are no longer visible (e.g., when two nodes are directly connected by a virtual link which in reality consists of a sequence of congested routers). An analogy exists in virtual machines (VMs) where high-level virtualization often is not sufficient for applications that need explicit and low-level control over the host's resources, leading to the growing popularity of low-level approaches like Xen and VMware [9, 19]. We are proposing a similar solution for the network resources.

The demand for flexibility exists *a fortiori* in scientific applications where a direct link between two nodes in a wide area network instead of a multi-hop connection makes a huge difference to latency-sensitive applications and the need to shield TCP-friendly connections from more aggressive TCP-like protocols is pressing. As different applications have different (networking) needs and because it is impractical to connect physically every node with every other node, there is a strong case for a management infrastructure that allows one to modify the network topology and dimensioning on the fly.

Currently, two obstacles must be removed before the attractive properties of optical network technology can be exploited: (1) the absence of an easy-to-use management infrastructure to unlock the functionality and (2) the lack of knowledge of how to use the advanced functionality offered by novel networks in applications. Both are addressed in this proposal.

6.2 Scientific question and expected results

Scientific applications are increasingly executed on large-scale distributed resources. Examples of such resources include the DAS/DAS-2 distributed supercomputer in the Netherlands [26], the French Grid'5000, UK Grid, the US TeraGrid, and others. Sharing resources from multiple sites to solve complex problems or perform collaborative research also presents important problems to the application programmers. Often, the infrastructure is complicated and heterogeneous. To a large extent, heterogeneity can be solved by hiding it behind virtual machines. For instance, projects like Legion [16] and Ibis [20] have shown that virtual machines in user space are well-suited to implement grid applications, even for high-performance applications. As mentioned before, virtualization can also be pushed to the lower levels, leading to the hard partitioning of a node's resources that we mentioned earlier. This is provided by VMware and Xen, and also by our own Open Kernel Environment [12].

Unfortunately, until now such virtualization was not possible for network resources. For instance, for some given application, a 10 Gbit/s ring may be the optimal topology, while the physical network may have a completely different topology (e.g., a star) and bandwidth. We may virtualize the network at the IP level by mapping a ring on top of the real topology, but this is likely to lead to increased latency, jitter and possibly packet loss, due to the additional queuing, switching and routing points to what ideally would be direct connections. Note that this is also true for QoS-aware network technology, such as ATM, MPLS, etc. Moreover, in most existing networks there are no resource guarantees, so applications suffer from interference by other applications. It would be desirable to allow for network partitioning to prevent cross-interference, for example to protect TCP-friendly connections. Finally, the protocol stack to be used by the application is often determined *a priori* (typically variations of TCP/IP) even though these protocols may not at all reflect the application's needs.

Another problem with resource sharing is that resources may be in use by other applications. This problem can be addressed by resource partitioning and reservation. There are many such systems for computing nodes, but for networks such systems do not exist at this level. Preliminary work in network partitioning and reservation was conducted in ATM networks and involved one of the researchers at the VU [10, 11]. This shows that network resources can be partitioned at higher levels, but not without extra switching/routing and, hence, potential congestion and latency. Recently researchers have also worked on optical networks, but the timescales that are considered are much longer than what is required in our project [30]. Existing approaches aim for long-lived (semi-permanent) connections and are therefore not suitable for the StarPlane which targets connections that can be changed in less than a second.

Moreover, the *ability* to partition resources is merely a first step toward unlocking the flexibility provided by optical networks. The second step is to supply applications with just the virtual network *they need*, without requiring them to supply detailed resource descriptions in complex resource specification languages (RSLs). The third step is the use of the resources. Some applications may be expected to request resources at a very fine level of granularity, and once granted, keep all control in their own hands. However, for many others this is too complex. Rather, they would like to express their needs in simple terms, for example "an IP network in a ring topology with high-speed links". The behavior of the allocated infrastructure should not be noticeably different from that of a matching physical network.

To realize this goal, we do not propose to (re-)invent resource description languages or job-description standards. Instead, we plan to build on existing solutions and employ a broker service to generate the appropriate resource specifications automatically from high-level requests. The broker maintains knowledge about the resources and uses inference to determine what resources are needed, given a high-level description and local policies [18]. As the success of the project will be determined to an important extent by the ease at which it is picked up by users (and bearing in mind the failure of complex resource reservation schemes in ATM in the past), we consider this component a crucial part of the StarPlane project. Similar inference-based approaches are now emerging as an important new direction in self-managing systems, in both academia[1] and industry [13, 21].

Note that the StarPlane approach does not preclude building networks in which links are shared in a best-effort manner. On the contrary, shared best-effort networks may coexist with specialized dedicated networks and may cater to a large set of applications.

In summary, many distributed applications stand to benefit from a management plane that provides clients with application-specific sets of resources and protocols. The research question of the proposal is twofold. First, we investigate how and at what granularity applications can be given access to the potential flexibility of optical networks. Second, we research how applications may best exploit the provided flexibility. The expected result is the implementation of a management plane that provides the required functionality and demonstrable knowledge of how to integrate it with a set of real applications. The management plane should make it easy to request, manage and use these resources, and access should be allowed at various levels of granularity. As a long-term result, the software implemented in the StarPlane will

[1] An existing research-project at the VU is using the approach for management of distributed resources, including a case study for automatic handling of failures in the PBS job submission system: `www.few.vu.nl/~wdb/betagis`. It involves both researchers from the VU.

facilitate the integration of the resources of future networks with the demand of future applications.

6.3 Research approach and methodology

For clarity, we will first make our goals explicit. In the proposed StarPlane project, we will develop support for:

1. fast, application-specific allocation of the network resources with deterministic characteristics;
2. application-specific composition of the protocol stack that is used to control the resources;
3. low-level resource partitioning (and, hence, no interference);
4. high-level requests (whereby policies and inference are used to assist the user).

To achieve and validate these goals the project will deliver the following software:

5. the implementation of the StarPlane management infrastructure;
6. the implementation of an intelligent broker service to handle high-level requests;
7. the modification of a set of real applications to exploit the functionality of such a management plane;
8. a library of 'standard' components (protocols, middleware) to support and build new applications.

The StarPlane project allows applications to allocate on-demand the required resources, such as nodes, links, protocols, etc. The management plane assists the user in determining what needs to be allocated, by employing a knowledge-base and inference engine. The knowledge-base stores knowledge not just about the hardware and software resources available at different sites, but also about applications and even abstract concepts (e.g., it may know what it means if an application requests a 'ring topology').

By allocating resources at the lowest possible level, the resulting configuration aims to be indistinguishable from a physical allocation of resources. Specifically, we partition the resources for different applications, and guarantee that if there exists a direct link in a *virtual* topology, it will not incur queuing in any intermediate hop, even if there is no direct connection in the *physical* topology. StarPlane's approach to realizing this goal is to connect direct lightpaths between two nodes of the application (see also Section 6.4.1). Part of the application's components may be pre-defined (e.g., an existing protocol stack), while another part may be controlled directly by the application. To our knowledge, no such management system exists today, mainly because handling network resources at this level is difficult.

After the resources have been provided, they can be used by the application by means of *control* operations. The StarPlane approach allows for different control architectures to be active simultaneously. For instance, while one application uses TCP Reno, another may opt for a TCP version that is optimized for high-speed optical WANs (of course, different protocols may be used for different links). In an extreme case, control architectures that are radically different can be used simultaneously by different applications (e.g., ATM and IP/POS). Indeed, as the network resources are completely under the control of the application, this is an ideal environment to experiment with entirely new protocols, without worrying about interference with other applications. The development of new transport protocols is currently a very active research domain, especially since standard TCP performs so poorly on fast WANs.

We will also adapt a set of applications to exploit the features offered by the management plane. As it is our goal to provide generic support for e-Science applications, we will develop a library of components that applications may use to compose their own preferred configuration of resources (e.g., protocol stacks, authorization services, local policies, etc.). Policies are used to decide who is allowed to manage what resources under what conditions. In this way we build on the well-established concepts of policy-based management (see [18, 23]) and already existing policy-specification languages (e.g., Ponder and PDL).

An important aspect of the project is to use real applications to validate our ideas. Although many applications are said to be 'data-intensive', it is still an open question which applications can beneficially use bandwidths of 10-1000 Gbit/s. Still, there are many important applications that do have such extremely high requirements:

- In the VL-e project, the subprogram 'Data intensive science' studies applications from earth observation, astro-particle physics, and high-energy physics that use very large amounts of data. A good example is the CERN LHC (Large Hadron Collider) project. The data storage center in Amsterdam (at SARA and NIKHEF) is expected to deal with 1 PetaByte of data per year from this project by 2007. Other subprograms in VL-e deal with medical imaging data and bioinformatics databases, which also are very large data sets that are accessed remotely.

- Many distributed scientific instruments produce massive amounts of data at different locations that have to be processed remotely. Depending on the application, the data may have to be gathered at a central location (e.g., to do correlations) or may have to be scattered (e.g., CERN has remote users in most countries). VLBI (very long baseline interferometry) data in astronomy is a good example. The bandwidth requirements of these data transfers reach many hundreds of Gbit/s. In the search for Ultra-High Energy Cosmic Rays in the LOFAR astronomy project, each event generates 100 GB of data that needs to be transferred to a data collection point. Per day there may be many such events. Similarly, the data rate from a LOFAR station to a collector is approximately 10 Gbit/s, and the rate between the central core and central processor may be higher than 600 Gbit/s.
- Large scale computer simulations often also produce huge amounts of distributed data that have to be exchanged in various ways. Such applications often use complicated communication patterns, such as broadcast, scatter/gather, reduction, and all-to-all exchanges, each of which puts different demands on the underlying network topology [25]. The group at the VU has ample experience with distributed supercomputing applications and has done much research on running (non-trivially parallel) applications on large-scale geographically distributed systems and grids (e.g., GridLab and DAS-2). A good example is heuristic search using transposition tables, for which we developed an asynchronous, bandwidth-intensive algorithm that can run efficiently on a grid with sufficient bandwidth [6]. We have used a similar latency-insensitive algorithm to computationally solve the game of Awari [17]. The bandwidth requirement of this application increases with CPU speed and will exceed a Gbit/s *per node* for modern CPUs. If executed on a large-scale grid, the application would require on the order of 100 Gbit/s wide-area bandwidth.
- Often, the output of instruments or simulations has to be visualized remotely, also resulting in huge bandwidth requirements. Several demonstrations of visualizing large remote data sets have been given over the previous years at conferences like Supercomputing and iGrid. Optical networks could be a key enabling technology to make further progress in this area. VL-e also contains a subprogram that focuses on 'visualization on the Grid'.
- Workflow applications often need high bandwidth and low latency and operate across a WAN. Workflows are an integral part of the Globus4 grid toolkit and are also studied extensively in VL-e.

Our approach will be to study several of these applications, using our contacts with VL-e and other national and international projects.

6.4 Organization of the research project

We organize the project in two tracks: (I) the basic StarPlane management infrastructure, and (II) the applications and their needs. In track (I) we implement the core components that are needed to build the StarPlane. At least one control architecture will be built, but the focus of track (I) is mostly on how to enable the lower layer infrastructure to change the topology, re-establish connections after a change, and reveal the topology to the higher layers. As it is not easy to process data at 10 Gbit/s, we also study how to make use of the high-speed connections. In previous work we have shown that a single node is able to handle high link rates if packet copies and context switching are minimized [2].

In track (II), we will use our experience in the field of distributed applications to investigate optimal choices for several applications. The control architectures for these applications will be implemented on top of the core StarPlane. An important aspect of the research will be comparing the performance of application-specific networks with that of a fixed, simple topology. In other words, we evaluate for what applications and under what circumstances the additional flexibility improves the overall performance. In this track, we will also develop a service that uses automated reasoning to provide clients with the network infrastructure they need (and help them in managing these resources), based on simple requests and a knowledge base. As mentioned previously, similar approaches are emerging as an important new direction in self-managing systems. Track (II) serves two purposes: on the one hand, it serves as validation for track (I), and on the other, as research into a knowledge based approach to managing a complex system.

6.4.1 Track I: the basic StarPlane infrastructure

The ability to allocate and connect light-waves end-to-end at short time-scales has only recently become feasible. The key idea is that we exploit the abundance of dark fiber provided by new infrastructure projects (e.g., the SURFnet6 optical network of the GigaPort-NG project). Consider for instance Figure 1 which shows the topology of the DAS-3 distributed cluster computer (subject of an existing project proposal). SURFnet has agreed to dedicate a full band of up to eight lambdas[2] between each two connected nodes in the ring. These will be available for direct control by the

[2] A lambda is a lightwave with specific wavelength.

StarPlane researchers. As a result, we will allow network users to request direct point-to-point lightpaths to their peers. These connections are very basic 10 Gbit/s links, usually with Ethernet framing. The Ethernet frames are put directly on their own lambda on the nation-wide fiber infrastructure and do not enter through congestion prone switches or routers. The connections therefore have deterministic and known performance characteristics.

Using the StarPlane, applications are able to interact with the switching infrastructure to connect, for instance, the nodes in Figure 1 in star, ring, and other topologies. Applications are limited only by the (growing) number of lambdas and lambda consumption by other applications. Thus the network becomes an allocatable device for the computer, hardly different from memory and other local resources. The StarPlane will provide this abstraction, taking into consideration access control and authorization policies.

The StarPlane project will work closely with the DAS-3 project and SURFnet to implement the vision of a DWDM (dense wavelength division multiplexing) backplane for grid clusters. If the DAS-3 proposal is not funded we will still be able to research the main concepts outlined in this proposal, although with a more limited testbed (see Section 6.4.3).

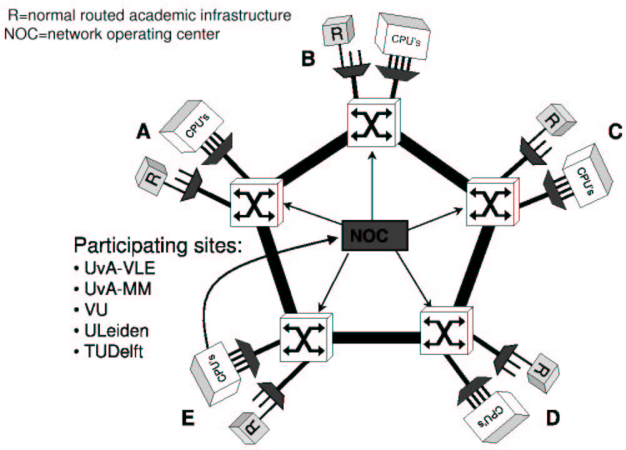

Figure 1: Proposed DAS-3 configuration

Lightpath establishment. We intend to investigate and pilot two complementary ways of lightpath construction. One way is to interact with the SURFnet Network Operations Center (NOC) by means of requests that use web services technology for accessing dynamic elements in NetherLight and the Common Photonic Layer (CPL) elements in the nationwide network [31]. This allows for switching 10 Gbit/s connections in the network between the sites and through NetherLight and the Global Lambda Integrated Facility [29] to international destinations. This approach builds on work done in de Laat's AIR (Advanced Internet Research) group at the UvA.

The other way is to add optical switching equipment at the client sites. A Micro Electro Mechanical Switch (MEMS) costs about one/tenth per port compared to a layer-2 Ethernet switch. These client side switches need to be grid-enabled by virtualizing them using web services. The compute nodes and other grid resources can then be equipped with 10 Gbit/s Ethernet cards which will all be connected to the optical switch (Figure 2). The individual DWDM lightpaths from SURFnet6 would also be connected to that switch. The switch can connect individual nodes to the lightpaths under control of the application middleware. The combination of the two will give a very powerful dynamically switched network.

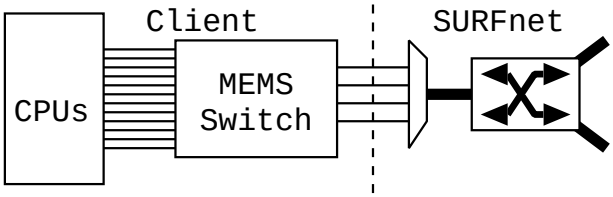

Figure 2: Configuration at the client

The management/control plane is key to the realization of the goals listed earlier. World wide there is a handful of projects working in this domain, from different angles. We will discuss other projects in Section 6.6.

6.4.2 Track II: Interaction with applications

In track II we investigate the higher levels of the problem space which involve the interaction with real applications. This includes both adapting existing applications to exploit the increased flexibility of the network, and making the low-level features of track I available in an easy-to-use manner.

To adapt existing scientific applications we need to identify the ideal topology and dimensioning of the network for the applications, and also determine the appropriate protocols to be used. As current networks are a given, there is little knowledge in this area.

After the most appropriate environments have been identified we need to build the control architectures for using and controlling them. The StarPlane project aims to deliver the means to compose the most appropriate environment for new applications by reusing small and basic components. Therefore, research is needed into the granularity, boundaries and interfaces of these components. While lightpaths are obvious candidates, the scope of the project also includes links, protocols, control and management. Note that promising results in this area have been achieved in related, albeit

more restricted domains, in projects like STREAMS (AT&T), x-kernel (University of Arizona), and x-bind/Genesis (Columbia University). Similarly, there has been work in grid computing to collect resources at multiple sites on behalf of an application. However, none of these projects have addressed low-level partitioning and replumbing as required for StarPlane.

We will also develop libraries of components that can be used to construct the appropriate environment for an application. A network with an appropriate control architecture can be constructed using these libraries. As will be explained below, however, we will also encode explicit knowledge about how to use the components, so as to automate most of this work.

Of crucial importance is the interaction between applications and StarPlane. On the one hand, we want to provide maximum flexibility. On the other hand, the average programmer should not be required to use elaborate resource specification languages (RSLs) just to obtain the necessary resources. It is important to lower the threshold for using the advanced features of a system, otherwise users will get lost in the complexity of resource specification procedures. The proposers experienced this problem at first hand both in grid computing (complex RSLs) and ATM (which failed partly because of the complexity [24]).

For this reason, we plan to build a high-level broker service that employs explicit knowledge and inference to translate high-level requests to low-level resources. We will clarify the idea with an oversimplified example of a workflow application. Suppose a user wants to run an e-Science application A and have its output rendered by a tool T. The broker has knowledge about A and T. T may exist only on a specific cluster and require input in format f_1. A may be a single-cluster application that is available at a number of sites and produces high-bandwidth data in output format f_2. With this knowledge the broker may infer an appropriate topology and dimensioning of the application-specific network (also taking into account security and usage policies). It may even infer that if $f_1 \neq f_2$, conversion is needed. If the converter is only available on a third cluster, it will include connections to and from this cluster. Furthermore, it may choose an appropriate protocol stack to instantiate the application.

Most of the job of resource management is taken out of the hands of the programmer who may focus on application writing (although, if desired, low-level direct access will still be possible). It is important to note that the StarPlane, where possible, will employ existing tools and protocols for resource management, including schedulers, reservation schemes and signalling protocols such as RSVP.

6.4.3 Dependencies

A potential dependence between track I and track II is that the StarPlane core should be finished before the applications can be modified so as to use it. However, as we treat lightpaths as simple bitpipes, we will remove the dependence by initially using an overlay and simple TCP/IP pipes. The only dependence is that the interfaces to the StarPlane should be determined early. In Section 7 we show that our planning reflects this.

Another dependency is that we need a management infrastructure for our experiments. Our ideal scenario here is to use the DAS-3 system. If the DAS-3 proposal is accepted, we will have an outstanding testbed consisting of five cluster computers connected by SURFnet-6, as described above. This testbed would be available early 2006. In this case, we will use the StarPlane hardware budget to extend the switching equipment at the VU (financed by the DAS-3 proposal), to allow a large set of nodes of the DAS-3 cluster direct connectivity to the StarPlane management infrastructure.

If the DAS-3 proposal does not get funded, we will set up a smaller scale experimentation environment as follows. We will use the StarPlane hardware budget to install an optical switch at the VU and use it in combination with the already available optical switching equipment at the UvA. As compute nodes we will use an Opteron cluster at the UvA and an existing cluster at the VU. These clusters will be logically partitioned to emulate multiple 'sites', and these sites will be connected via lightpaths. The lightpaths can go through remote locations, allowing us to emulate long-distance connections even between nodes within the same cluster. In this way, we are able to implement any testbed we want, even though the actual partitions may be smaller and located at close proximity of each other. Therefore, we can still follow the StarPlane approach that is proposed in this project, albeit in a smaller scale than with DAS-3.

6.5 Scientific importance of the proposed research

Importance for e-Science The Netherlands will be spending about 80 million euro research funding in total over the next 4-5 years to set up an optical network infrastructure and to develop virtual laboratories for e-Science. However, bridging the gap between such an advanced infrastructure on the one hand and the complex applications on the other hand is challenging. Essentially, fundamental research is required to answer many difficult open questions (see Section 6.3). Therefore, if our project is successful, it may have a huge impact.

Benefits to other research domains Observe that the ability to construct, use and delete fully controlled, resource-safe virtual networks provides benefits to domains other than scientific computing as well. For instance, in network research it may be used to experiment with disaster scenarios such as the outbreak of worms and viruses. Because the resources are partitioned, the effects will never propagate to sites beyond the limits of the application resources. In addition, programmers may use the partitioning to separate different types of traffic, for research in QoS provisioning, security, routing, and signalling. These are all of major interest to network operators that may share a single physical infrastructure, while providing their QoS related service level agreements. The same need for full isolation exists for specific target groups (e.g., financial corporations and the military). While none of these areas are addressed in this project, they will all benefit directly from the results.

6.6 Relationship with research done elsewhere

Some of the issues addressed in this proposal have been studied earlier with limited success in the context of ATM and SONET networks, but with optical networks real connections (lightpaths) rather than virtual connections are managed, making the problem quite different (e.g., queuing and switching/routing can be eliminated completely).

An early attempt in ATM at providing similar functionality as provided in the StarPlane is known as the Tempest/switchlets work at the University of Cambridge and involved one of the researchers at the VU (Bos) [10, 11]. The network control was devolved from the switches and virtual networks were constructed by partitioning the switch resources. Unfortunately, some resources (e.g., switch buffers) could not be fully partitioned and there was no access to the hardware's admission control procedure. In addition, the nature of ATM does not allow for direct (unswitched) connections between any two points in the physical network, as intermediate switches always process the ATM cells.

The key to enable lightpath setup is the control plane implementation for the networks involved. World wide there are around five initiatives to tackle this complex problem from different angles. One approach is to virtualize all elements in an optical network and publish all components using a package named UCLP (User Controlled Lightpath) developed by Canadian researchers [30]. In this approach users are empowered to collect, organize, subdivide and exchange components in the architecture with peers and thus construct networks. Another approach is called Photonic Domain Controller and Photonic Interdomain Negotiator, developed by EVL in the OptIPuter project. This approach builds on GMPLS/ASON which is augmented with topology, routing and negotiation functions to make it work across domains by using RSVP-like protocols. The Just In Time (JIT) approach developed by MCNC aims at setting up the path by hardware assist at the moment the data starts to flow. The AIR group at the UvA is tackling the problem of authorization of the usage of resources in different domains and approaches the control plane from that angle. A complete solution is most probably a combination of several of the above mentioned approaches as no method currently covers all aspects of the control plane needed for client initiated lightpath setup through different domains on very short time scales.

The NSF funded OptIPuter [28] project, in which the UvA participates, aims at the re-optimization of the entire grid stack of software abstractions, learning how to "waste" bandwidth and storage in order to conserve "scarce" computing in this new world of inverted values. The StarPlane project takes this vision one step further by integrating the national multi-color optical networking capabilities between the participants in the DAS-3 project in the pool of allocatable and modifiable resources on very short time scales (subseconds). This differs significantly from the CANET4 [30] and OptIPuter projects as they build on fairly long lived optical connections. Therefore, the StarPlane project will be the first to aim at coupling a nationwide optical network with the workflow management middleware of a grid application at short time-scales.

6.7 Embedding in our current research projects

The proposed project will be embedded in the GigaPort research projects at the Advanced Internet Research group at the UvA. Current research in the group includes web-services techniques to virtualize components in optical networks, transport mechanisms for high speed bulk data, and multi-domain authorization architectures and implementations for the usage of multiple resources in different administrative domains. StarPlane will complement the ongoing research in this group by concentrating on very fast DWDM-based lightpath provisioning. The project will also be embedded in the high-performance distributed computing group at the VU (Bal, Bos), which currently studies programming environments, applications, and networking issues for large-scale grids, as well as packet handling at multi-gigabit rates (e.g., in the EU SCAMPI and LOBSTER projects).

The proposers participate in two large BSIK projects: the Virtual Laboratory for e-Science and the GigaPort Research on Networks project. The StarPlane project aims to develop the necessary glue between these two projects by bridging the gap between the applications (in VL-e) and the networking infrastructure (in GigaPort-NG). StarPlane will be embedded in the ASCI research school using the DAS-3 infrastructure. It will also use the international optical

infrastructure GLIF. Obviously StarPlane will closely collaborate with OptIPuter, in which the UvA has participated since the beginning.

7 Work program

The proposed timeline for the project, including an indicative set of milestones, is shown in Figure 3. Track I will be done by a Ph.D. student (4 years) and track II by a postdoc (3 years). The Ph.D. student will become a member of ASCI and will follow the educational program of ASCI. The programmer will be shared by both tracks. Initially the programmer will assist in implementing the core code of the StarPlane. In a later stage the programmer will be instrumental in integrating the two tracks.

Year	Track	Milestone
1	I	(i) analysis of available lightpath equipment and solutions (ii) preliminary design of the StarPlane (e.g., interface definition)
	II	(i) analysis of requirements of e-Science applications (ii) initial design of intelligent broker (e.g., interface definition)
2	I	implementation of StarPlane and prototype control architecture
	II	implementation/modification of a set of applications to use the StarPlane/broker functionality (overlay networks are used until the StarPlane prototype is available)
3	I	evaluation of the prototype and refinement of the architecture
	II	(i) development of control architectures for applications (ii) development/evaluation of component libraries for future applications
4	I	PhD thesis

Figure 3: Proposed timeline in years

8 Expected use of instrumentation

We expect to use the SURFnet-6 hardware infrastructure and (if accepted) the DAS-3 system. The only new equipment we need to buy for this project is an optical switch (or an extension of the DAS-3 switch) for the VU. See Section 6.4.3 for more detail.

9 Requested budget

The requested budget is shown on the right. The equipment budget is to either buy an optical switch or to extend the DAS-3 optical switch. The programmer will be shared by the UvA and VU.

	Euros
Postdoc VU (3 years)	166.407
Ph.D. student UvA (4 years)	160.029
Scientific programmer (2 years)	117.644
Benchfee postdoc	5.000
Benchfee Ph.D. student	5.000
Equipment VU	50.000
total	504.080

10 Literature

Key publications

[1] T. DeFanti, C. de Laat, J. Mambretti, K. Neggers, and B. St. Arnaud. TransLight: a global-scale LambdaGrid for e-science. In *Communications of the ACM*, 46(11):34–41, November 2003.

[2] Herbert Bos, Willem de Bruijn, Mihai Cristea, Trung Nguyen, and Georgios Portokalidis. FFPF: Fairly Fast Packet Filters. In *Proceedings of Operating System Design and Implementation (OSDI'04)*, San Francisco, CA, December 2004.

[3] J. Maassen, R. van Nieuwpoort, R. Veldema, H. Bal, T. Kielmann, C. Jacobs, and R. Hofman. Efficient Java RMI for parallel programming. In *ACM Transactions on Programming Languages and Systems*, 23(6):747–775, November 2001.

[4] K. Verstoep, R.A.F. Bhoedjang, T. Rühl, H.E. Bal, and R.F.H. Hofman. Cluster communication protocols for parallel-programming systems. In *ACM Transactions on Computer Systems*, 22(3):281–325, August 2004.

[5] Cees de Laat, Erik Radius, and Steven Wallace. The rationale of the current optical networking initiatives. In *iGrid2002 special issue, Future Generation Computer Systems*, 19(6), 2003.

Further references

[6] J. Romein, H. Bal, J. Schaeffer, and A. Plaat. A performance analysis of transposition-table-driven work scheduling in distributed search. In *IEEE Transactions on Parallel and Distributed Systems*, 13(5):447–459, May 2002.

[7] Thilo Kielmann, Philip Hatcher, Luc Bouge, and Henri E. Bal. Enabling java for high-performance computing: Exploiting distributed shared memory and remote method invocation. In *Communications of the ACM*, 44(10):110–117, 2001.

[8] Gabrielle Allen, Kelly Davis, Konstantinos N. Dolkas, Nikolaos D. Doulamis, Tom Goodale, Thilo Kielmann, André Merzky, Jarek Nabrzyski, Juliusz Pukacki, Thomas Radke, Michael Russell, Ed Seidel, John Shalf, and Ian Taylor. Enabling applications on the grid: A Gridlab Overview. In *Int. J. of High Performance Computing Applications: Special issue on Grid Computing: Infrastructure and Applications*, 17(4):449–466, 2003.

[9] Paul Barham, Boris Dragovic, Keir Fraser, Steven, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, and Andrew Warfie. Xen and the art of virtualisation. In *SOSP'03*, Bolton Landing, NY, October 2003. `www.intel-research.net/Publications/Cambridge/110720030715_178.pdf`.

[10] Kobus van der Merwe and Ian Leslie. Service specific control architecture for ATM. In *JSAC*, 16(13):424–436, April 1998.

[11] Herbert Bos. Elastic Network Control (Ph.D.thesis). *PhD thesis, Cambridge University Computer Laboratory*, Cambridge, U.K., August 1999. Also available as Technical Report TR 483.

[12] Herbert Bos and Bart Samwel. Safe kernel programming in the OKE. In *Proceedings of OPENARCH'02*, New York, USA, June 2002.

[13] David D. Clark, Craig Partridge, J. Christopher Ramming, and John T. Wroclawski. A knowledge plane for the Internet. In *Proc. of SIGCOMM'03*, Karlsruhe, Germany, August 2003.

[14] Nicodemos Damianou, Naranker Dulay, Emil Lupu, and Morris Sloman. The Ponder policy specification language. In *Lecture Notes in Computer Science*, 1995, 2001.

[15] A. Antony, J. Blom, F. Dijkstra, and C. de Laat, Experimental Studies Using Hybrid Protocols Based upon UDP to Support Applications Requiring Very High Volume Data Flows. In *FGCS special issue*, accepted for publication.

[16] Andrew S. Grimshaw, Wm. A. Wulf, and The Legion Team. The Legion vision of a worldwide virtual computer. In *Communications of the ACM*, 40(1):39–45, 1997.

[17] J. W. Romein and H. E. Bal. Solving the game of awari using parallel retrograde analysis. In *IEEE Computer*, 38(10):26–33, October 2003.

[18] M. Sloman. Policy driven management for distributed systems. In *Journal of Network and Systems Management*, **2**:333, 1994.

[19] Jeremy Sugerman Ganesh Venkitachalam and Beng-Hong Lim. Virtualizing I/O devices on VMware workstation's hosted virtual machine monitor. In *Proc. of USENIX Annual Technical Conference*, Boston, Mass., June 2001.

[20] Rob V. van Nieuwpoort and Jason Maassen and Gosia Wrzesinska and Rutger Hofman and Ceriel Jacobs and Thilo Kielmann and Henri E. Bal Ibis: a Flexible and Efficient Java-based Grid Programming Environment In Concurrency & Computation: Practice & Experience, **16**:1, pp.1–29, 2004.

[21] John Mark Agosta and Simon Crosby (Intel). Network integrity by inference in distributed systems. In *Proc. of NIPS'03*, Whistler, Canada, December 2003.

[22] Rob V. van Nieuwpoort, Thilo Kielmann, and Henri E. Bal. Efficient Load Balancing for Wide-area Divide-and-Conquer Applications. In *Proc. Eighth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP'01),*Snowbird, UT, June 18-19, 2001.

[23] Leon Gommans, Cees de Laat, Bas van Oudenaarde, Arie Taal Authorization of a QoS Path based on Generic AAA In *iGrid2002 special issue, Future Generation Computer Systems*, **19**:6, 2003

[24] Simon Crosby, Sean Rooney, Rebecca Isaacs and Herbert Bos. A perspective on how ATM lost Control. In *ACM SIGCOMM Computer Communications Review*, **32**:5, November 2002.

[25] T. Kielmann, R.F.H. Hofman, H.E. Bal, A. Plaat and R.A.F. Bhoedjang MAGPIE: MPI's Collective Communication Operations for Clustered Wide Area Systems In *Seventh ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP'99)*, Atlanta, GA, pp. 131–140, May 1999.

[26] Das-2: Distributed ASCI supercomputer. `www.cs.cu.nl/das2`, 2002.

[27] `http://www.vl-e.nl`, Virtual lab for e-science, 2004.

[28] The OptIPuter project. http://www.OptIPuter.net/index.html.

[29] The Global Lambda Integrated Facility (GLIF). http://www.glif.is/index.html.

[30] The Canadian Optical Network project CA*net4. http://www.canarie.ca/canet4/.

[31] The Common Photonic Layer technology in SURFnet6. http://www.nortelnetworks.com/products/01/cpl/.