Experimental Demonstration of Heterogeneous Cross Stratum Broker for Scientific Applications

A. Castro[(1)], A. P. Vela[(2)], Ll. Gifre[(2)], R. Proietti[(1)], C. Chen [(3)], J. Yin[(3)], X. Chen[(3)], Z. Cao[(4)], Z. Zhu[(3)], L. Velasco[(2)], and S. J. B. Yoo[(1)]

[(1)] *University of California (UC Davis), Davis, USA, albcastro@ucdavis.edu*
[(2)] *Universitat Politècnica de Catalunya (UPC), Barcelona, Spain*
[(3)] *University of Science and Technology of China (USTC), Hefei, China*
[(4)] *Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China*

Abstract: We propose and demonstrate cross-stratum Broker orchestration for scientific applications and heterogeneous resources reservation in DCs, HPC facilities and networks belonging to different operators. Experiments were performed in a distributed set-up spanning across three continents.

OCIS codes: (060.4250) Networks; (060.4251) Networks, assignment and routing algorithms.

1. Introduction

Scientific applications often require intimate interactions between theoretical analysis and experimental measurements. Nowadays, scientific experiments demand increasingly more resources, such as storage and processing, challenging not only high performance computing (HPC), but also communications networks. In a scientific experiment, sensors detect events and generate data that is collected, pre-processed, and stored. Sensors can either all be placed in the same geographical experimental facility, like in the CERN's Large Hadron Collider [1] and the IceCube Neutrino Observatory [2], or spread worldwide, such as in the Comprehensive Nuclear Test Ban Treaty Organization (CTBTO) sensor network [3]. In such scientific experiments, sensors generate large amounts of data that contains both meaningful physical measurements and noise. Thus, data filtering and pre-processing is performed before even storing and transmitting data. Because of the significant data volume generated, dedicated hardware (Hw) (e.g., FPGAs) is frequently used. The final stage consists in running complex physical models, which requires a HPC facility. Although each experiment has its own needs, in general, they follow the aforementioned stages (Fig. 1).

The experimental facility and computational resources may belong to the same organization. However, since experiments are conducted from time to time, computational resources are underutilized, which entails a high cost. In view of that, we propose and experimentally validate an architecture for scientific experiments to share computational facilities in geographically diverse locations and to provide a single entrance point to request heterogeneous connectivity (i.e., MPLS-TP, WDM, and flexgrid) and IT (i.e., storage, specialized Hw, and HPC) resources. In that regard, we extend our previous paper [4] to add, among others, heterogeneous IT brokerage.

2. Proposed architecture

Note that scientific experiments are usually carried out in two phases: *i*) data collection and pre-processing, and *ii*) model computation. Therefore, in between, pre-processed data must be stored and ready to be conveyed to the selected HPC facility. A scientific application must then be able to request specific resources to be provided immediately or to be reserved in advance. Another interesting requirement is the ability to reserve Hw resources (FPGA) located in datacenters (DC) to be loaded with the specific *bitstream* for the scientific experiment.

In consequence, we propose an architecture where scientific applications can request connectivity and IT resources to our Cross Stratum Broker, which is able to request IT resources (virtual machines, storage, and

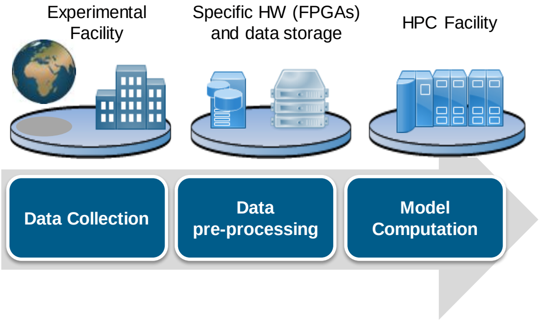


Fig. 1: Extreme-scale scientific applications model.

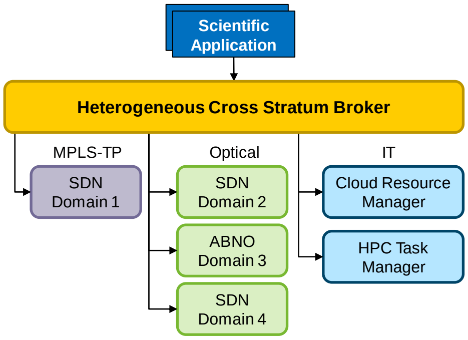


Fig. 2: Broker as a single entrance point for scientific applications.

This work was supported in part by U.S. Army Research Laboratory under Grant # W911NF-14-2-0114 and by NSF NeTS Grant # 1302719, and from the Spanish MINECO SYNERGY project TEC2014-59995-R.

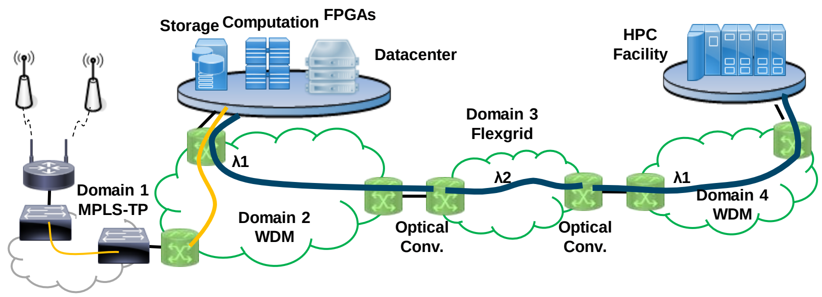


Fig. 3: Example of distributed scenario connecting data collection, DC, and HPC.

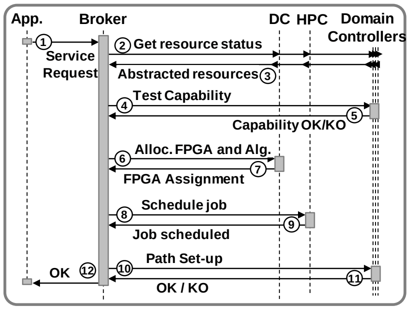


Fig. 4: Proposed workflow.

specialized Hw) to a set of DCs and computation slots at HPC facilities (Fig. 2).

Since selected IT facilities could be placed in geographically distant locations and connected to different network operators, it is clear that the broker must support heterogeneous technologies at both data and control planes. To convey collected data to DCs either MPLS-TP or optical connections can be created, depending on the data volume. The same is applicable for conveying pre-processed data from DCs to the selected HPC facility. Fig. 3 shows an example of distributed scenario connecting data collection, DC for data filtering and an HPC facility. Four domains are shown, where MPLS-TP domain 1 aggregates collected data towards optical domain 2 that transports collected data to the selected DC(s). Collected data is pre-processed in the FPGA using the specific bitstream for the scientific experiment and data is stored waiting to be sent to the HPC facility. Once all data is available and before the scheduled slot in the HPC facility, an end-to-end lightpath is set-up crossing three optical domains belonging to different network operators. Optical conversion capability at the different ingress and egress OCXs allow the broker to find a feasible end-to-end lightpath by performing conversion when no transparent spectrum can be found.

3. Scientific experiment set-up workflow

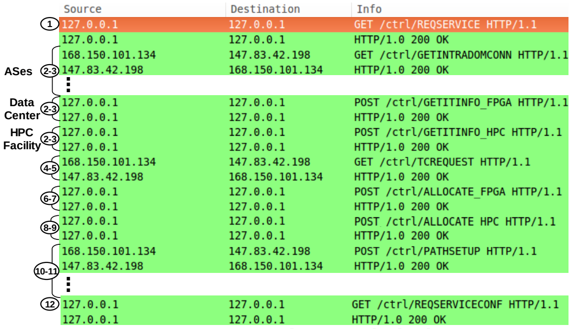
Before start accepting requests, the Broker needs to discover the available resources in every domain, i.e., storage, FPGAs, etc. in DCs, queues and priorities in HPCs, and network capabilities and inter-domain topology from each domain controller (see [4] for details); a key network capability is optical conversion at inter-domain ingress and egress OXCs. When a scientific application needs resources for an experiment, it issues a request to the Broker specifying the IT and connectivity resources required together with some temporal constraints (step 1 in Fig. 4). When the Broker receives such request, it collects the current status of the resources in every domain/facility (steps 2-3) and finds the set of resources that better fit the specific experiment needs.

In the case that some specific capability needs to be applied to release resources that are currently being used, the broker requests to apply such capability to the specific controller/manager. For instance, let us assume that the broker has found a path between the DC in domain 2 and the HPC facility in domain 4. However, no transparent wavelength assignment/spectrum allocation could be found. The broker might decide to apply the optical conversion capability in domain 3 to convert λ1 to λ2 (4-5). If a converter is available at the ingress and egress nodes in domain 3, the complete resource allocation can be performed. Therefore, the broker allocates resources in DCs (6-7), schedules jobs in HPC facilities (8-9), and establishes connections in the network domains (10-11). Finally, when the domain controllers confirm that all resources have been allocated, the Broker replies back to the scientific application informing the requested service availability and specifying the details of the job scheduling at the HPC facility to enable the scientific application to access the output results (12).

4. Experimental assessment

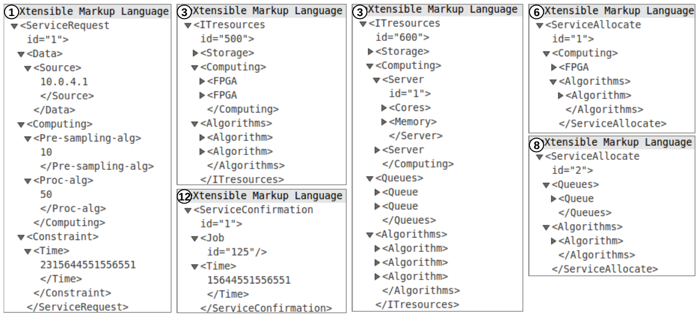
The experimental assessment has been carried out in a distributed test-bed spanning three continents. From the control plane, SDN/ABNO controllers have been deployed in Davis (USA) (domain 1 and 2), Barcelona (Spain) (domain 3), and Hefei (China) (domain 4). In contrast, the data plane, is in UCDavis labs, including data generation, OXCs with flexgrid WSSs and tunable lasers. Regarding the management plane, to enable the broker to orchestrate the experiment, we developed an HTTP REST API at the broker, which is implemented by the SDN controllers. For each API function a specific XML has been devised; these XML messages act as input/output parameters for the API functions. For the experiment, let us assume that an already established connection in domain 3 is using λ1, and the end-to-end connection from the DC in domain 2 to the HPC facility in domain 4 also needs λ1.

Fig. 5 shows the exchanged messages from the broker viewpoint. For clarity purposes, message numbering used in the workflow has been also included. Despite it is not shown in Fig. 5 for space reasons, the workflow starts with the domains, DCs and HPC facilities advertisement [4]. Once the scientific application sends its request the broker triggers the proposed workflow. Message 1 in Fig. 6 depicts the service request received by the broker. The scientific application specifies the data source, which algorithms must be used to process the data, and the time constraint for the whole experiment. Note that, the scientific application can define as many constraints as needed.



```
Source              Destination         Info
127.0.0.1           127.0.0.1           GET /ctrl/REQSERVICE HTTP/1.1
127.0.0.1           127.0.0.1           HTTP/1.0 200 OK
168.150.101.134     147.83.42.198       GET /ctrl/GETINTRADOMCONN HTTP/1.1
147.83.42.198       168.150.101.134     HTTP/1.0 200 OK
⋮
127.0.0.1           127.0.0.1           POST /ctrl/GETITINFO_FPGA HTTP/1.1
127.0.0.1           127.0.0.1           HTTP/1.0 200 OK
127.0.0.1           127.0.0.1           POST /ctrl/GETITINFO_HPC HTTP/1.1
127.0.0.1           127.0.0.1           HTTP/1.0 200 OK
168.150.101.134     147.83.42.198       GET /ctrl/TCREQUEST HTTP/1.1
147.83.42.198       168.150.101.134     HTTP/1.0 200 OK
127.0.0.1           127.0.0.1           POST /ctrl/ALLOCATE_FPGA HTTP/1.1
127.0.0.1           127.0.0.1           HTTP/1.0 200 OK
127.0.0.1           127.0.0.1           POST /ctrl/ALLOCATE HPC HTTP/1.1
127.0.0.1           127.0.0.1           HTTP/1.0 200 OK
168.150.101.134     147.83.42.198       POST /ctrl/PATHSETUP HTTP/1.1
147.83.42.198       168.150.101.134     HTTP/1.0 200 OK
⋮
127.0.0.1           127.0.0.1           GET /ctrl/REQSERVICECONF HTTP/1.1
127.0.0.1           127.0.0.1           HTTP/1.0 200 OK
```

Fig. 5: Messages Exchange at the broker.



```
Xtensible Markup Language
▼<ServiceRequest
  id="1">
 ▼<Data>
  ▼<Source>
    10.0.4.1
    </Source>
   </Data>
 ▼<Computing>
  ▼<Pre-sampling-alg>
    10
    </Pre-sampling-alg>
  ▼<Proc-alg>
    50
    </Proc-alg>
   </Computing>
 ▼<Constraint>
  ▼<Time>
    2315644551556551
    </Time>
   </Constraint>
  </ServiceRequest>
```

```
Xtensible Markup Language
▼<ITresources
  id="500">
 ▶<Storage>
 ▼<Computing>
  ▶<FPGA
  ▶<FPGA
   </Computing>
 ▼<Algorithms>
  ▶<Algorithm>
  ▶<Algorithm>
   </Algorithms>
  </ITresources>
```

```
Xtensible Markup Language
▼<ServiceConfirmation
  id="1">
 ▼<Job
   id="125"/>
 ▼<Time>
   15644551556551
   </Time>
  </ServiceConfirmation>
```

```
Xtensible Markup Language
▼<ITresources
  id="600">
 ▶<Storage>
 ▼<Computing>
  ▼<Server
    id="1">
   ▶<Cores>
   ▶<Memory>
    </Server>
  ▶<Server
   </Computing>
 ▼<Queues>
  ▶<Queue
  ▶<Queue
   </Queues>
 ▼<Algorithms>
  ▶<Algorithm>
  ▶<Algorithm>
  ▶<Algorithm>
   </Algorithms>
  </ITresources>
```

```
Xtensible Markup Language
▼<ServiceAllocate
  id="1">
 ▼<Computing>
  ▶<FPGA
  ▼<Algorithms>
   ▶<Algorithm>
    </Algorithms>
   </ServiceAllocate>
```

```
Xtensible Markup Language
▼<ServiceAllocate
  id="2">
 ▼<Queues>
  ▶<Queue
   </Queues>
 ▼<Algorithms>
  ▶<Algorithm>
   </Algorithms>
  </ServiceAllocate>
```

Fig. 6: Detail of selected XML messages.

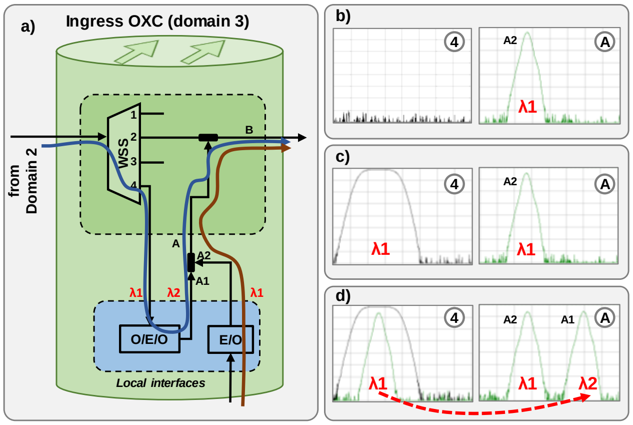


Fig. 7: Experiments at the data plane.

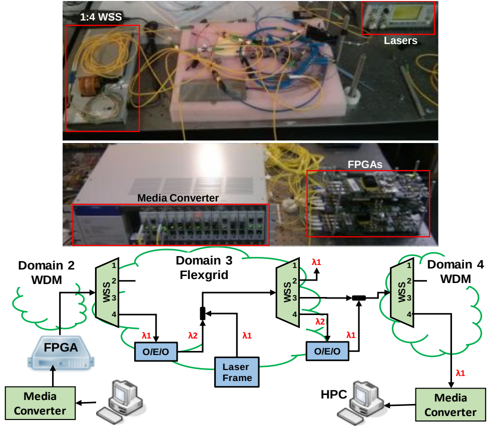


Fig. 8: Experimental deployment.

Next, the broker updates its resources database (network and IT resources). Messages 3 in Fig. 6 show the information exchange between the broker and the DC (left), and the HPC facility (right). In the case of DC, the broker gathers, e.g., the availability of the FPGAs and which algorithms can be instantiated. From the HPC facility data, the broker finds out the status of the computing queues, the available Hw resources and algorithms.

The workflow continues with the broker computing a solution, realizing that this solution implies applying a capability for conversion [4]. After the capability is confirmed, the broker allocates the IT resources. Messages 6 and 8 in Fig. 6 show the messages sent by the broker to the DC and the HPC facility, respectively. Then, the path is set up. Eventually, the broker receives all the resource allocation confirmations, it replies to the scientific application with the *job id* for the request and the expected value of the constraints (see message 12 in Fig. 6).

Fig. 7a shows the node architecture for the ingress OXC in domain 3 (a similar architecture is used in the egress OXC). A flexgrid-enabled 1:4 WSS from Finisar implements the switching element in the OXCs. Output port 4 of the WSS implements the drop port connecting with the local interfaces module (client layer). A power coupler combines the signal from WSS's output port 2 with the signals coming from the local interfaces (add ports), which are multiplexed using another power coupler. We assume wavelength tunable local transponders. Fig. 7b-d (left) show the optical spectrum at the OXC drop port (WSS port 4), whereas Fig. 7b-d (right) show the spectrum at the add port (local interfaces coupler output A). In Fig. 7b, WSS's port 4 is not configured and the already established connection occupying λ1 is shown. In Fig. 7c, the agent in the OXC has configured WSS's output 4 centered at λ1 and configured the O/E/O converter to use λ2. Finally, in Fig. 7d, the optical connection is established and the λ1 signal received in port 4 is converted into λ2 and multiplexed with the already established connection using λ1.

5. Conclusions

An architecture for sharing geographically distributed computational facilities among several scientific experiments has been proposed. A cross stratum heterogeneous Broker orchestrates resource reservation in DCs and HPC facilities and networks belonging to different operators. In particular, FPGAs available in DCs are used for data pre-processing and HPC facilities are used for computing complex scientific models. Experiments were carried out in a distributed field trial set-up connecting premises in three continents.

References

[1] CERN's Large Hadron Collider -LHC-, http://home.web. cern.ch/topics/large-hadron-collider.

[2] F. Halzen and S. Klein, "IceCube: An instrument for neutrino astronomy," Review of Scientific Instruments, vol. 81, 2010.

[3] J. Coyne et al. "CTBTO: Goals, Networks, Data Analysis and Data Availability," New Manual Seismological Obs. Pr 2 (NMSOP-2), 2012.

[4] A. Castro et al., "Experimental Demo of Brokered Orchestration for end-to-end Service Provisioning and Interoperability across Heterogeneous Multi-Operator (Multi-AS) Optical Networks," in ECOC, 2015.