AN EDGE SERVICES FRAMEWORK (ESF)* FOR EGEE∫, LCG#, AND OSG§

Abhishek Singh Rana†, Frank Würthwein‡
University of California at San Diego, CA 92093, USA

Kate Keahey, Timothy Freeman, Alexandre Vaniachine, David Malon, Ed May, Stuart Martin
Argonne National Laboratory, IL, USA

Robert Gardner, Borja Sotomayor, University of Chicago, IL, USA

Burt Holzman, Ruth Pordes, Dane Skow, Fermi National Accelerator Laboratory, IL, USA

Torre Wenaus, Razvan Popescu, Brookhaven National Laboratory, NY, USA

Jim Shank, Boston University, MA, USA

Erwin Laure, Ian Bird, Markus Schulz, Marteen Litmaath, Simone Campana, David Smith
CERN (European Organization for Nuclear Research), Geneva, Switzerland

Barry Blumenfield, Johns Hopkins University, MD, USA

Kaushik De, The University of Texas, Arlington, TX, USA

Matthew Vranicar, John Weicher, Piocon, IL, USA

Preston Smith, Purdue University, IN, USA

Shaowen Wang, University of Iowa, IA, USA

Abstract

We report on first experiences with building and operating an Edge Services Framework (ESF) based on Xen virtual machines instantiated via the Workspace Service in Globus Toolkit, and developed as a joint project between EGEE, LCG, and OSG. Many computing facilities are architected with their compute and storage clusters behind firewalls. Edge Services (ES) are instantiated on a small set of gateways to provide access to these clusters via standard grid interfaces. Experience on EGEE, LCG, and OSG has shown that at least two issues are of critical importance when designing an infrastructure in support of ES. The first concerns ES configuration. It is impractical to assume that each Virtual Organization (VO) using a facility will employ the same ES configuration, or that different configurations will coexist easily. Even within a VO, it should be possible to run different versions of the same ES simultaneously. The second issue concerns resource allocation: it is essential that an ESF be able to effectively guarantee resource allocation (e.g., memory, CPU, and networking) to different VOs. By providing virtualization on the level of instruction set architecture, virtual machines (VMs) allow configuration of independent software stacks for each VM executing on a resource. Modern implementations of this abstraction are extremely efficient and have outstanding fine-grained enforcement capabilities. To securely deploy VMs, we have extended the design of, and use, Workspace Service in the Globus Toolkit, which allows a VO administrator to dynamically launch appropriately-configured system images. In addition, we are developing a library of such images, reflecting the needs of presently participating communities ATLAS, CMS, and CDF.

In this paper, we report on first experiences designing, building and operating this Edge Services Framework.

†Corresponding Author. Abhishek Singh Rana is Senior Software Architect at UC San Diego, (Email: rana@fnal.gov, abhisheksinghrana@gmail.com).
‡Frank Würthwein is Associate Professor of Physics at UC San Diego, (Email: fkw@fnal.gov).

*ESF Activity in OSG, (URL: http://www.opensciencegrid.org/esf).
∫Enabling Grids for E-sciencE, (URL: http://public.eu-egee.org).
#LHC Computing Grid, (URL: http://lcg.web.cern.ch/LCG).
§Open Science Grid Consortium, (URL: www.opensciencegrid.org).

INTRODUCTION

Resources today are often configured so that all resources within a site are available only via a private network (LAN). The site can be accessed across WAN through a limited number of public addresses. An external client can use these to communicate with specialized services acting on its behalf within the site's private boundary. Such services are sometimes referred to as *gateways* or *gatekeepers*, e.g., Compute Element (CE), Storage Element (SE). Since these services expose their interfaces and execute at the edge of this public/private network boundary, we call them *Edge Services or ES*.

Given differing requirements of different VOs, we envision a scenario, shown in Fig. 1, in which resources will be partitioned into specific, VO-dedicated Edge Services alongside shared, open grid services such as a regular CE (GRAM) and SE (SRM) used by many VOs.

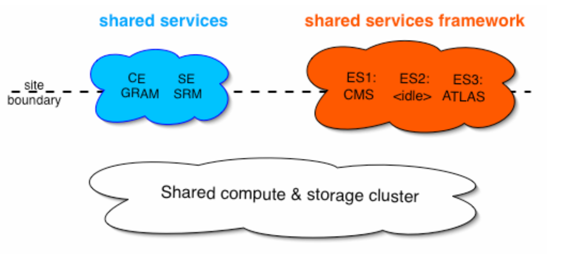

Fig. 1. A model of a typical Site on Grid in future.

Each VO may want to use different software to implement any particular kind of an Edge Service, and each VO may put different requirements on Edge Services in terms of resource usage. Thus, it is important that Edge Services have two properties:

1) There are easy ways to install and deploy them, preferably based on a pre-configured package. For example, ATLAS and CMS may decide on different VO-specific software they may want to run and neither should be obliged to use the other's configuration. The system should also allow for easy ways of upgrading this VO-specific software: if an upgrade or even a rollback is needed, it should happen seamlessly.

2) We need a way of guaranteeing dedicated use of resources allocated to the Edge Services such as memory, CPU or networking.

In this paper, we describe our recent work designing, developing, and deploying a flexible shared services infrastructure, Edge Services Framework (ESF) using emerging virtual machine (VM) technologies, which addresses the vision illustrated in Fig. 1. ESF attempts to answer the fundamental question: *On future global grids, can there be a shared services framework that is as appealing to Sites, as it is beneficial to the VOs themselves?* This implies, at a site, there is:

- No login access to strangers.
- Isolation of VO services, such that VOs cannot affect each other's operation and receive a strictly controlled environment.
- Encapsulation of services, where VO service instances can receive security reviews by site, before they get deployed.
- Ease in deployment, flexibility in dynamic resource allocation, and ease in subsequent dismantling, of VO service instances based on resource provider-consumer agreements between a site and a VO.
- Division of labor, and clear separation of concerns, between VO administrators and site administrators. VOs provision pre-configured service instances, site provision resources to host services.

AN EDGE SERVICES FRAMEWORK FOR GLOBAL GRID ENVIRONMENTS

Approaches Prior to ESF

Static deployments of VO services are common in various grids. These static models suffer from many drawbacks, a few are:

- No isolation of VO services.
- There may be a need to create proxy certificate repositories for all Users of all VOs.
- No encapsulation of services.
- An absence of ease in configuration and of flexibility in dynamic resource allocation - truly static models.
- Security hazards due to all of the above.
- Inefficient division of labor between VO and site personnel, and unregulated fixed sharing of site's resources across VOs.

We will not go into technical details of these approaches; nevertheless, there is a growing demand for more flexible and secure models of shared services. This is where ESF shows promise as a suitable alternative to previous approaches.

ESF Terminology, Architecture and Model

We have attempted to define basic technical terms in the ESF architecture, since no such components existed in previous approaches (with the exception of Workspace Service).

- Edge Services Wafer (ES Wafer). A specific instance of a dynamically-created VM (Workspace) is called an Edge Services Wafer. An ES Wafer can have several Edge Services running. A VO can have multiple ES Wafers up at a site.
- Edge Services Slot (ES Slot). An ES Slot has hardware characteristics specified by the site administrator. An ES Slot can be leased by a VO to host an ES Wafer.
- Edge Service (ES). A VO-specific service instantiated by a VO in a Wafer.
- Workspace Service (WSS). Service at a site that allows VOs to instantiate ES Wafers in ES Slots. This service is included in the standard Globus Toolkit, and has been extended as part of ESF.

Making use of the above terms, a simplified architecture of ESF is shown in Fig. 2.

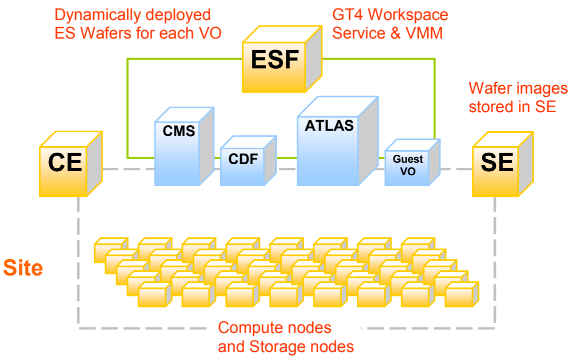

Fig. 2. Simplified ESF Architecture. A snapshot of ES Wafers implemented as Virtual Workspaces is shown.

In first stages of ES wafer development, a wafer is in form of an image file and resides with the VO administrators. Once deployed, a wafer runs inside a virtual machine, Xen VM (domainU), on a remote site, and has virtual dual-homed capabilities, i.e., it is accessible by both private and public network addresses. The following section describes this process in more detail.

EXAMPLE SCENARIOS IN ESF MODEL

Scenario 1: VO Administrator dynamically deploying an ES on a remote site.

This is outlined in Fig. 3. During site operation, Edge Service workspaces are dynamically retrieved, provisioned, and deployed by a VO administrator authorized using a VOMS credential. For example, when ATLAS analysis jobs require a database (DB) cache of a specific type, an ATLAS administrator deploys the DASH Edge Service.

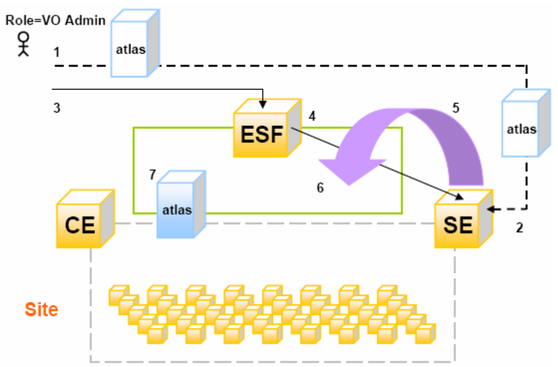

Fig. 3. A VO administrator dynamically deploying an ES Wafer as a functional Edge Service on a remote site.

Scenario 2: VO User using ES at a site.

This is outlined in Fig. 4, continuing from the above scenario. On deployment, the database cache initializes using remote data repositories over its public network connection and is then available on the private network to the jobs submitted by ATLAS users to the site.

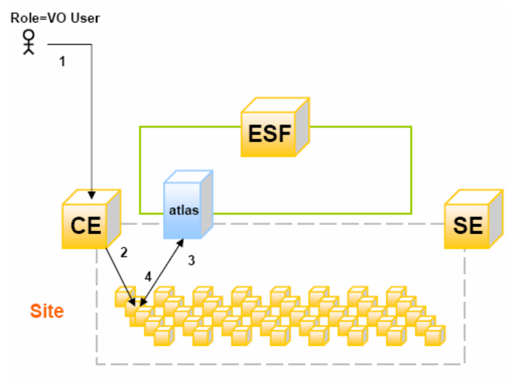

Fig. 4. A VO User using Edge Service at a remote site.

VO'S AND SITES: RESPONSIBILITIES, BENEFITS AND CONCERNS

Responsibilities of Site Administrators

The role of a site administrator is to provision hardware resources that can be used for Edge Services, ensure their proper configuration, and maintain them. A site administrator also provisions storage space in a local Storage Element (SE) for storage and retrieval of ES wafer images.

- Deploy: Xen, Custom kernel for domain0 (grub bootloader is required), Custom kernel for domainU, Prepare RAMdisk image if needed, GT4, GT4 Workspace Service.
- Provision per VM: One public IP, One private IP, Host certificates, Disk space.
- Declare available ES slots and their properties to ESF.

Responsibilities of VO Administrators

The role of a VO administrator is to prepare, configure, and test an ES wafer. The wafer image can then be shared within the VO, transported to deployment sites, and stored within the local site SE where it can be retrieved for deployment by any of the VO administrators. In the current deployment, images stored on a site are further configured with required IP addresses, and a pre-generated credential; we are working towards automating this process as part of workspace deployment.

- Fetch a standard OS filesystem image from a central ESF library.
- Deploy the desired service on OS filesystem image. Thus, prepare (freeze) ES wafer instance.
- Develop portable methods to dynamically configure (or purge) all ES networking properties at a remote site. Package these with the wafer.
- Prepare this image into file for transport, transfer the image to remote site's SE.

- Use ESF to launch a Xen VM with VO services (ES wafer) at remote site, from image file in remote SE, using role based authorization.
- Advertise the functional Edge Services as needed.

Benefits and Concerns

Our ESF deployment experiences to date are encouraging. Advantages for VOs include portability of Edge Services distributions, guaranteed use of dedicated resources at sites based on timed leases, ease of hosting various OS solutions to cater to specific sub-community needs, and an increased control to customize services configuration. Advantages for sites are flexibility in hardware provisioning (resources can be freed on expiration of Edge Services leases or reallocated dynamically), freedom from deployment of diverse sets of services resulting in an ability to support more VOs with less effort, and a relief from providing direct configuration support to VO-specific services that can be more efficiently handled by VOs themselves.

A potential disadvantage for sites is decreased control over services and interfaces exposed on the local infrastructure, therefore a consequent concern about their soundness and security. In our current infrastructure, this is addressed by vetting wafer images by site personnel. We are working on reducing the impact of such procedures by fragmentation as well as on managing trust using digitally signed images.

FIRST EXPERIENCES AND STATUS

General Status

Current ESF deployment spans integration-level testbed sites and production-level sites on OSG. Integration-level sites have Xen2.0.7 and include ANL, FNAL, University of Chicago and UCSD. Production-level deployment has Xen3.0.0 and is at the DISUN site at SDSC.

- Pure OS filesystem images built were: SL3.0.3, SL4, LTS 3, LTS 4, FC4, CentOS4.
- USCMS Edge Service: FroNTier (Squid DB).
- USATLAS Edge Service: DASH (MySQL DB).
- General Edge Service: A subset of OSG 0.4 CE.
- Preliminary stress and throughput testing was performed at ANL and UCSD.

Currently, ATLAS and CMS procure one piece of hardware at their sites on OSG, which runs ESF (called ESF node). Dual CPU and 4GB RAM (Xen2 has no PAE support, Xen3 has) is recommended. Since wafers can be large in size (5 to ~10 GB), we use regular compression and fast transport mechanisms (GridFTP) as well as high-end Storage Elements (SEs) such as dCache/SRM.

Partial list of features added to GT4 WSS (WSS Release: VM Technology Preview 1.1)

- Support for a new, *allocate* networking method that allows the WSS administrator to specify pools of IP addresses (and DNS information) which are then assigned to virtual machines on deployment.
- The resource properties have been extended to publish deployment information about a workspace, such as its IP address.
- Workspace metadata validation has been extended to support requirement checking for specific architecture, Xen version, and CPU. The workspace factory advertises supported qualities as a resource property; requirement section of workspace metadata is checked against the supported set.
- WSS can now accept and process VOMS credentials and GridShib SAML attributes.
- Support for Xen3 has been added.
- The workspace client interface has been extended to enable subscribing for notifications and specifying the resource allocation information at command-line.
- Installation has been improved. The client now requires only a minimal installation (as opposed to the full service installation).

FUTURE WORK

In near-term, we will continue to evaluate performance, functionality, robustness of Xen VM technology. We also look forward to gaining production usage experience of Edge Services running inside VMs, alongside the same services hosted on regular hardware. Based on needs and experience, we will further evolve GT4 Workspace Service design. We also plan to widen deployment to more USCMS and USATLAS sites, using CMS & ATLAS services as use cases.

Over long-term, we envision ESF will evolve and be able to perform brokering and scheduling of a cluster of ES slots with different properties.

CONCLUSION

We reported on first experiences with designing, building and operating an Edge Services Framework (ESF) developed as a joint endeavor between EGEE, LCG, and OSG.

ACKNOWLEDGEMENTS

The authors would like to express sincere gratitude to all contributing members of ESF Activity in OSG. This work was supported in part by the National Science Foundation (NSF) under Award No. 0533280; by the NSF CSR program; by the Mathematical, Information, and Computational Sciences Division subprogram of Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy (U.S. DOE); and by the U.S. DOE under Award No. DE-FC02-01ER41201.

REFERENCES

[1] http://www.opensciencegrid.org/esf
[2] http://workspace.globus.org
[3] http://www.xensource.com