System and Network Engineering Group, UvA

UNIVERSITEIT VAN AMSTERDAM

SNE

System and Network Engineering

Generic Architecture for Cloud Infrastructure as a Service (IaaS) Provisioning Model

Release 1

Yuri Demchenko, Jeroen van der Ham, Rudolf Strijkers, Mattijs Ghijsen, Canh Ngo, Mihai Cristea

15 April 2011

Abstract

This document provides information about the proposed architectural framework for Cloud Infrastructure as a Service (IaaS) provisioning model that includes the following components: the Composable Services Architecture (CSA) that intends to provide a conceptual and methodological framework for developing dynamically configurable virtualised infrastructure services; the Infrastructure Services Modeling Framework (ISMF) that provides a basis for the infrastructure resources virtualisation and management, including description, discovery, modeling, composition and monitoring; the Service Delivery Framework (SDF) that provides a basis for defining the whole composable services life cycle management and supporting infrastructure services.

The document intends to provide a starting point for further discussion and possible coordination with other project developments and standardisation activities.

This development has been facilitated by a number of the projects and activities in which the SNE Group at the University of Amsterdam takes active part, in particular projects GEYSERS, GEANT3 JRA3 Composable Services, NOVI, SURFnet e-Science, and Open Grid Forum Research Group on Infrastructure Services On-Demand provisioning (ISOD-RG).

SNE technical report SNE-UVA-2011-03

Table of Contents

1 Introduction

This document provides initial information about the development of the architectural framework for Cloud Infrastructure as a Service (IaaS) provisioning model which development is facilitated by a number of the projects in which the SNE Group at UvA takes active part, in particular GEYSERS, GEANT3 JRA3 Composable Services, NOVI, SURFnet e-Science, and Open Grid Forum Research Group on Infrastructure Services On-Demand provisioning (ISOD-RG).

The proposed architectural framework includes the following components: the Composable Services Architecture (CSA) that intends to provide a conceptual and methodological framework for developing dynamically configurable virtualised infrastructure services; the Infrastructure Services Modeling Framework (ISMF) that provides a basis for the infrastructure resources virtualisation and management, including description, discovery, modeling, composition and monitoring; the Service Delivery Framework (SDF) that provides a basis for defining the whole composable services life cycle management and supporting infrastructure services. Additional architectural components may be defined in the future revisions such as the Security Infrastructure for on-demand Infrastructure services provisioning and CSA/IaaS management framework.

The presented architecture is the result of the ongoing cooperative efforts of the two EU projects GEANT3 JRA3 Composable Services [1] and GEYSERS [2] and currently considered to be contributed to the Open Grid Forum (OGF) standardisation activity.

In cooperation with TNO the proposed architectural framework and provisioning model is intended to be applied in several use cases, such as the development of early warning systems in smart infrastructures, e-Science applications, and data centre management, to gain experience in practice.

The document intends to provide a starting point for further discussion and possible coordination of other project developments and standardisation activities.

2 Emerging Technologies in On-Demand Infrastructure Services Provisioning

2.1 Cloud computing as an emerging provisioning model for complex infrastructure services

Modern e-Science and high-technology industry require high-performance infrastructure to handle large volume of data and support complex scientific applications and technological processes. Dynamicity of projects and collaborative group environment require that such infrastructure is provisioned on demand and capable of dynamic (re-) configuration. Large amount of currently available e-Science/research infrastructures is currently available on Grid, which in case of Europe are coordinated by European Grid Initiative (EGI) [3]. Future research infrastructures will inevitably will evolve in the direction of using Cloud resources and will combine both Grid and Cloud resources.

Currently large Grid projects and Cloud Computing providers use their own dedicated network infrastructure that can handle the required data throughput but typically are over-provisioned. Their network infrastructure and security model are commonly based on the traditional VPN model that spreads worldwide, creates distributed environment for running their own services geographically distributed (like Google and Amazon), and provides localised access for users and local providers. Their service delivery business model and consequently security model are typically based and governed by a Service Level Agreement (SLA) that in general defines mutual provider and user expectations and obligations.

Most of Grid/Cloud usage scenarios for collaboration can benefit from combined computer/IT and network resources provisioning that besides improving performance can address such issues as application-centric manageability, consistency of the security services and becoming currently more important energy efficiency. The combined Grid/Cloud and network resources provisioning requires that a number of services and resource controlling systems interoperate at different stages of the whole provisioning process.

Recently, Cloud technologies [4, 5, 6] are emerging as infrastructure services for provisioning computing and storage resources, and gradually evolving into the general IT resources provisioning. Cloud Computing can be considered as natural evolution of the Grid Computing technologies to more open infrastructure-based services. Clouds “elasticity” as recognized by researchers and technology practitioners brings a positive paradigm shift in relation between the problem and the problem solving infrastructure from sizing a problem to infrastructure to sizing infrastructure to the problem.

The current Cloud services implement 3 basic provisioning models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). There are many examples of the latter two models, PaaS and SaaS, that are typically built using existing SOA and Web Services or REST technologies. However, the IaaS model, if intended to provision user or operator manageable infrastructure services, requires a new type of the service delivery and operation framework what is discussed in this paper.

This document presents the ongoing research aimed at developing an architectural framework that will address known problems in on-demand provisioning virtualised infrastructure services that may include both computing resources (computers and storage) and transport network. The solutions for pooling, virtualising and provisioning computing resources are provided by current Grid and Cloud infrastructures. New solutions should allow the combination of IT and network resources, supporting abstraction, composition and delivery for individual collaborating user groups and applications.

2.2 Use Case in e-Science

An increasing number of e-Science applications need on-demand access to large amounts of computational resources or need to tune their runtime demands. Grid technology enables resource sharing amongst various organizations and the use of large amounts of shared computational resources. Grid resource management, however, is optimized towards fair usage. The drawback is that on-demand access to Grid resources is difficult to achieve in practice, even if multiple Grids are available.

E-Science applications can be composed of a workflow of smaller applications that solve parts of a larger problem set. The execution of the workflow becomes particularly important for applications in which execution time is bound to time constraints. For example, in cardiovascular research, wave propagation models play an important role. The development of these models towards patient-specific simulations involves many model parameters based on data measured in-vivo and subject to uncertainties. One potential application of the wave propagation model is to help doctors to locate the position of the hemodialysis fistula for patients with renal failure. Because the time needed to complete wave propagation simulations vary with the state of the patient, the required computing resources to vary from one run to another. In practice, medical doctors might need the results within hours for urgent cases.

Multiple simulations executed concurrently shorten the total execution time of the wave propagation simulations. However, concurrent execution of the simulations requires a large amount of computational resources at once. Although Grid technology enables access to computing and storage resources, it does not guarantee that the computing resources will be available within a given time. In practice, the resource broker can delay a job with a large claim on the Grid resources so long that it leads to starvation. So, even though a Grid might have enough resources to execute a number of concurrent simulations, users may be better off submitting many small jobs rather than one large job.

The AMOS system proposed by authors in [7] provides support for on-demand execution of e-Science applications. It extends a workflow management system and, besides Grids, utilizes the Amazon EC2 Cloud to offload computation. Rather than extending Grid technologies to support prioritization or to support Clouds, AMOS adds an additional abstraction layer, and manages (virtual) infrastructure on behalf of the application. For example, AMOS can instantiate a transient Grid for each application or add and remove Virtual Machines (VM) to a transient Grid when applications demand more resources at run-time.

2.3 Use Case in Smart Infrastructures

In recent years, many initiatives have emerged, in which researchers collect enormous amounts of data from the environment, such as dikes, the sky or from scientific experiments, such as CERN’s LHC detector. By using Grid and Cloud infrastructures, a large amount of resources are at the disposal for such

applications, which would otherwise be technically or financially unfeasible to achieve with dedicated systems. The term Smart Infrastructures loosely defines these types of applications.

Smart Infrastructures are difficult to realize, because network and computation need to be coordinated to process the geographically dispersed data within the infrastructure constraints. On one hand, applications such as e-VLBI, only need high-speed network connections at the time of an experiment, but the required link connectivity may change while the experiment progresses. On the other hand, an early warning system for dike failure might need to redirect sensor data to intermediate nodes in the network for filtering or aggregation before feeding it to computation nodes. Because sensors cannot know on beforehand to where the data needs to be sent, the network has to be configured on beforehand or adapted at run-time. Fortunately, the with Cloud technologies such behavior can be achieved, because networks and infrastructure can be managed from the application domain.

2.4 General use cases for on-demand infrastructure services provisioning

The two basic use cases for on-demand infrastructure services provisioning can be considered: large scientific infrastructure and transport network infrastructure provisioning. These use cases represent the two different perspectives in developing infrastructure services – users and application developers perspective, on one side, and providers perspective, on the other side. Users are interested in uniform and simple access to the resource and the services that are exposed as Cloud/Grid resources and can be easily integrated into the scientific or business workflow. Infrastructure providers are interested in infrastructure resource pooling and virtualisation to simplify their on-demand provisioning and extend their service offering and business model to Virtual Infrastructure provisioning.

Figures 1 and 2 illustrate the typical e-Science infrastructure that includes Grid and Cloud based computing and storage resources, instruments, control and monitoring system, visualization system, and users represented by user clients. The diagram also reflects that there may be different types of connecting network links: high-speed and low-speed which both can be permanent for the project or provisioned on-demand.

The figure can also illustrate a typical usecase when a high performance infrastructure is used by two or more cooperative users/researcher groups in different locations. In order to fulfil their task (e.g. cooperative image processing and analysis) they require a number of resources and services to process raw data on distributed Grid or Cloud data centers, analyse intermediate data on specialist applications and finally deliver the result data to the users/scientists. This use case includes all basic components of the typical e-Science research process: data collection, initial data mining and filtering, analysis with special scientific applications, and finally presentation and visualisation to the users.

With the growing complexity and dynamicity of collaborative projects and applications, they will require access to network control and management functions to optimise their performance and resources usage. Currently, transport network, even if provided as VPN, is set up statically or can only be re-configured by an network engineer. Recently developed and successfully demonstrated new generation of the on-demand Network Provisioning Systems and Network Service Planes can provide the connectivity on demand optimized for and controlled by applications [8,32].

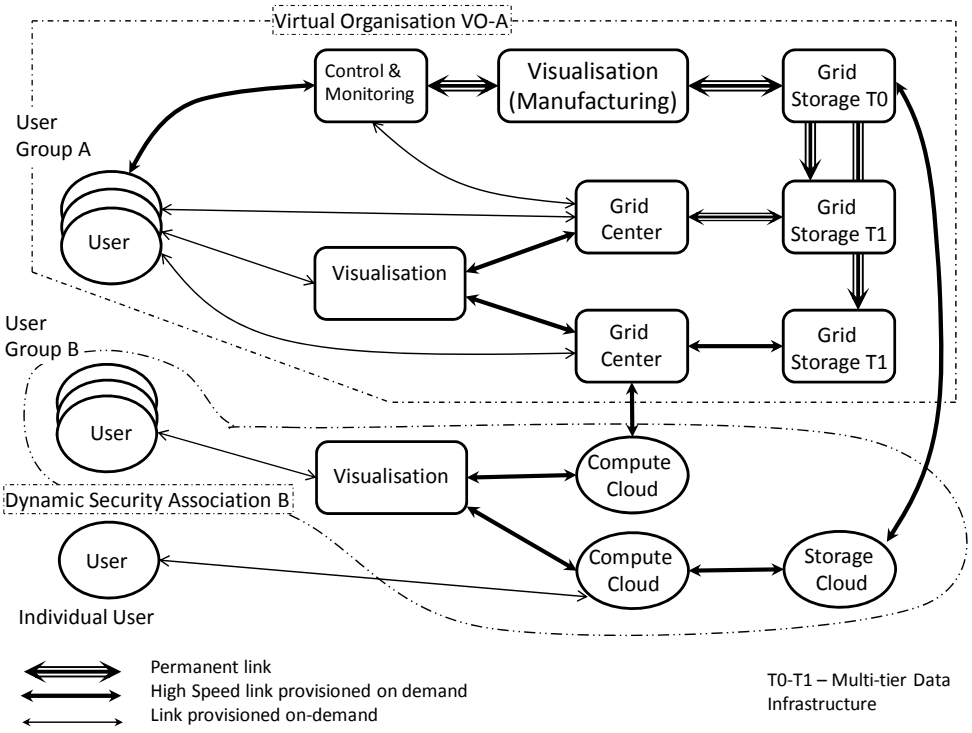

Figure 1. Components of the typical e-Science infrastructure involving multidomain and multi-tier Grid and Cloud resources and network infrastructure.

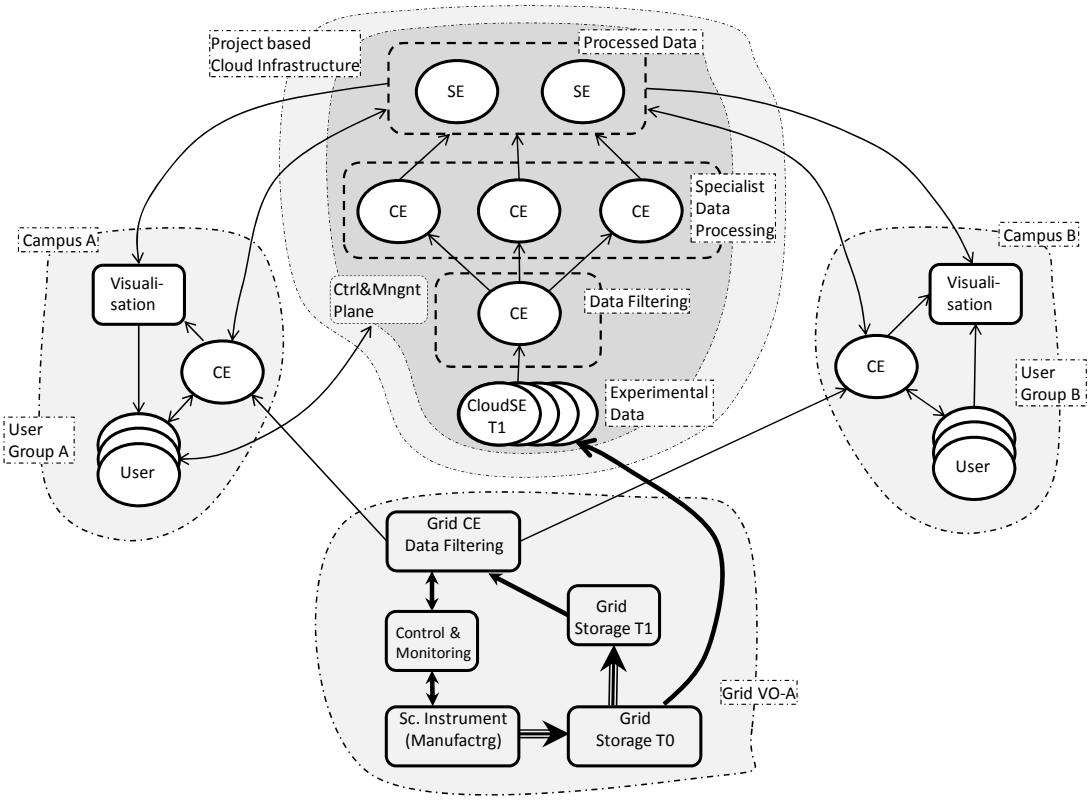

Figure 2. Project oriented collaborative infrastructure containing Grid based Scientific Instrument managed by Grid VO-A, 2 campuses A and B, and Cloud based infrastructure provisioned on-demand.

3 Abstract Model for On-demand Infrastructure Services Provisioning

Figure 3 below illustrates the abstraction of the typical project or group oriented Virtual Infrastructure (VI) provisioning process that includes both computing resources and supporting network that commonly referred as infrastructure services. The figure also shows the main actors involved into this process, such as Physical Infrastructure Provider (PIP), Virtual Infrastructure Provider (VIP), Virtual Infrastructure Operator (VIO).

The required supporting infrastructure services are depictured on the left side of the picture and includes functional components and services used to support normal operation of all mentioned actors. The Virtual Infrastructure Composition and Management (VICM) layer includes the Logical Abstraction Layer and the VI/VR Adaptation Layer facing correspondingly lower PIP and upper Application layer. VICM related functionality is described below as related to the proposed Composable Services Architecture (CSA).

The proposed architecture is a SOA (Service Oriented Architecture) [9] based and uses the same basic operation principle as known and widely used SOA frameworks, what also provides a direct mapping to the possible VICM implementation platforms such as Enterprise Services Bus (ESB) or OSGi framework [10, 11].

The infrastructure provisioning process, also referred to as Service Delivery Framework (SDF), is adopted from the TeleManagement Forum SDF [12, 13] with necessary extensions to allow dynamic services provisioning. It includes the following main stages: (1) infrastructure creation request sent to VIO or VIP that may include both required resources and network infrastructure to support distributed target user groups and/or consuming applications; (2) infrastructure planning and advance reservation; (3) infrastructure deployment including services synchronization and initiation; (4) operation stage, and (5) infrastructure decommissioning. The SDF combines in one provisioning workflow all processes that are run by different supporting systems and executed by different actors.

Physical Resources (PR), including IT resources and network, are provided by Physical Infrastructure Providers (PIP). In order to be included into VI composition and provisioning by the VIP they need to be abstracted to Logical Resource (LR) that will undergo a number of abstract transformations including possibly interactive negotiation with the PIP. The composed VI need to be deployed to the PIP which will create virtualised physical resources (VPR) that may be a part, a pool, or a combination of the resources provided by PIP.

The deployment process includes distribution of common VI context, configuration of VPR at PIP, advance reservation and scheduling, and virtualised infrastructure services synchronization and initiation, to make them available to Application layer consumers.

The proposed abstract models allows outsourcing the provisioned VI operation to the VI Operator (VIO) who is from the user/consumer point of view provides valuable services of the required resources consolidation - both IT and networks, and takes a burden of managing the provisioned services.

The proposed architecture provides a basis and motivates development of the generalised framework for provisioning dynamic security infrastructure that includes Security Services Lifecycle Management model (SSLM), common security services interface (CSSI), and related security mechanisms to allow the consistency of the dynamically provisioned security services operation. The required security infrastructure should provide a common framework for operating security services at VIP and VIO layer and be integrated with PIP’s legacy security services.

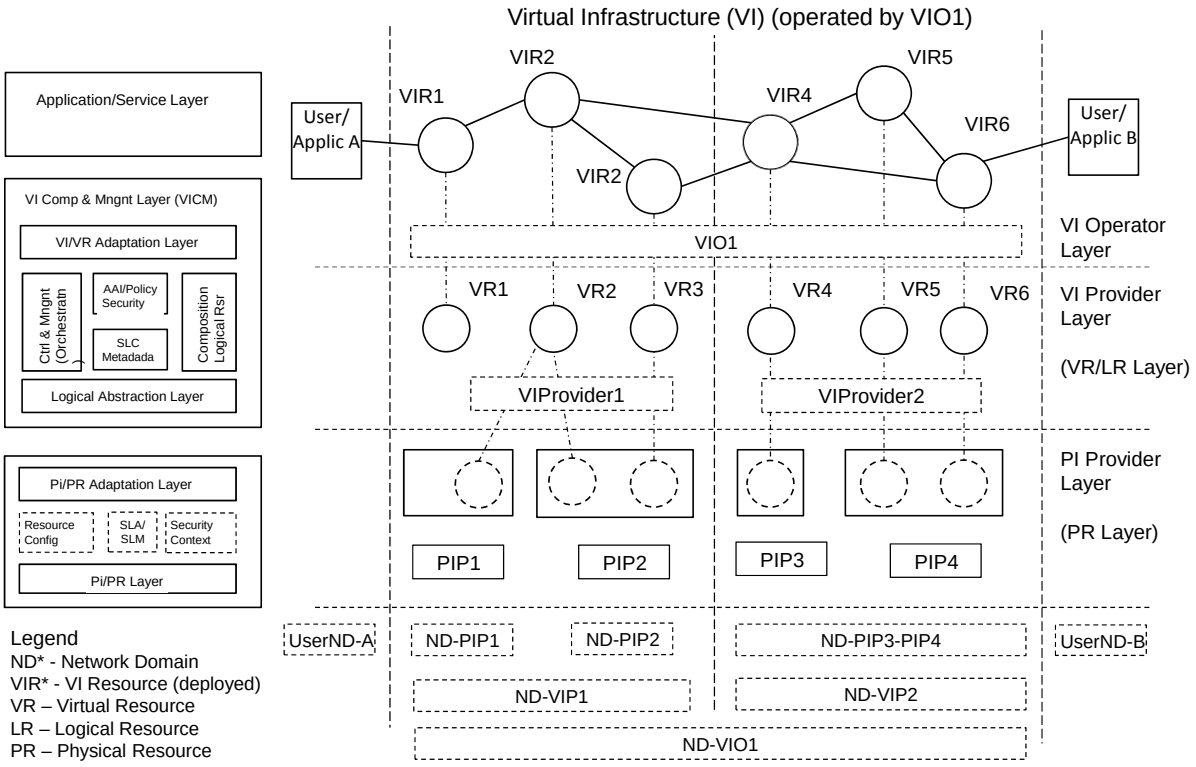

Figure 3. Main actors, functional layers and processes in on-demand infrastructure services provisioning

The proposed architecture provides a basis and motivates development of the generalised framework for provisioning dynamic security infrastructure that includes the Dynamic Access Control Infrastructure (DACI), Security Services Lifecycle Management model (SSLM), Common Security Services Interface (CSSI), and related security services and mechanisms to ensure the consistency of the dynamically provisioned security services operation. The required security infrastructure should provide a common framework for operating security services at VIP and VIO layer and be integrated with PIP's legacy security services.

Figure 4 illustrates security and trust domain related aspects in the infrastructure provisioning. It shows trust domains related to VIO, VIP and PIP that are defined by the corresponding trust anchors (TA) denoted as TA1, TA2, TA3. The user (or requestor) trust domain is denoted as TA0 to indicate that the dynamically provisioned security infrastructure is bound to the requestor's security domain. The Dynamic Security Association (DSA) is created as a part of the provisioning VI. It actually supports the VI security domain and is used to enable consistent operation of the VI security infrastructure.

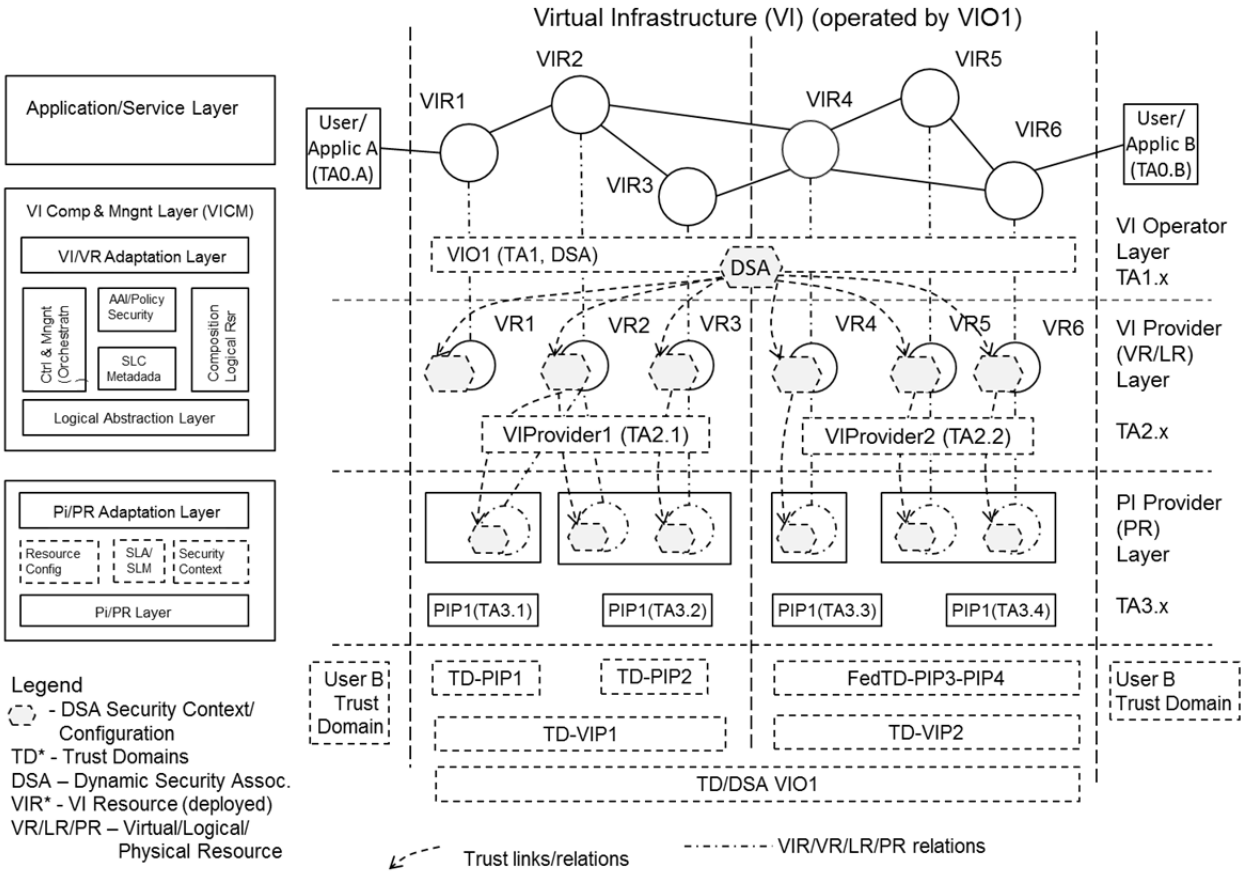

Figure 4. Dynamic Security Association (DSA) to support Security Infrastructure provisioned on-demand as a part of the overall infrastructure.

4 Infrastructure Services Modeling Framework

The Infrastructure Services Modeling Framework (ISMF) provides a basis for virtualization and management of infrastructure resources, including description, discovery, modeling, composition, and monitoring. In this paper we mainly focus on the description of resources and the lifecycle of these resources. The described model in this section is being developed in the GEYSERS project [2].

4.1 Resource Modeling

The two main descriptive elements of the ISMF are the infrastructure topology and descriptions of resources in that topology. Besides these main ingredients, the ISMF also allows for describing QoS attributes of resources, energy related attributes, and attributes needed for access control.

The main requirements for the ISMF are, that it should allow for describing Physical Resources (PR) as well as Virtual Resources (VR). Describing physical aspects of a resource means that a great level of detail in the description is required while describing a virtual resource may require a more abstract view. Furthermore, the ISMF should allow for manipulation of resource descriptions such as partitioning and aggregation. Resources on which manipulation takes place, and resources that are the outcome of manipulation are called Logical Resources (LR).

The ISMF is based on semantic web technology. This means that the description format will be based on the Web Ontology Language (OWL) [30]. This approach ensures the ISMF is extensible and allows for easy abstraction of resources by adding or omitting resource description elements. Furthermore, this

approach has enabled us to re-use the Network Description Language [31] to describe infrastructure topologies.

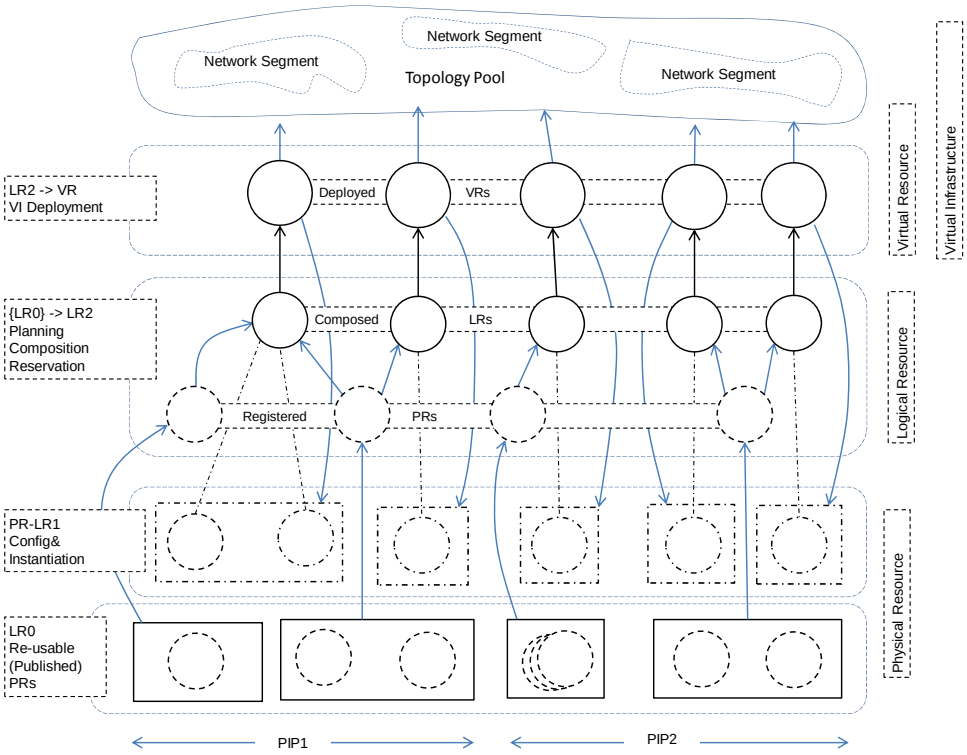

Figure 5. Relation between different resource presentations in relation to different provisioning stages (refer to Fig. 3 for the initial VI presentation).

4.2 Virtual Resource Lifecycle

Figure 5 illustrates relations between different resource presentations along the provisioning process that can also be defined as the Virtual Resource lifecycle.

The Physical Resource information is published by a PIP to the Registry service serving VICM and VIP. This published information describes a PR. The published LR information presented in the commonly adopted form (using common data or semantic model) is then used by VICM/VIP composition service to create the requested infrastructure using a combination of (instantiated) Virtual Resources and interconnecting them with a network infrastructure. In its own turn the network can be composed of a few network segments run by different network providers.

It is important to mention that physical and virtual resources discussed here are in fact complex software enabled systems with their own operating systems and security services. The VI provisioning process should support the smooth integration into the common federated VI security infrastructure by allowing the definition of a common access control policy. Access decisions made at the VI level should be trusted and validated at the PIP level. This can be achieved by creating dynamic security associations during the provisioning process.

5 The Composable Services Architecture

The Infrastructure as a Service provisioning involves dynamics creation of an infrastructure consisting of different types of resources together with necessary (infrastructure wide) control and management planes, all provisioned on-demand. The proposed CSA provides a framework for the design and operation of the

composite/complex services provisioned on-demand. It is based on the component services virtualisation, which in its own turn is based on the logical abstraction of the (physical) component services and their dynamic composition. Composite services may also use the Orchestration service provisioned as a CSA infrastructure service to operate composite service specific workflow.

5.1 Architecture Layers

The CSA adopts the general Web Services layering model to address requirement of the vertical and horizontal interoperability and integration to allow working in multidomain environment [14, 15].

The following functional layers are defined:

Networking and Transport Layer: this layer provides a possibility to apply technologies typical for distributed enterprise applications, such as VPN based network layer security and TLS/SSL based transport layer security.

Messaging Layer: this layer defines message handling functionality such as message routing, message format transformation, etc.

Virtualisation Layer (that actually consists of the Logical Abstraction Layer and the Composition and Orchestration layer): this layer provides functionality to compose services and supports their interaction (e.g. with workflows).

Application Layer: this layer represents applications, where the major goal is application related data handling and interact with the user.

Security services are applied at multiple layers to ensure consistent security. Management functions are also present at all layers and can be seen as the management plane. Control and management services related to Virtualisation Layer are defined as a part of the CSA.

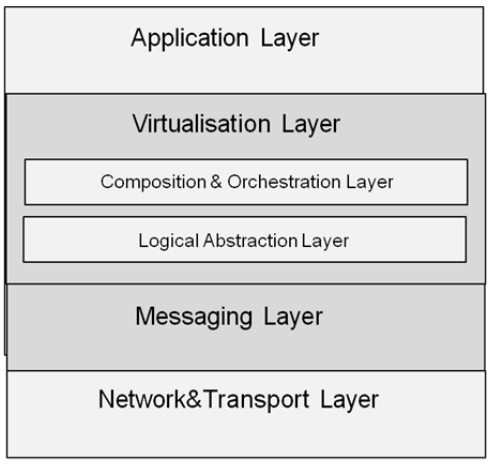

Figure 7. Composable Service Architecture Layers.

5.2 Main CSA Functional Components

Figure 8 shows that major functional components of the proposed CSA and their interaction. The central part of the architecture is the CSA middleware that should ensure smooth service operation during all stages of the composable services lifecycle.

Composable Services Middleware (CSA-MW) provides common interaction environment for both (physical) component services and complex/composite services, built with them. Besides exchanging messages, CSA-MW also contains/provides a set of basic/general infrastructure services required to support reliable and secure (composite) services delivery and operation:

- Service Lifecycle Metadata Service (MD SLC) that stores the services metadata, including the lifecycle stage, the service state, and the provisioning session context.

- Registry service that contains information about all component services and dynamically created composite services. The Registry should support automatic services registration.
- Logging service that can be also combined with the monitoring service.
- Middleware Security services that ensure secure operation of the CSA/middleware.

Note, both logging and security services can be also provided as component services that can be composed with other services in a regular way.

The CSA defines also Logical Abstraction Layer for component services and resources that is necessary part of creating services pool and virtualisation. Another functional layer is the Services Composition layer that allows presentation of the composed/composite services as regular services to the consumer.

The Control and Management plane provides necessary functionality for managing composed services during their normal operation. It may include Orchestration service to coordinate component services operation, in a simple case it may be standard workflow management system.

CSA defines a special adaptation layer to support dynamically provisioned Control and Management plane interaction with the component services which to be included into the CSA infrastructure must implement adaptation layer interfaces that are capable of supporting major CSA provisioning stages, in particular, service identification, services configuration and metadata including security context, and provisioning session management.

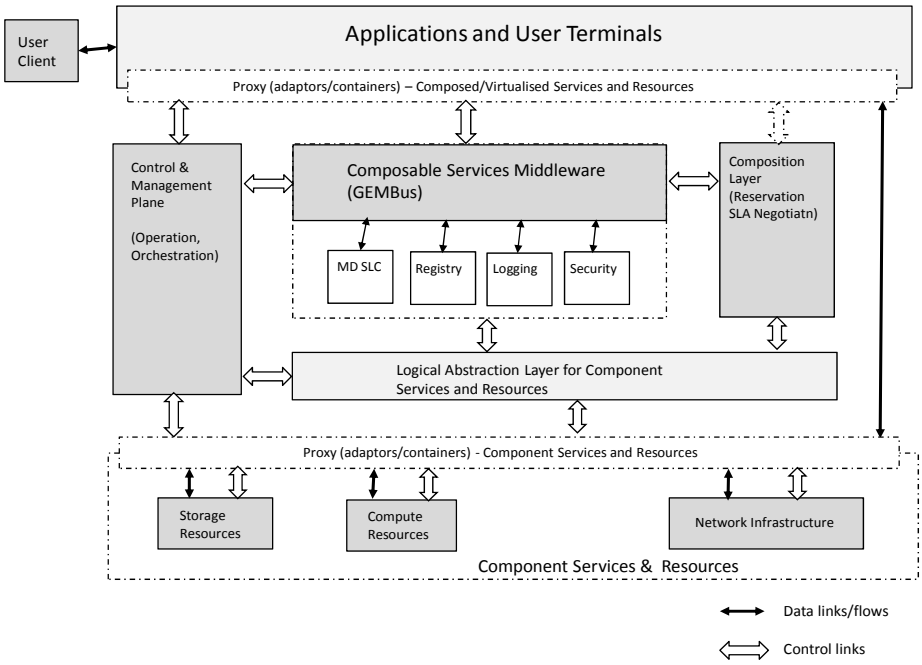

Figure 8. Composable Service Architecture and main functional components.

6 Service Delivery Framework (SDF)

Service Oriented Architecture (SOA) [9] allows for better integration between business process definition with higher abstraction description languages and dynamically composed services and provides a good basis for creating dynamically composable services that should also rely on the well-defined services lifecycle management (SLM) model. Most of existing SLM frameworks and definitions are oriented on rather traditional human-driven services development and management. Dynamically provisioned and re-configured services will require re-thinking of existing models and proposing new security mechanisms at each stage of the typical provisioning process.

The Service Delivery Framework (SDF) [12] proposed by the TeleManagement Forum (TMF) provides a common basis for defining Software Enabled Services [13] lifecycle management framework that includes both the service delivery stages and required supporting infrastructure services.

6.1 SDF Workflow

Figure 9 illustrates the main service provisioning or delivery stages:

Service Request (including SLA negotiation). The SLA can describe QoS and security requirements of the negotiated infrastructure service along with information that facilitates authentication of service requests from users. This stage also includes generation of the Global Reservation ID (GRI) that will serve as a provisioning session identifier and will bind all other stages and related security context.

Composition/Reservation that also includes **Reservation Session Binding** with GRI what provides support for complex reservation process in potentially multidomain multi-provider environment. This stage may require access control and SLA/policy enforcement.

Deployment, including services **Registration and Synchronisation**. Deployment stage begins after all component resources have been reserved and includes distribution of the common composed service context (including security context) and binding the reserved resources or services to the GRI as a common provisioning session ID. The Registration and Synchronisation stage (that however can be considered as optional) specifically targets possible scenarios with the provisioned services migration or re-planning. In a simple case, the Registration stage binds the local resource or hosting platform run-time process ID to the GRI as a provisioning session ID.

Operation (including Monitoring). This is the main operational stage of the provisioned on demand composable services. Monitoring is an important functionality of this stage to ensure service availability and secure operation, including SLA enforcement.

Decommissioning stage ensures that all sessions are terminated, data are cleaned up and session security context is recycled. Decommissioning stage can also provide information to or initiate services usage accounting.

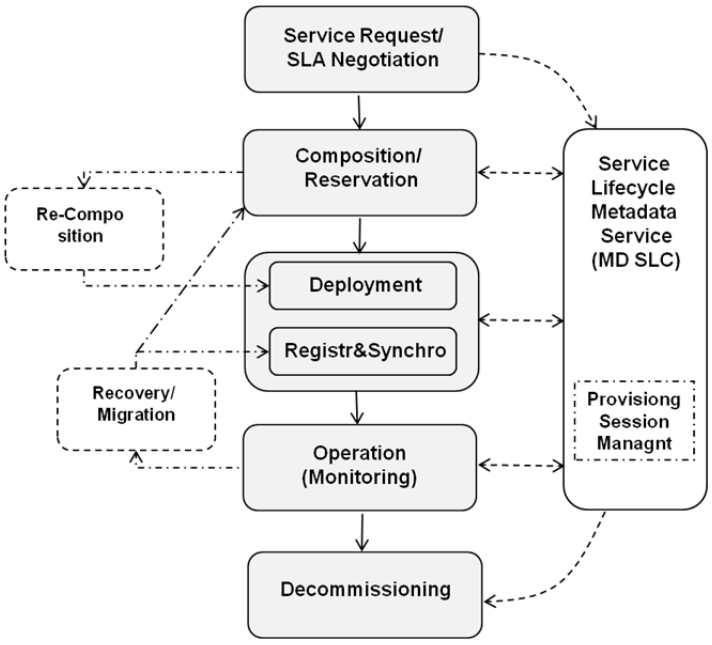

Figure 9. On-demand Composable Services Provisioning Workflow.

The two additional (sub-)stages can be initiated from the Operation stage and/or based on the running composed service or component services state, such as their availability or failure:

Re-composition or **Re-planning** that should allow incremental infrastructure changes.

Recovery/Migration can be initiated both the user and the provider. This process can use MD-SLC to initiate full or partial resources re-synchronisation, it may also require re-composition.

6.2 Infrastructure services to support SDF

Implementation of the proposed SDF requires a number of special Infrastructure Support Services (ISS) to support consistent (on-demand) provisioned services lifecycle management (similar to mentioned above TMF SDF) that can be implemented as a part of the CSA middleware.

The following services are essential to support consistent Service Lifecycle Management:

- Service Repository or Service Registry that supports services registration and discovery
- Service Lifecycle Metadata Repository (MD SLC as shown on Figure 3) that keeps the services metadata during the whole services lifecycle that include services properties, services configuration information and services state;
- Service and Resource Monitor, additional functionality that can be implemented as a part of the CSA middleware and provides information about services and resources state and usage.

7 CSA/IaaS Implementation Suggestions

7.1 GEMBus as a Framework for Enabling Composable Services

GÉANT Multi-domain service Bus (GEMBus) is being developed as a middleware for Composable Services in the framework of GÉANT3 project [1]. GEMBus incorporates the SOA services management paradigm in on-demand service provisioning. The GEMBus is built upon the industry accepted the Enterprise Service Bus (ESB) [10] and will extend it with the necessary functional components and design pattern to support multidomain services and applications.

The goal of GEMBus is to establish seamless access to the network infrastructure and the services deployed upon it, using direct collaboration between network and applications, and therefore providing more complex community-oriented services through their composition.

Figure 10 illustrates the suggested GEMBus architecture. GEMBus infrastructure includes three main groups of functionalities:

- GEMBus Messaging Infrastructure (GMI) that includes, first of all, messaging backbone and other message handling supporting services such as message routing, configuration services, secure messaging, event handler/interceptors. The GMI is built on and extends the generic ESB functionality to support dynamically configured multidomain services as defined by GEMBus.
- GEMBus infrastructure services that support reliable and secure composable services operation and the whole services provisioning process. These include such services as Composition, Orchestration, Security, and the also important Lifecycle Metadata Service, which are provided by the GEMBus environment/framework itself.
- Component services, although typically provided by independent parties, need to implement special GEMBus adaptors or use special “plug-in sockets” that allow their integration into the GEMBus/CSA infrastructure.

The following issues have been identified to enable GEMBus operation in the multidomain heterogeneous service provisioning environment:

- Service registries supporting service registration and discovery. Registries are considered as an important component to allow cross-domain heterogeneous services integration and metadata management during the whole services lifecycle.
- Security, access control, and logging should provide consistent services and security context management during the whole provisioned services lifecycle.
- Service Composition and Orchestration models and mechanisms should allow integration with the higher level scientific or business workflow.
- Messaging infrastructure should support both SOAP-based and RESTful (conforming to Representational State Transfer (REST) architecture) services [16].

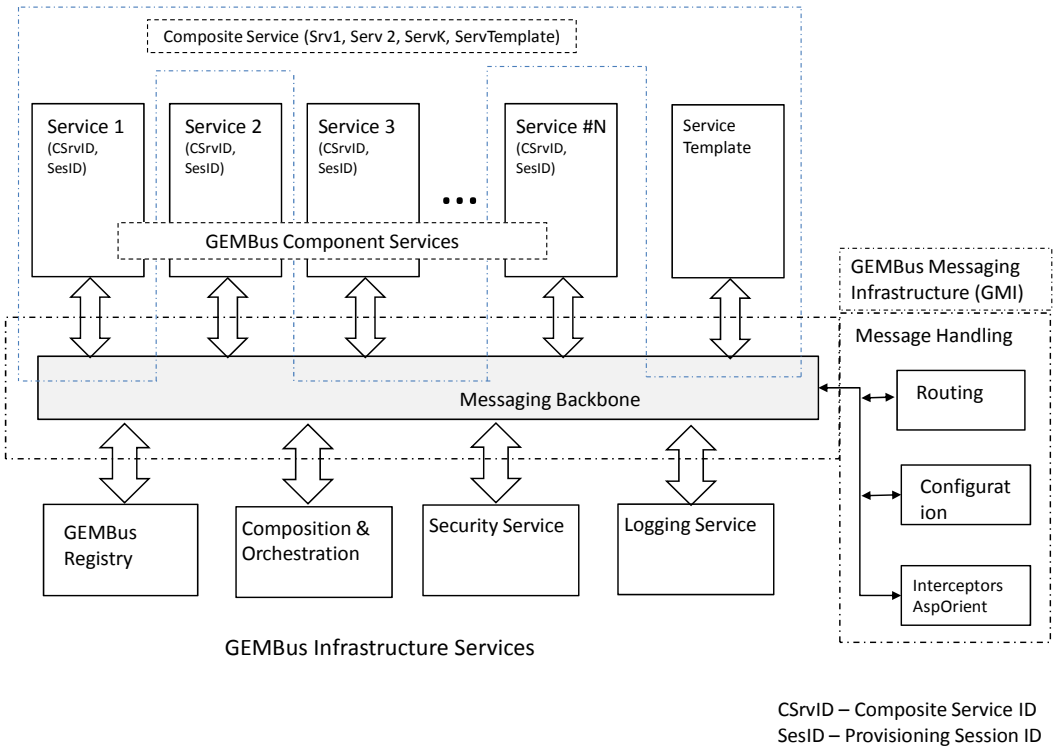

Figure 10. GEMBus infrastructure, including component services, service template, infrastructure services, and core message-processing services

The GEMBus and GMI in particular are built on the top of the standard Apache/Fuse messaging infrastructure that includes the following components [17, 18]:

- Fuse Message Broker (Apache ActiveMQ) messaging processor
- Fuse Mediation Router (Apache Camel) normalised message router

The GEMBus services and applications can be deployed on the standard Fuse or Apache ESB servers as component services that can be integrated with the standard OSGi [11] and Spring [19] compliant service development frameworks and platforms such as Fuse Services Framework/Apache CXF and Fuse ESB/Apache ServiceMix.

Figure 11 illustrates two examples of the composite services that are composed of four component services. In the second case the composite service contains a special Frontend service that is created of the corresponding service template that should be available for specific kind of applications. Examples of such services templates can be a user terminal (or rich user client), or a visualisation service. Requiring the GEMBus framework or toolkit to provide a number of typical service templates will provide more flexibility in delivery/provisioning composite services.

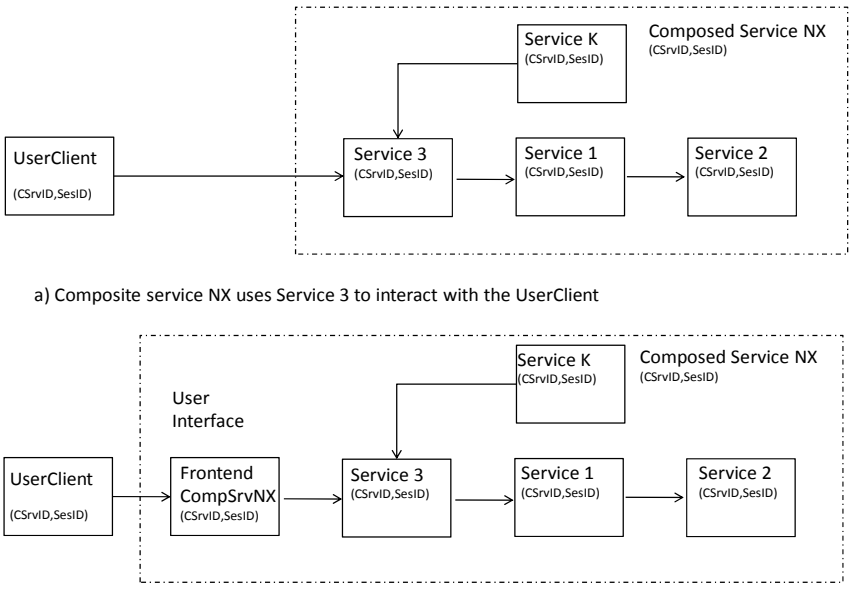

Figure 11. Example composite service composed of services Service 1, Service 2, Service 3, Service 4.

8 IaaS Security Infrastructure

8.1 General Requirements to Dynamically Provisioned Security Services

On-demand provisioning of Cloud infrastructure services drives paradigm change in security design and operation. Considering evolutional relations between Grids and Clouds, it is interesting to compare their security models. This is also important from the point of view that future e-Science infrastructures will integrate both Grid based core e-Science infrastructure and Cloud based infrastructures provisioned on-demand. Grid security architecture is primarily based on the Virtual Organisations (VO) that are created by the cooperating organisations that share resources (which however remain in their remaining in their ownership) based on mutual agreement between VO members and common VO security policy. In Grids, VO actually acts as a federation of the users and resources that enables federated access control based on the federated trust and security model [20, 21]. In general, the VO based environment is considered as trusted.

In the Clouds data are sent to and processed in the environment that is not under the user or data owner control and potentially can be compromised either Clouds insiders or by other users sharing the same resource. Data/information must be secured during all processing stages – upload, process, store, stream/visualize. Policies and security requirements must be bound to the data and there should be corresponding security mechanisms in place to enforce these policies.

The following problems/challenges arise from the Cloud IaaS environment analysis for security services/infrastructure design:

- Data protection both stored and “on-wire” that include beside the traditional confidentiality, integrity, access control services, also data lifecycle management and synchronization.
- Access control infrastructure virtualisation and dynamic provisioning, including dynamic/automated policy composition or generation.
- Security services lifecycle management, in particular service related metadata and properties, and their binding to the main services.
- Security sessions and related security context management during the whole security services lifecycle, including binding security context to the provisioning session and virtualisation platform.

- Trust and key management in provisioned on demand security infrastructure, and support of the Dynamic Security Associations (DSA) that should provide fully verifiable chain of trust from the user client/platform to the virtual resource and the virtualisation platform.
- SLA management, including initial SLA negotiation and further SLA enforcement at the planning and operation stages.

The security solutions and supporting infrastructure to support the data integrity and data processing security should provide the following functionality:

- Secure data transfer that possibly should be enforced with the data activation mechanism
- Protection of data stored on the Cloud platform
- Restore from the process failure that entails problems related to secure job/application session and data restoration.

The security solutions and supporting infrastructure should support consistent security sessions management:

- Special session for data transfer that should also support data partitioning and run-time activation and synchronization.
- Session synchronization mechanisms that should protect the integrity of the remote run-time environment.
- Secure session fail-over that should rely on the session synchronization mechanism when restoring the session.

Wider Clouds adoption by industry and their integration with advanced infrastructure services will require implementing manageable security services and mechanisms for the remote control of the Cloud operational environment integrity by users.

8.2 The Proposed Security Services Lifecycle Management Model

Most of the existing security lifecycle management frameworks, such as defined in the NIST Special Publication 800-14 “Generally Accepted Principles and Practices in Systems Security” [22], provide a good basis for security services development and management, but they still reflect the traditional approach to services and systems design driven by engineers force. The defined security services lifecycle includes the following typical phases: Initiation, Development/Acquisition, Implementation, Operation/Maintenance, and Disposal.

Figure 12 (b) illustrates the proposed Security Services Lifecycle Management (SSLM) model [23] that reflects security services operation in generically distributed multidomain environment and their binding to the provisioned services and/or infrastructure. The SSLM includes the following stages:

- **Service Request** and generation of the GRI that will serve as a provisioning session identifier (SessionID) and will bind all other stages and related security context. The Request stage may also include SLA negotiation which will become a part of the binding agreement to start on-demand service provisioning.
- **Reservation stage** and Reservation session binding that provides support for complex reservation process including required access control and policy enforcement.
- **Deployment stage** begins after all component resources have been reserved and includes distribution of the security context and binding the reserved resources or services to the Global Reservation ID (GRI) as a common provisioning session ID.
- **Registration&Synchronisation stage** (that however can be considered as optional) that specifically targets possible scenarios with the provisioned services migration or failover. In a simple case, the Registration stage binds the local resource or hosting platform run-time process ID to the GRI as a provisioning session ID.
- During **Operation stage** the security services provide access control to the provisioned services and maintain the service access or usage session.

- **Decommissioning stage** ensures that all sessions are terminated, data are cleaned up and session security context is recycled.

The proposed SSLM model extends the existing SLM frameworks and earlier proposed by authors the CRP model [3] with the new stage "Registration & Synchronisation" that specifically targets such security issues as the provisioned services/resources restoration (in the framework of the active provisioning session) and provide a mechanism for remote data protection by binding them to the session context.

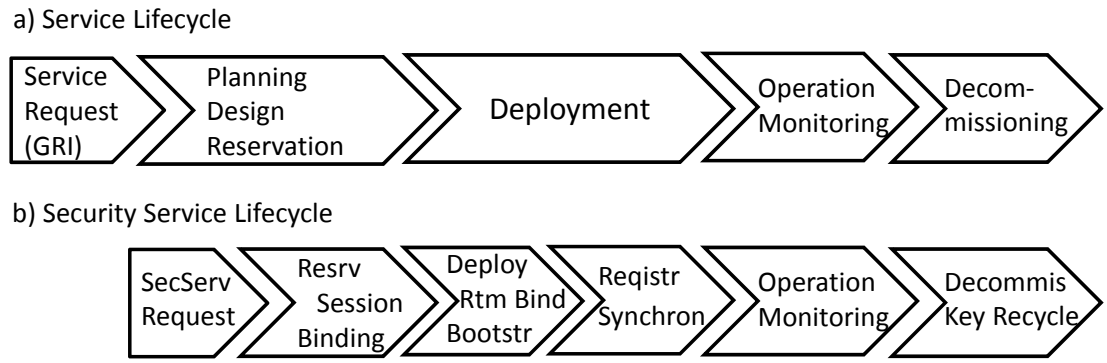

Figure 12. The proposed Security Services Lifecycle Management model.

Table A explains what main processes/actions take place during the different SLM/SSLM stages and what general and security mechanisms are used:

- SLA – used at the stage of the service Request placing and can also include SLA negotiation process.
- Workflow is typically used at the Operation stage as service Orchestration mechanism and can be originated from the design/reservation stage.
- Metadata are created and used during the whole service lifecycle and together with security services actually ensure the integrity of the SLM/SSLM.
- Dynamic security associations support the integrity of the provisioned resources and are bound to the security sessions.
- Authorisation session context supports integrity of the authorisation sessions during Reservation, Deployment and Operation stages.
- Logging can be actually used at each stage and essentially important during the last 2 stages – Operation and Decommissioning.

The proposed SSLM model extends the existing SLM frameworks with the additional stages "Reservation Session Binding" and "Registration & Synchronisation" which especially target such scenarios as the provisioned services/resources restoration, re-planning or migration (in the framework of the active provisioning session) and provide a mechanism for remote data protection by binding them to the session context. Important role in these processes belongs to the consistent security context management and dynamic security associations that should be supported by dynamic trust anchors binding and special bootstrapping procedure or protocol. However, it is perceived that implementing such functionality will require the service hosting platform that supports Trusted Computing Platform Architecture (TCPA) [24, 25].

Table A. Relation between SSLM/SLM stages and supporting general and security mechanisms

SLM stages	Request	Design/Reserv. Development	Deployment	Operation	Decomissioning
Process/ Activity	SLA Negotiation	Service/ Resource Composition Reservation	Composition Configuration	Orchestration/ Session Management	Logoff Accounting
Mechanisms/Methods					
SLA	V				V
Workflow		(V)		V	
Service Lifecycle Metadata	V	V	V	V	
Dynamic Security Associatn		(V)	V	V	
AuthZ Session Context		V	(V)	V	
Logging		(V)	(V)	V	V

9 Dynamically Provisioned Access Control Infrastructure (DACI)

Developing a consistent framework for dynamically provisioned security services requires deep analysis of all underlying processes and interactions. Many processes typically used in traditional security services infrastructure need to be abstracted, decomposed and formalized. First of all, it is related to the security services setup, configuration and security context management that in many present solutions/frameworks is provided manually, during the service installation or configured out-of-band.

The general security framework for on-demand provisioned infrastructure services should address two general aspects: (1) supporting secure operation of the provisioning infrastructure what is typically provided by the providers Authentication and Authorisation Infrastructure (AAI) supported also by the Federated Identity Management services (FIdM), and (2) provisioning a dynamic access control infrastructure (DACI) as part of the provisioned on-demand virtual infrastructure. The first task is primarily focused on the security context exchanged between involved services, resources and access control services. The virtualised DACI must be bootstrapped to the provisioned on-demand VI and VIP/VIO trust domains as entities participating in the handling initial request for VI and legally and securely bound to the VI users. Such security bootstrapping can be done at the deployment stage.

Virtual access control infrastructure setup and operation is based on the mentioned above DSA that will link the VI dynamic trust anchor(s) with the main actors and/or entities participating in the VI provisioning – VIP and the requestor or target user organisation (if they are different). As discussed above, the creation of such DSA for the given VI can be done during the reservation and deployment stage. Reservation stage will allow to distribute the initial provisioning session context and collect the security context (e.g. public key certificates) from all participating infrastructure components. The deployment stage can securely distribute either shared cryptographic keys or another type of security credentials that will allow validating information exchange and apply access control to VI users, actors, services

Figure 13 illustrates in details interaction between main actors and access control services during the reservation stage and includes also other stages of provisioned infrastructure lifecycle. The request to create

VI (RequestVI) initiates a request to VIP that will be evaluated by VIP-AAI against access control policy, what next will be followed by VIP request to PIP for required or selected physical resources PR's, which in its own turn will be evaluated by PIP-AAI. It is an SDF and SSLM requirements that starting from the initial RequestVI all communication and access control evaluations should be bound to the provisioning session identifier GRI. The chain of requests from the User to VIO, VIP and PIP can also carry corresponding trust anchors TA0…TA2, e.g. in a form of public key certificate (PKC) [23] or WS-Trust security tokens [24].

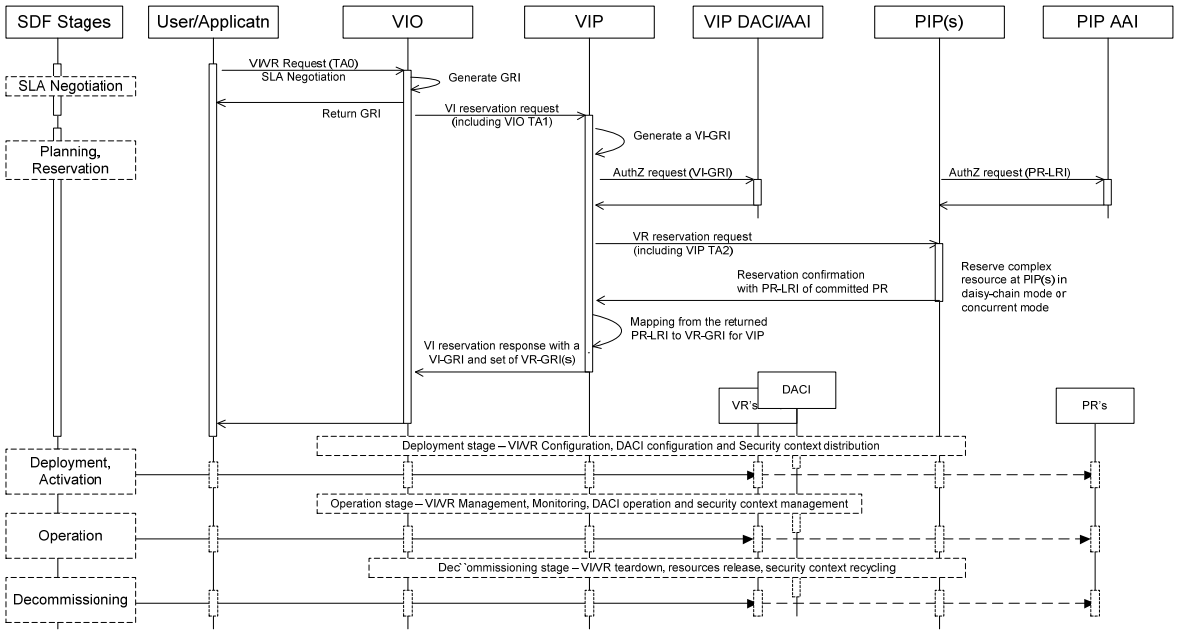

Figure 13. Security Context Management during the VI Provisioning and Operation

DACI is created at the deployment stage and controls access to and use of the VI resources, it uses dynamically created security association of the users and resources. The DACI bootstrapping can be done either by fully pre-configuring trust relations between VIP-AAI and DACI or by using special bootstrapping registration procedure similar to those used in TCPA [22].To ensure unambiguous session context and all involved entities and resources identification the following types of identifiers are used:

- Global Reservation ID (GRI) – generated at the beginning of the VI provisioning, stored at VIO and returned to User as identification of the provisioning session and the provisioned VI.
- VI-GRI – generated by VIP as an internal reservation sessions ID, which can be also re-folded GRI, depending on VIP provisioning model.
- PR-LRI and VR-LRI – provide identification of the committed or created PR@PIP and VR@VIP

9.1 Common Security Services Interfaces (CSSI)

Native to SOA and ESB [9, 10] the WS-Security framework [26] provides necessary security mechanisms and interface for virtualised resources interconnection, but their practical use in multi-domain/inter-domain virtualised environment will be complicated with the necessary namespaces and trust relations configuration at each communicating entity. The CSSI provides a simplified protocol and a Request/Response messages format what should simplify the dynamically provisioned virtualised security services integration with other infrastructure services and applications. Technically CSSI combines the core functionality of the GSS-API [27] for authentication service, GAAA-NRP authorisation service [28, 29] and adds special functionality for session management. The CSSI can be used together with WS-Security

by introducing a simplified SOAP security header structure that uses a common SecurityContext container for all security calls with the following structure:

`SecurityContext (AuthenticationData, AuthorisationData, SessionData, SecurityData)`

Such approach will allow more flexibility in defining security data format and semantic that will be actually exchanged between the virtualised services and the provider services, which due to their dynamicity will have high variability of the data structure and semantics. Instant CSSI and DACI will be configured together with provisioned VI at the deployment stage and will incorporate the provisioned infrastructure services and data semantics.

10 Summary and future developments

This document presents the ongoing research on developing architecture and framework for dynamically provisioned and reconfigurable infrastructure services to support modern e-Science and high-technology industry applications that require both high-performance computing resources (provisioned as Grids or Clouds) and high-speed dedicated transport network.

The document proposes the generalised model for provisioning infrastructure services on demand and proposes the Composable Services Architecture (CSA) that is intended to provide a conceptual and methodological framework for developing dynamically configurable virtualised infrastructure services.

The document proposes the architectural framework for on-demand infrastructure services provisioning that comprises of the 3 main components: the Composable Services Architecture (CSA) that intends to provide a conceptual and methodological framework for developing dynamically configurable virtualised infrastructure services; the Infrastructure Services Modeling Framework (ISMF) that provides a basis for the infrastructure resources virtualisation and management, including description, discovery, modeling, composition and monitoring; the Service Delivery Framework (SDF) that provides a basis for defining the whole composable services life cycle management and supporting infrastructure services.

The proposed SDF includes such additional stages such as “Registration and Synchronisation” (as part of the general services deployment process) and “Reservation Session Binding” (as part of the general services composition/reservation stage). The proposed extensions specifically target such scenarios as the provisioned resources restoration or migration/re-planning and provide a mechanism for consistent security services provisioning as an important component of the provisioned on-demand infrastructure services.

The proposed CSA is currently being implemented in the framework of the GEANT3 Project as an architectural component of the GEANT Multidomain service bus (GEMBus). The GEMBus extends the industry adopted Enterprise Service Bus (ESB) technology with the additional functionality to support multidomain services provisioning. The GEMBus infrastructure intended to allow dynamic composition of the infrastructure services to support collaboration of the distributed groups of researchers.

The future developments will include further development of all defined components and further definition of the security architecture and related security services and mechanisms to support creation of the dynamic security and trust associations.

Special attentions will be also given to the definition of the Infrastructure Services Modelling Framework, in particular, integrating related developments on defining common network and IT resources description framework.

11 References

[1] GEANT Project. [Online] Available at http://www.geant.net/pages/home.aspx

[2] Generalised Architecture for Dynamic Infrastructure Services (GEYSERS Project) [Online] Available at http://www.geysers.eu/

[3] European Grid Infrastructure (EGI). [Online] Available at https://www.egi.eu/

[4] NIST SP 800-145, "A NIST definition of cloud computing", [Online] Available: http://csrc.nist.gov/publications/drafts/ 800-145/Draft-SP-800-145_cloud-definition.pdf

[5] NIST Cloud Computing Reference Architecture, v1.0. [Online] Available at http://collaborate.nist.gov/twiki-cloud-computing/pub/CloudComputing/ReferenceArchitectureTaxonomy/NIST_CC_Reference_Architecture_v1_March_30_2011.pdf

[6] GFD.150 Using Clouds to Provide Grids Higher-Levels of Abstraction and Explicit Support for Usage Modes. [Online] Available at http://www.ogf.org/documents/GFD.150.pdf

[7] Rudolf Strijkers, Willem Toorop, Alain van Hoof, Paola Grosso, Adam Belloum, Dmitry Vasuining, Cees de Laat, Robert Meijer, *AMOS: Using the Cloud for On-Demand Execution of e-Science Applications*, in IEEE e-Science 2010 Brisbane, Australia: IEEE Computer Society, 2010

[8] Willner, A., C.Barz, J.Garcia-Espin, J.F.Riera, S.Figuerola. *Harmony: Advance reservation in heterogeneous multi-domain environment*. Proceedings of the 8th IFIP Networking conference, Springer's LNCS, 5 2009. ISBN: 978-3-642-01398-0

[9] OASIS Reference Architecture Foundation for Service Oriented Architecture 1.0, Committee Draft 2, Oct. 14, 2009. [Online] Available at http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/soa-ra-cd-02.pdf

[10] Chappell, D., *Enterprise service bus*, O'Reilly, June 2004. 247 pp.

[11] OSGi Service Platform Release 4, Version 4.2. - http://www.osgi.org/Download/Release4V42

[12] TMF Service Delivery Framework. [Online] Available at http://www.tmforum.org /servicedeliveryframework/4664/home.html

[13] TMF Software Enabled Services Management Solution. [Online] Available at http://www.tmforum.org/BestPracticesStandards/SoftwareEnabledServices/4664/Home.html

[14] Web Services Architecture. W3C Working Group Note 11 February 2004. [Online]. Available at: http://www.w3.org/TR/ws-arch/

[15] Deliverable DJ3.3.2: GEMBus Architecture. GEANT3 Project Deliverable. January, 2011.

[16] Pautasso, C., O.Zimmermann, F.Leymann, *RESTful Web Services vs. Big Web Services: Making the Right Architectural Decision*, 17th International World Wide Web Conference (WWW2008), Beijing, China.

[17] Fuse ESB - OSGi based ESB. [Online] Available at http://fusesource.com/products/enterprise-servicemix/#documentation

[18] Apache ServiceMix an Open Source ESB. [Online] Available at http://servicemix.apache.org/home.html

[19] Spring Security. Reference Documentation. [Online] Available at http://static.springsource.org/spring-security/site/ docs/3.1.x/reference/springsecurity-single.html

[20] Demchenko, Y., C. de Laat, O. Koeroo, D. Groep, *Re-thinking Grid Security Architecture*. Proceedings of IEEE Fourth eScience 2008 Conference, December 7–12, 2008, Indianapolis, USA. Pp. 79-86. IEEE Computer Society Publishing. ISBN 978-0-7695-3535-7 / ISBN 978-1-4244-3380-3.

[21] GFD.80 *The Open Grid Services Architecture, Version 1.5*. Open Grid Forum, September 5, 2006.

[22] NIST Special Publication 800-14 Generally Accepted Principles and Practices for Securing Information Technology Systems. National Institute of Standards and Technology. September 1996. [Online] Available at http://csrc.nist.gov/ publications/nistpubs/800-27/sp800-27.pdf

[23] Demchenko, Y., D.R. Lopez, J.A. Garcia Espin, C. de Laat, *Security Services Lifecycle Management in On-Demand Infrastructure Services Provisioning*, International Workshop on Cloud Privacy, Security, Risk and Trust (CPSRT 2010), 2nd IEEE International Conference on Cloud Computing Technology and Science (CloudCom2010), 30 November - 3 December 2010, Indianapolis, USA.

[24] Trusted Computing Group (TCG). [Online]. Available at https://www.trustedcomputinggroup.org/home

[25] Practical Applications of Trusted Computing in the Cloud. TCPA. [Online] Available at http://www.trustedcomputinggroup.org/files/resource_files/0B5CE066-1A4B-B294-D080E746B422584E/Practical%20Applications%20of%20Trusted%20Computing%20in%20the%20Cloud.pdf

[26] Web Services Security: SOAP Message Security 1.1 (WS-Security 2004). OASIS Standard Specification, 1 February 2006. [Online] Available at http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf

[27] RFC2853 Generic Security Service API Version 2: Java Bindings. June 2000. [Online] Available at http://www.ietf.org/rfc/rfc2853.txt

[28] GAAA Toolkit pluggable components and XACML policy profile for ONRP, Phosphorus Project Deliverable D4.3.1. September 30, 2008. [Online]. Available at http://www.ist-phosphorus.eu/files/deliverables/Phosphorus-deliverable-D4.3.1.pdf

[29] RFC 2904 AAA Authorization Framework. J. Vollbrecht, P. Calhoun, S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege, D. Spence, August 2000 - ftp://ftp.isi.edu/in-notes/rfc2904.txt

[30] OWL 2 Web Ontology Language [Online] Available at http://www.w3.org/TR/owl2-overview/

[31] J. van der Ham, F.Dijkstra, P.Grosso, R. van der Pol, A.Toonk, C. de Laat, *A distributed topology information system for optical networks based on the semantic web*, Elsevier Journal on Optical Switching and Networking, Volume 5, Issues 2-3, June 2008, pp 85-93

[32] Guok, C., D..Robertson, E.Chaniotakis, M.Thompson, W.Johnston, Brian Tierney, *A User Driven Dynamic Circuit Network Implementation*, Proceedings DANMS2008 Conference, IEEE, July 2008.

Additional References

[33] Phosphorus Project. [Online]. Available at http://www.ist-phosphorus.eu/

[34] Demchenko, Y., M. Cristea, C. de Laat, E. Haleplidis, *Authorisation Infrastructure for On-Demand Grid and Network Resource Provisioning*, Proceedings Third International ICST Conference on Networks for Grid Applications (GridNets 2009), Athens, Greece, 8-9 September 2009. ISBN: 978-963-9799-63-9

[35] GEYSERS Deliverable D2.1. Initial GEYSERS Architecture and Interfaces Specification.

[36] The Open Group Service Integration Maturity Model (OSIMM). [Online] Available at https://www.opengroup.org/projects/osimm/ uploads/40/17990/OSIMM_v0.3a.pdf

[37] TMF New Generation Operations Systems and Software (NGOSS). [Online] Available at http://www.tmforum.org/BestPracticesStandards/ SolutionFrameworks/1911/Home.html

[38] NIST Special Publication 800-27 - Engineering Principles for Information Technology Security (A Baseline for Achieving Security), June 2001. National Institute of Standards and Technology. [Online] Available at http://csrc.nist.gov/ publications/nistpubs/800-27/sp800-27.pdf

[39] ITU-T Recommendation Y.2011 (2004) - General principles and general reference model for Next Generation Networks.

[40] ITU-T Recommendation Y.2234 (09/2008) - Open service environment capabilities for NGN

[41] Securing the Cloud: Designing Security for a New Age, Dec. 10, 2009. [Online] Available at http://i.zdnet.com/whitepapers/ eflorida_ Securing_Cloud_Designing_Security_New_ Age.pdf

[42] Cloud Computing: Benefits, risks and recommendations for information security, Editors Daniele Catteddu, Giles Hogben, November 2009. [Online] Available at http://www.enisa.europa.eu/ act/rm/files/deliverables/cloud-computing-risk-assessment

[43] Amazon Web Services: Overview of Security Processes. November 2009. http://aws.amazon.com/security

[44] Security Guidance for Critical Areas of Focus in Cloud Computing V2.1. Cloud Security Alliance, December 2009. [Online] Available at http://www.cloudsecurityalliance.org/csaguide.pdf

Appendix A Existing Service Lifecycle Management Frameworks

The SOA based technologies provide a good basis for creating composable services which in case of advancing to dynamically re-configurable services should also rely on the well-defined services lifecycle management (SLM). Most of existing SLM frameworks and definitions are oriented on rather traditional human-driven services development and composition. Dynamically provisioned and re-configured services will require re-thinking of existing models and proposing new security mechanisms at each stage of the typical provisioning process.

Figure A.1 illustrates typical sequence of stages defining the provisioned service lifecycle that includes service request, design or development, deployment or implementation, operation, retire or disposal. These stages are quite intact with the proposed Complex Resource Provisioned (CRP) model which was proposed for on-demand Network Resources Provisioning in the Phosphorus project.

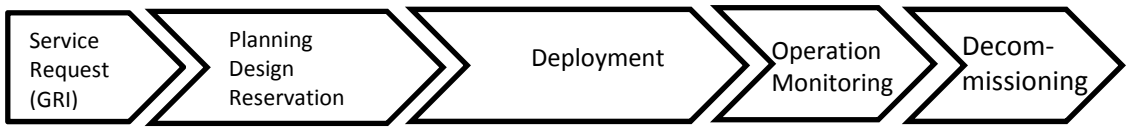

Figure A.1. General Services Lifecycle Management model

To answer dynamic character of the New Generation Networks (NGN) concept, the TeleManagement Forum (TMF) proposed the Service Delivery Framework (SDF) as a part of their New Generation Operations Systems and Software (NGOSS) solutions framework. The main motivation behind developing SDF is achieving automation of the whole service delivery and operation process, in particular:

- End-to-end service management in a multi-service providers environment
- End-to-end service management in a composite, hosted and/or syndicated service environment
- Management functions to support a highly distributed service environment, for example unified or federated security, user profile management, charging etc.
- Any other scenario that pertains to a given phase of the service lifecycle challenges, such as on boarding, provisioning, or service creation.

SDF services lifecycle corresponds to the general services lifecycle management model shown on Figure 1. Figure A.2 illustrates the major SDF components and their interactions during 3 main stages Design, Deployment, Operation. The SDF defines two basic supporting systems: Management Support Service (SDF MSS) and Infrastructure Support Service (SDF ISS). The Service instance, delivered or provisioned on demand, is presented as containing: (2) Service Management Interface, (3) Service Functional Interface, and (3a) Service Consumer Interface.

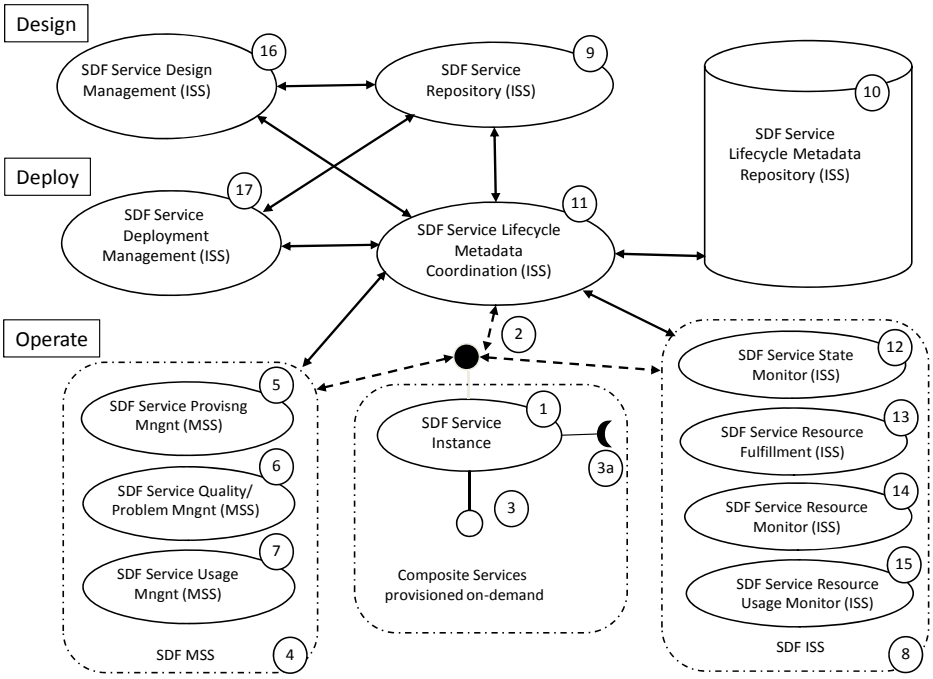

Figure A.2. SDF main components and their interaction during Design, Deployment and Operation Stages

The framework specifies the following components involved into the service provisioning at different lifecycle stages:

SDF Stages				
REQUEST	DESIGN stage	DEPLOYMENT stage	OPERATION stage	DECOMMISSION
SLA Negotiation	9 - Service Repository	10 - Service Lifecycle Metadata Repository	5 - Service Provisioning Management	10 - Service Lifecycle Metadata Repository
Provisioning Session ID assignment	10 - Service Lifecycle Metadata Repository	11 - Service Lifecycle Metadata Coordinator	6 - Service Quality/Problem Management	11 - Service Lifecycle Metadata Coordinator
11 - Service Lifecycle Metadata Coordinator	16 - Service Design Management	17 - Service Deployment Management	7 - Service Usage Monitor	14 - Service Resource Monitor
			11 - Service Lifecycle Metadata Coordinator	15 - Resource Usage Monitor
			12 - Service State Monitor	

			13 - Service Resource Fulfillment	
			14 - Service Resource Monitor	
			15 - Resource Usage Monitor	

It is important to mention that the key component of the SDF model is Services/resource Lifecycle Metadata management. This system and a process is designated to ensure the consistency of the service management and in particular important to support consistent security services provisioning and management.

Defining different lifecycle stages allows using different level of the services presentation and description at different stages and addressing different aspects and characteristics of the provisioned services. However, to ensure integrity of the service lifecycle management, the consistent services context management mechanisms should be defined and used during the whole service lifecycle. In particular case of the security services, the security services should ensure integrity of the service context management together with ensuring integrity of the security context itself. The problem here is that such mechanisms are generically stateful what impose problems for SOA environment which is defined as generically stateless.

The NIST Special Publication 800-14 “Generally Accepted Principles and Practices in Systems Security” together with SP 800-27 “Engineering Principles for Information Technology Security” [9] define a basic set of the generally accepted principles and practices for designing and managing security services. The defined security services lifecycle includes the following phases: Initiation, Development/Acquisition, Implementation, Operation/Maintenance, and Disposal. Providing a good basis for security services management, these principles still reflect the traditional approach to services and systems design driven by engineers force.