Draft Technical Standard

Service Oriented Infrastructure Reference Framework

This draft specification is made public for feedback and comment. It is not intended as a basis for implementation. It does not represent the considered view of The Open Group, and does not reflect the consensus of Open Group members.

This draft specification has not been verified for avoidance of possible third-party proprietary rights. In any use of this draft, usual procedures to ensure the respect of possible third-party intellectual property rights should be followed.

Contents

1 Introduction 7
1.1 Objective 7
1.2 Overview 7
1.3 Conformance 7
1.4 Future Directions 8

2 Foundation for Service Oriented Infrastructure 9
2.1 What is Service Oriented Infrastructure? 9
2.2 List of Definitions 9
2.3 SOI Reference Model 11
2.3.1 Business Requirements 12
2.3.2 Service-Level Requirements 12
2.3.3 BPM 12
2.3.4 IMF 12
2.3.5 Service 12
2.3.6 Infrastructure Service 12
2.3.7 Virtualized services 13
2.3.8 Physical Services 13
2.3.9 Services Exposed to and Consumed by IT 13

3 Infrastructure Services 14
3.1 Infrastructure Service consumers 14
3.2 SOI Services Publication and Exposure 16
3.3 Service Transparency 16
3.4 Composite Services 16

4 Design of Infrastructure Services 17
4.1 Infrastructure Service Characteristics 17
4.2 SOI Service Design 17
4.3 Life Cycle Management of SOI Services 20

5 SOI In Practice 21
5.1 Virtualization 21
5.1.1 Application virtualization 21
5.1.2 Database virtualization 21
5.1.3 Middleware virtualization 21
5.1.4 Server or Host Virtualization 21
5.1.5 Network Virtualization 22
5.1.6 Storage Virtualization 22
5.2 Integrated Service Management and Monitoring 22

Preface

The Open Group

The Open Group is a vendor-neutral and technology-neutral consortium, whose vision of Boundaryless Information Flow™ will enable access to integrated information within and between enterprises based on open standards and global interoperability. The Open Group works with customers, suppliers, consortia, and other standards bodies. Its role is to capture, understand, and address current and emerging requirements, establish policies, and share best practices; to facilitate interoperability, develop consensus, and evolve and integrate specifications and Open Source technologies; to offer a comprehensive set of services to enhance the operational efficiency of consortia; and to operate the industry's premier certification service, including UNIX® certification.

Further information on The Open Group is available at www.opengroup.org.

The Open Group has over 15 years' experience in developing and operating certification programs and has extensive experience developing and facilitating industry adoption of test suites used to validate conformance to an open standard or specification.

More information is available at www.opengroup.org/certification.

The Open Group publishes a wide range of technical documentation, the main part of which is focused on development of Technical and Product Standards and Guides, but which also includes white papers, technical studies, branding and testing documentation, and business titles. Full details and a catalog are available at www.opengroup.org/bookstore.

This Document

This document is the Draft Technical Standard for the Service Oriented Infrastructure Reference Framework. It has been developed by the Service-Oriented Infrastructure project of The Open Group's SOA Working Group. It is a draft that is made available for comment, and is not a formal Open Group publication. An Open Group Publication may eventually result, following comment, revision, and review. That publication will not necessarily reflect the contents of this draft in any way.

Trademarks

Boundaryless Information Flow™ and TOGAF™ are trademarks and Making Standards Work®, The Open Group®, UNIX®, and the "X" device are registered trademarks of The Open Group in the United States and other countries.

The Open Group acknowledges that there may be other brand, company, and product names used in this document that may be covered by trademark protection and advises the reader to verify them independently.

1 Introduction

1.1 Objective

The mission of The Open Group SOA Working Group is to develop and foster common understanding of SOA in order to facilitate alignment between the business and information technology communities.

It does this by conducting a work program to produce definitions, analyses, recommendations, reference models, and standards to assist business and information technology professionals within and outside of the Open Group to understand and adopt SOA.

Infrastructure Architecture is regarded by many as one of the three pillars of Information Technology, together with Business Architecture and Application Architecture. Service-oriented infrastructure results from applying the principles of service-orientation to this architectural pillar.

It is related to Service Oriented Architecture (SOA) which is most commonly used to refer to the use of this principle in Application Architecture. A Service-Oriented Infrastructure forms an appropriate foundation for a service-oriented Application Architecture (and for other styles of application architecture also), and can be regarded as a natural part of a service-oriented Enterprise Architecture.

The ultimate objective is to provide an industry-standard reference framework for service-oriented infrastructure.

The objectives of this Draft Technical Standard are to:

- Record the results of the work done by the Service-Oriented Infrastructure project to date; and
- Be a starting point for definition of an approved Open Group Technical Standard reference framework for service-oriented infrastructure.

1.2 Overview

This chapter provides a general introduction to the document. Chapter 2 describes the basic concepts of service-oriented infrastructure. Chapter 3 describes the characteristics of infrastructure services. Chapter 4 describes some design considerations for infrastructure services. Finally, Chapter 5 gives some examples of SOI in practice.

1.3 Conformance

This draft technical standard provides a reference framework that will assist architects. No concept of conformance to this framework is defined.

1.4 Future Directions

This is a draft technical standard. It is intended as a starting point for the creation of an approved Technical Standard.

2 Foundation for Service Oriented Infrastructure

2.1 What is Service Oriented Infrastructure?

Service Oriented Infrastructure or SOI is architecture for describing IT infrastructure in terms of services. The Architecture encompasses all phases of the SOI life cycle, Design, Provisioning, Operation and Decommissioning and Management of the services. It is relevant to the discovery of SOA-enabled applications, and the mapping to business processes of underlying infrastructure and IT assets. Infrastructure Services typically use and/or provide a virtualized pool of shared resources (Servers, Network, Storage, Infrastructure software) which are deployed and managed in a highly automated way. An SOI provides foundational support for a Service Oriented Application Architecture or other application architecture.

2.2 List of Definitions

This section contains a listing of the definitions used within the scope of this document. At a minimum it describes all the elements described in the Ontology Section.

Application

> An application is a collection of Business Services that address a specific domain, such as an ERP, CRM or composite applications in general.

Application Service

> An application service is consumed by end users. This service can be decomposed into a number of infrastructure services and application services.

Asset

> IT resource. In terms of the SOA Ontology, an asset is a kind of Technology Actor.

Business Service

> A business service is consumed by a business process. It is equivalent to the SOA working group's definition of "service." This service can be decomposed into a number of Infrastructure services, Application services and human services.

Composite Service

> A collection of services that are consumed as a single service.

Consumer

An entity (normally a person, organization or piece of technology, such as a software program) that uses a service. Every service has one or more consumers, and has effects that are of value to its consumers.

Infrastructure

The IT Infrastructure consists of the foundational building blocks on which applications and business processes run, it provides generic Infrastructure Services that can be used by multiple Applications.

Infrastructure Services

Infrastructure services and SOI Services are equivalent. The IT Infrastructure consists of the foundational building blocks on which applications and business processes run. They are consumed by other infrastructure services and application services. Infrastructure services can be decomposed into atomic services and other infrastructure services.

IT Repository

Information base of all IT components, services and relationships. The IT Repository is a federated view of the relevant Information Bases for this domain: the Application Management Database, the Configuration Management System (or CMDB), the Operational Management Database and the Architecture Repository. The IT Repository is a service that facilitates the full life cycle of Services: from Requirements Capturing to Solution Design, Implementation and Operation of the Services.

Physical Service

Physical services are consumed by Application Services and Infrastructure Services. Physical services are provided by the hardware components of the infrastructure

Provider

An entity (normally a person or organization) that takes responsibility for the performance of a service. Every service has a provider.

Service

A service is a logical representation of a repeatable business activity[1] that has a specified outcome (e.g., check customer credit; provide weather data, consolidate drilling reports). It is self-contained, may be composed of other services, and is a "black box" to its consumers.

Service Level

[1] This definition was developed by The Open Group's SOA Working Group as a general definition of the term "service". In the context of SOI, a service need not represent a business activity of the enterprise. Infrastructure services represent infrastructure activities that support the business activities.

A service can be viewed as a unit of work. Service level is a quantitative measurement of the time duration between the request and the consumption of the service.

Service Availability

A service can be viewed as a unit of work. Service Availability is a measurement of the percentage of time that the service is available to consumers.

Virtualization

Virtualization is a methodology in which software replicates Infrastructure Services offered by the hardware's atomic services. In this way, software that was developed to consume atomic services can be decoupled from the actual hardware components, allowing greater flexibility, mobility and agility.

Virtualized Service

A Service that appears to the software to be an atomic service but is, in fact, a service provided by the virtualization software

2.3 SOI Reference Model

Figure 2-1 shows the SOI Reference Model.

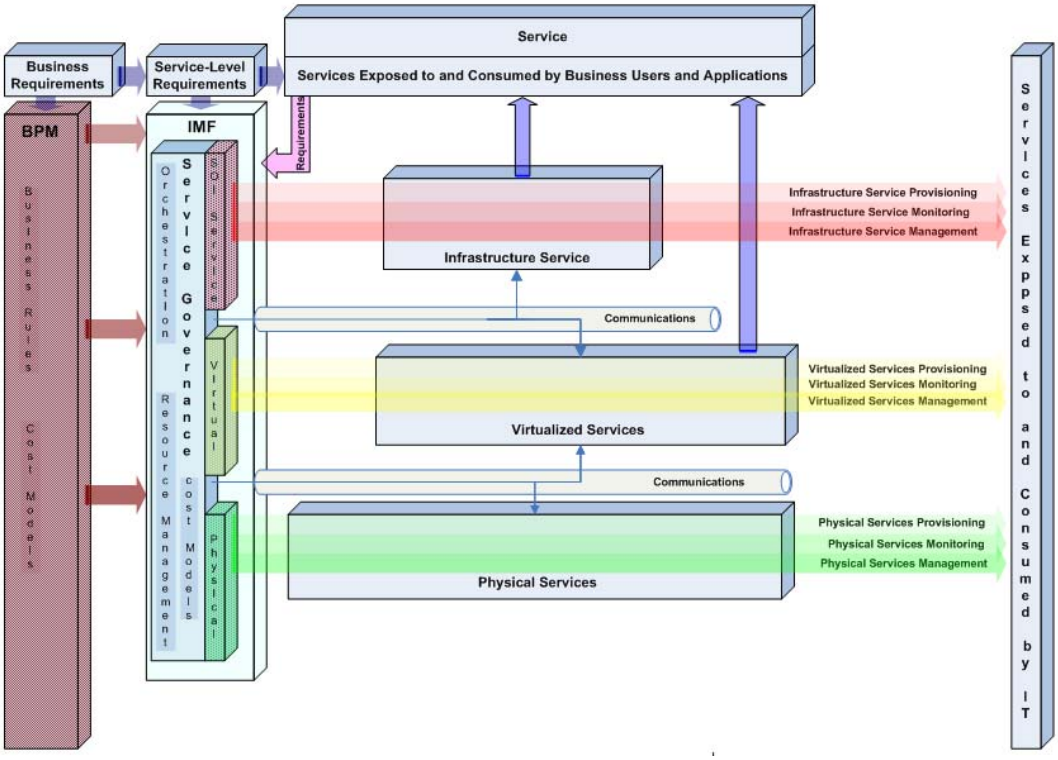


Figure 2-1: SOI Reference Model

This model shows the conceptual building blocks of SOI. These conceptual building blocks must be instantiated for each type of service or other element used in an architecture, when that architecture is implemented.

A key element of SOI as opposed to just a services oriented design is the tight coupling of the infrastructure services to applications and consumers thus capturing the full length and breadth of the IT service chain.

2.3.1 Business Requirements

In the SOI context Business Requirements are generated from both the business of the enterprise and the business of IT. The enterprise business requirements are captured in internal Service Level Agreements as well as other factors not always specified in the SLAs such as cost models. The Business of IT requirements are tied to the capability of the organization, cost metrics, vendor relationships, and strategic direction of the IT organization.

These combined requirements drive the activities of the BPM (Business Process Monitoring) and Service-Level Requirements building blocks of the reference model.

2.3.2 Service-Level Requirements

The Service-Level requirements define the levels of service to be delivered and the remedies when they are not. This includes the services delivered by the infrastructure to other services, services delivered to IT personnel consuming services and higher level Infrastructure Services consumed by business users and applications. (See section 2.2 – Service-Level and section 2.2 – service Availability)

2.3.3 BPM

Business Process Monitoring is the set of rules, cost models, and other factors that are derived from the business requirements of the enterprise and business of IT. This in turn is one of the key drivers for the IMF (Infrastructure Management Framework).

2.3.4 IMF

The Infrastructure Management Framework is the collection of COTS and home grown software, processes and procedures that are used to plan, build, and run the IT resources and services in accordance with the business and service level requirements.

2.3.5 Service

The Service building block in this model refers to the Infrastructure Services that are exposed to and consumed by the business users and business applications. It is expected that when SOI is implemented, the services contained in this box are further defined and no particular model or representation is required other then to maintain consistency with the implementer's notation, style, and syntax. (See section 2.2 – Service)

2.3.6 Infrastructure Service

The Infrastructure Services are those services that are provided by combinations of physical and virtualized services, as well as composite services made up of combinations of other

infrastructure services to provide unique capabilities that cannot be derived from atomic services.

2.3.7 Virtualized services

Virtualized Services are infrastructure services that are provided by virtualization software that is accessed by consumers using the same interface as the atomic services it is virtualizing.

2.3.8 Physical Services

The Physical Services are the atomic level of this model and represent the hardware, operating system, and personality (role based configuration) of computing, storage, and transport technologies.

2.3.9 Services Exposed to and Consumed by IT

Services that are exposed to and consumed by IT is the set of services and resources the IT organization uses to plan, build, and run the enterprise to deliver on the IT needs of the business.

3 Infrastructure Services

3.1 Infrastructure Service consumers

This section describes Infrastructure Service Types with regards to their Consumers: Application Services outside the scope of the SOI domain or Infrastructure services within the scope as part of Aggregate or Composite Services.

The key driver for the definition, development and operation of Infrastructure Services is that they are to be consumed. With respect to the target Consumers of Infrastructure Services, two distinct categories of service can be identified:

1. Infrastructure Services consumed by Application Services or Business Services
2. Infrastructure Services only consumed by other Services within the SOI domain

The key differentiator is the way these services need to be exposed to the consumers. Within the SOI domain, the Infrastructure Services need to be available to the Infrastructure developers and be accessible for the creation of Application and Business Services. Outside the SOI domain the Infrastructure Services should follow standard conventions for the Description, Discovery and Deployment of the services. E.g. for SOI Services exposed to consumers within a Service Oriented Application Architecture this could mean that the service is published using UDDI.

As an example we could build a Composite Database Service, which provides a Database Infrastructure Service to consumers outside the SOI domain (typically Application Services). The interface to the service needs to follow standards, e.g. ODBC. This provides an open interface, hiding the actual details of the service from its consumers. The ODBC service can rely on various building blocks. A possible stack is shown in Figure 3-1.

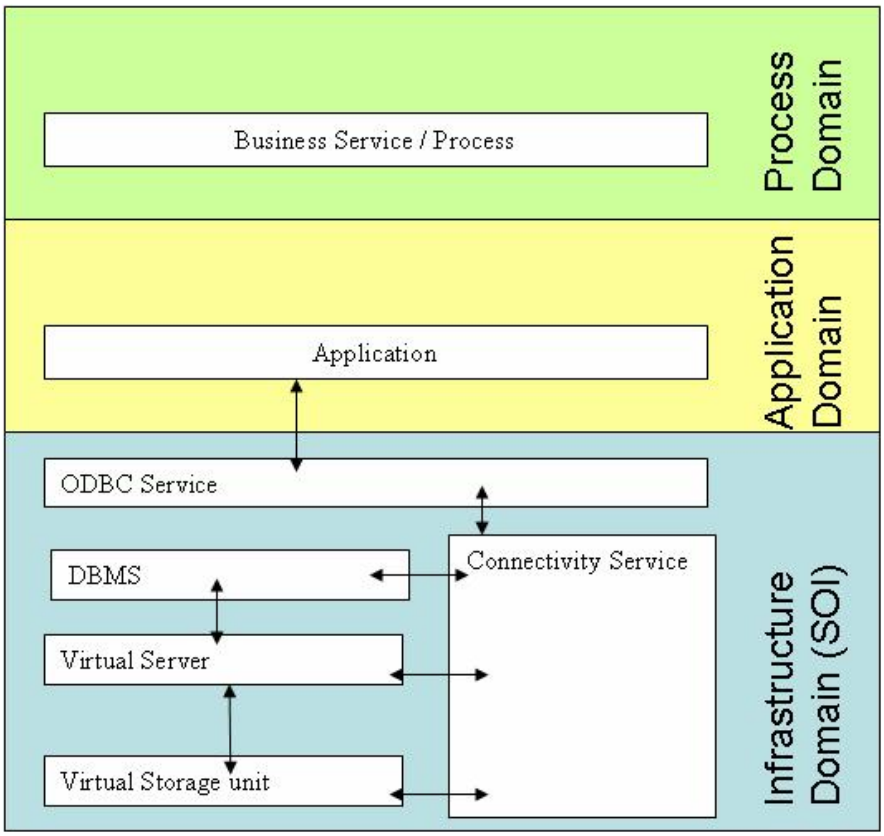


Figure 3-1: Example ODBC Service Stack

The DMBS service, the Virtual Server and the Storage Service can be deployed in other Composite Services as well. Some Services that are used in composite services can also be used as Atomic Services. In this case the Atomic Service should meet the standards for services exposed outside the SOI Domain.

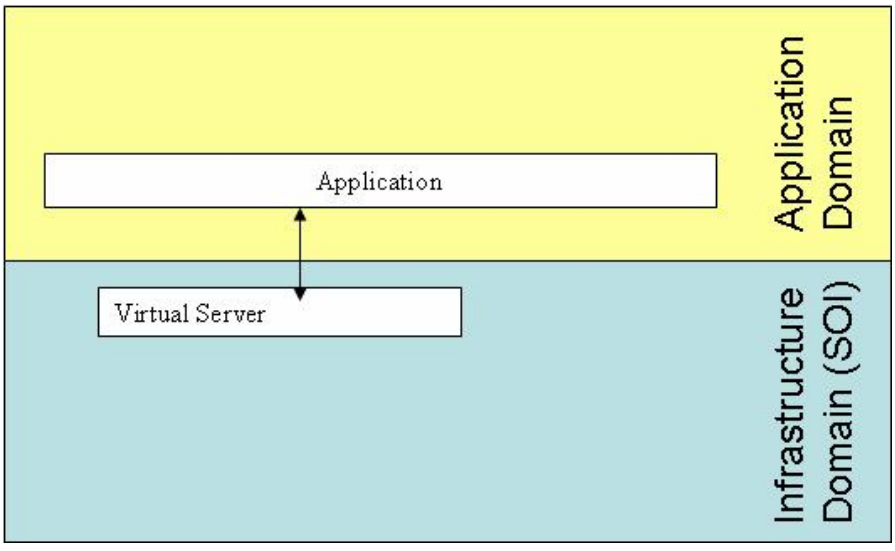


Figure 3-2: Virtual Server and Application

3.2 SOI Services Publication and Exposure

This section defines the interface between SOI and the outside world (SOA Applications or other) and probably links into the work being done within other projects of the SOA Working Group.

From an application perspective the availability of the services and their capabilities must be available from a consolidated catalog. This catalog should be part of the IT repository, a reference platform to support the entire life cycle of IT components, including the IT Infrastructure components and services.

All requirements and implementation details should be documented in the IT Repository. This repository should facilitate the discovery of services for potential consumers, in order to maximize reuse of services.

3.3 Service Transparency

Infrastructure Services can be either transparent or visible to the Service Consumers. Examples of Transparent Infrastructure Services are: Load Balancing, Encryption Services.

Transparent Services may be widespread in an organization. In line with the definition of Service within the Open Group SOA Working Group: in terms of developing an SOI Architecture it may not be relevant to expose all these services explicitly.

3.4 Composite Services

This section describes the way elementary Services are composed into aggregate or composite services.

SOI services can be standalone services, but mostly contain other SOI Services. The higher level or Composite Services use one or more Atomic Services. The composition of these services needs to be defined in an Orchestrating Function, which defines the interaction, the timing and the non-functional requirements to the SOI stack.

Besides Orchestrated SOI Services there are SOI Services that provide a service autonomously, based on a specific trigger. This autonomous behavior of Services is defined in the Choreography of SOI Services. An example of an SOI Service which provides such an autonomous service is a Service Management Agent, which based on external events, launches a certain service.

4 Design of Infrastructure Services

4.1 Infrastructure Service Characteristics

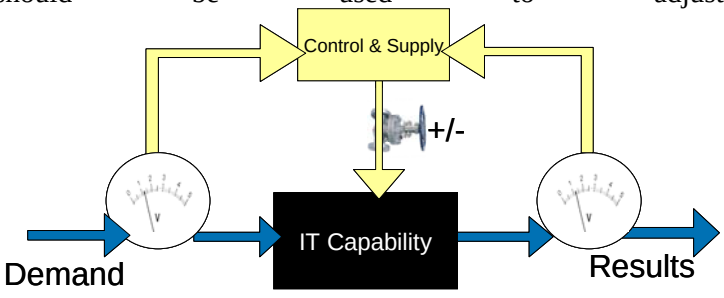
- Scalable : quantum of service used can go up or down
- Independent and modular: should be possible to be used stand alone.
- Discoverable in a network
- Configurable: should be possible to attach to a target user (application service) in real time.
- Dynamic : should be possible to switch from one user to another in real time
- Measurable : service level and service availability should be measurable
- Secure : should be possible to set levels of access based on definitions
- Location transparency
- Contingencies : what is to be done when the service is not available
- Cost: Cost of using the service in terms of resources and operational expenses
- Service should be exposed using standard based interfaces

4.2 SOI Service Design

Infrastructure services provide the application designer with a language that enables the designer to work with abstract concepts rather than implementation details.

IT Infrastructure services provide a standard response to a standard request. This definition could be extended with a classification of the bandwidth within which this statement is valid. Typically an Infrastructure Service provides its service based on the actual demand, but limited to an upper level of the service. This upper level typically is bound to the maximum capacity the service can use as provided by a resource pool supporting the service.

In order to assign the correct amount of resources to deliver the service in the right quality, the service needs to monitor the actual demand as well as the actual supply. This information should be used to adjust the service.



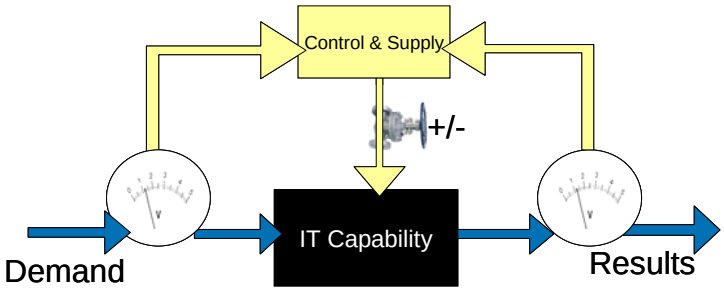


Figure 4-1: IT Infrastructure Service and the Demand-Supply Controls

Typically, Applications will require a number of Infrastructure Services in order to support their Application Interfaces. All the required Infrastructure Services need to be provided from their respective pools and assigned to the support of the specific application. The Application definition should specify a number of aspects:

- the required services or response of the service to a request;
- the initial capacity of the services
- the specific Quality Constraints of the service: or the bandwidth in which the service is able to meet its service level requirements.

Designing SOI solutions is the process of translating requirements into working solutions. With respect to the type of requirements, we can distinguish functional requirements and non-functional requirements. The functional requirements drive the capabilities within the Block Box: the (static) behavior of the IT Component on a standard request. The non-functional requirements determine the dynamics of the solution and control the provisioning of resources for the SOI Service.

New requirements typically do not lead to new physical instances within the IT Infrastructure. If an application requires a capability that is similar but not identical to an existing SOI capability there are several options:

- changing non-functional requirements (e.g. Higher/Lower Availability requirements. Higher/Lower Integrity requirements) could lead to a new SOI service, developed as a composite service using the same underlying pool of resources as the original service, but with a different Control & Supply management process.
- Changing functional requirements: this typically leads to a new service using a new pool of resources. The Control & Supply management process could be reused for the new pool of resources.

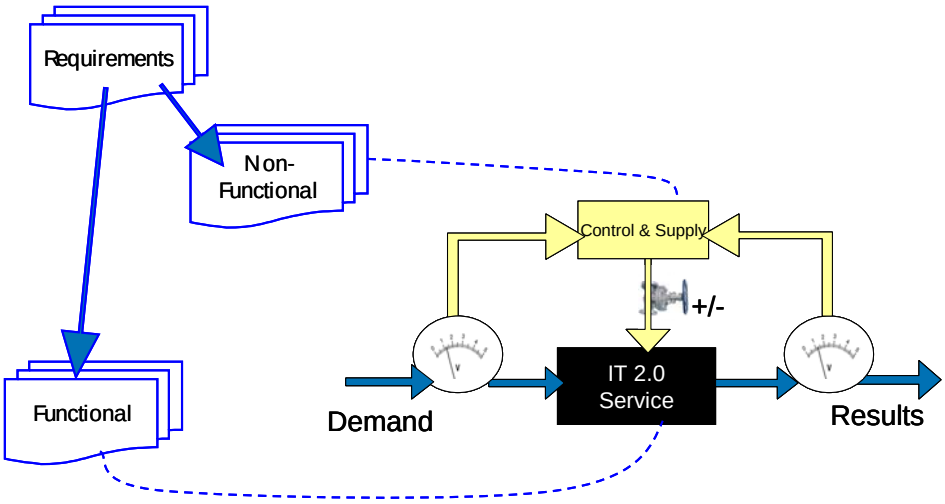


Figure 4-2: The Control and Supply Management Process

It is important to understand the distinction between a new application leading to the acquisition of new hardware and a new application leading to a better utilization of the available pool of resources.

The interface between the Application and the Infrastructure layer could include a Provisioning Service exposed by the Provisioning Engine of Service Oriented Infrastructure. The XML interface into the Provisioning Engine could accept specifications of a range of individual SOI services, e.g. for Compute resources, Database Resources, Application Server Resources, Messaging Resources, etc. The provisioning Engine will provide the resources based on the specifications received and provide a confirmation to the Application Owner once the SOI capabilities are available.

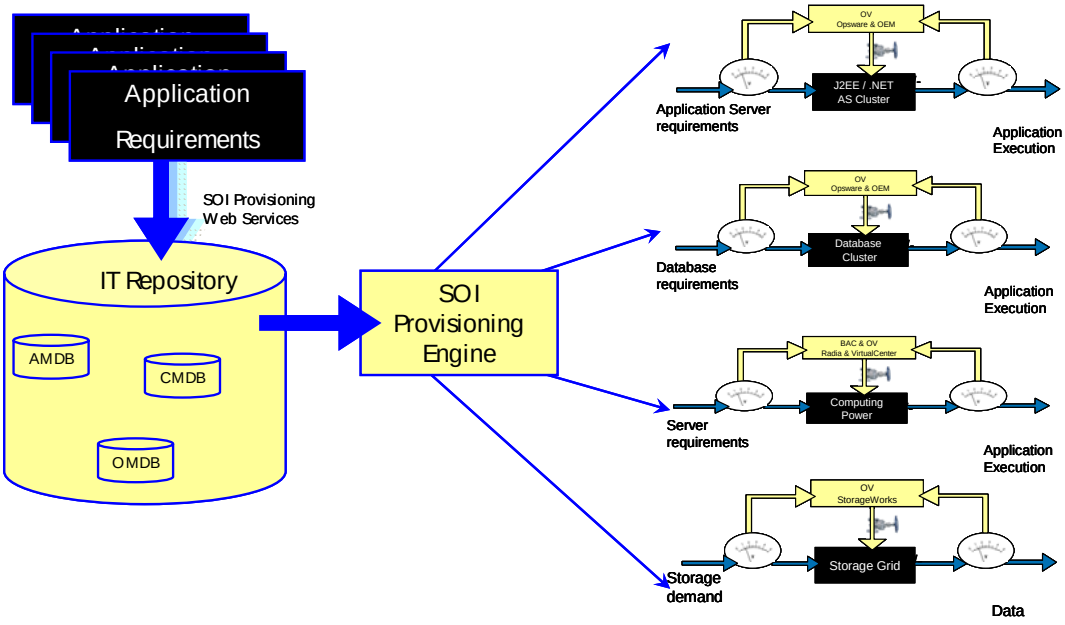


Figure 4-3: SOI Provisioning

The IT Repository is the central Information Store for the entire life cycle of all Infrastructure Services. Application Requirements are input to the repository. The process of linking Applications to SOI Services (the SOI Provisioning Process) is triggered by new or

changed Application Requirements. Changing requirements will lead to changing SOI capabilities designated for the support of these requirements.

So the Design process (new Application Requirements) impacts the SOI landscape, but this is not the only impact. The Application Requirements are also input for the feedback loop on the SOI Services. The information provided on this feedback link can also trigger a change in SOI service capacity in order to meet the requirements. This dynamic adjustment is crucial for the proper operation of a shared pool of resources as in the SOI domain.

New usage patterns therefore do not necessarily lead to new physical instances within the IT Infrastructure. Physical Instances are commissioned or decommissioned based on the Capacity Management Process, while new Application to SOI relationships are provided from the available pool of resources. Only if the initial capacity cannot be provided from the available pool, a signal will trigger an upgrade of the capacity of the pool of resources. It is important to understand the distinction between a new application leading to the acquisition of new hardware and a new application leading to a better utilization of the available pool of resources.

4.3 Life Cycle Management of SOI Services

This section describes the approach for constructing SOI Services and the approach for in life maintenance of the composite and elementary services: the Provisioning, Changing and Decommissioning of the Services.

Life Cycle Management for SOI is not different from Life Cycle Management of SOA. The SOI project will evaluate the result of the Life Cycle Management output of the SOA team to decide whether specific elements for SOI are needed.

5 SOI In Practice

This section describes the practical examples of SOI instantiation.

5.1 Virtualization

5.1.1 Application virtualization

Application virtualization aims to improve computer security and reliability by encapsulating applications from the physical hardware, operating system, and programs launched by them. When an application is launched, the virtualized software or operating system tries to contain the application by executing the code and feeding it to the system with access rules. This is different from software having direct access to the central processing unit (CPU), system hardware, etc. Through virtualization induced "containers," applications can be isolated from both the hardware and one another; thereby preventing configuration conflicts that often complicate their introduction into information technology (IT) systems.

5.1.2 Database virtualization

Database virtualization allows users to access various sources of disparately located data without knowing or caring where the data actually resides. Database virtualization allows the use of multiple instances of a DBMS, or different DBMS platforms, simultaneously and in a transparent fashion, regardless of their physical location. These practices are often employed in data mining and data warehousing systems.

5.1.3 Middleware virtualization

In the server-centric world, middleware is used to provide common infrastructure services, such as application containers, data and messaging. To ensure that the same model works in a cloud environment (i.e., a computing paradigm shift in which computing is moved away from personal computers or an individual application server to a "cloud" of computers), all those components must be virtualized. That is to say virtualization of the container, the data, and messaging. In so doing, the application is abstracted from the fact that it is running on a "cloud" and enables the transition from a server-centric model to a cloud computing model. Middleware virtualization breaks down the common infrastructure services (application containers, messaging, and data) and de-couples the APIs from the runtime platforms used to serve the API. The underlying runtime system provides common services for availability, and supports other characteristics of the service as defined in the contracts for the service (e.g., scalability, Integrity of the data, service window, etc.).

5.1.4 Server or Host Virtualization

Server or host virtualization handles the process of abstracting logical compute resources from physical compute resources. Operating system virtualizations encapsulate the entire operating system from the hardware.

5.1.5 Network Virtualization

In a virtualized network, multiple virtual networks co-exist on top of a shared substrate (Substrate refers to multiple physical networks). Different virtual networks provide alternate end-to-end packet delivery systems and may use different protocols and packet formats. Virtual networks are implemented by virtual routers connected by virtual links.

5.1.6 Storage Virtualization

Storage Virtualization refers to the process of abstracting logical storage from physical storage. The term is used to describe this abstraction at any layer in the storage software and hardware stack. Storage virtualization typically hides the complexity of the backend processes and devices from the users or servers that access the storage. Basic storage virtualization could combine two physical storage disks and make them appear as a single logical disk drive.

5.2 Integrated Service Management and Monitoring

Due to the growing complexity of distributed IT environments and increasing business dependence on technology and virtualized infrastructure, IT organizations need integrated service management processes that include technology as an interrelated component of the services IT provides to the business.

IT Service Management (ITSM) unifies service desk, incident, problem, change, asset life cycle, and service level management applications with a single configuration management database (CMDB), data model, workflow platform, and user interface. The mainstream implementation of Service Management is the Information Technology Infrastructure Library version 3. These processes facilitate Service Oriented Infrastructure by ensuring a controlled Life Cycle Management of the services as well as a management environment that ensures the proper behavior of all the SOI elements.

Similarly to the controlled lifecycle of an SOI Service, the process of managing an IT Environment as defined within ITIL V3 adopts a lifecycle approach of continuous improvement between Service Operation, Service Design and Service Transition. These ITIL V3 concepts are the functionally equivalent to the IT Service Functionality, the Feedback mechanism and the Feedforward mechanism of an SOI service. These structures are both designed to provide a controlled result for their specific domains. The ITIL processes provide the structure for an ongoing evolution of the SOI Services. An IT Service Provider could adopt SOI as a generic concept for developing IT Infrastructure services and use ITIL as a process for controlling the life cycle of the SOI services.