



Why Use OPTIMIS?

**Build and Run Services in the Most
Suitable Cloud Venues**

October 2012

Authors:

Csilla Zsigri (451 Research), Ana Juan Ferrer and Oliver Barreto (Atos), Raül Sirvent and Jordi Guitart (BSC), Srijith Nair (BT), Craig Sheridan (FLEXIANT), Karim Djemame (ULEEDS), Erik Elmroth and Johan Tordsson (UMU).

Table of Contents

1	INTRODUCTION	3
2	WHAT IS OPTIMIS?	4
2.1	OPTIMIS TARGET USERS	4
2.2	POSITIONING OF THE OPTIMIS TOOLKIT.....	4
2.3	THE OPTIMIS TOOLKIT	5
3	OPTIMIS VALUE	8
3.1	VALUE PROPOSITION	8
3.1.1	<i>Service Providers.....</i>	<i>8</i>
3.1.2	<i>Infrastructure Providers.....</i>	<i>9</i>
3.1.3	<i>Developers</i>	<i>9</i>
3.1.4	<i>End-users</i>	<i>10</i>
3.2	OPTIMIS VALUE IN ACTION	10
3.2.1	<i>A Typical Use Case</i>	<i>10</i>
3.2.2	<i>The Programming Model Use Case</i>	<i>10</i>
3.2.3	<i>The Cloud Brokerage Use Case</i>	<i>11</i>
3.2.4	<i>The Cloud-Bursting Use Case</i>	<i>12</i>
4	ADOPTING THE OPTIMIS TOOLKIT.....	13
5	TOOLKIT LAUNCH.....	13

1 Introduction

Private clouds have moved very much to the foreground through 2011, and increasingly, hybrid strategies will be defining the supply side agenda as organizations seek to make the best use of existing resources and domain skills together with the new opportunities that Cloud brings.

While the economic model remains the number one driver for building a business case to get into the Cloud, we believe that operational benefits, flexibility, agility and quicker time to market are the key drivers of ongoing adoption and value. As cloud deployments are being built out, complexity of integration with existing systems has become a key barrier for enterprise end-users, alongside security and internal resistance.

As a consequence of recent outages and performance problems, the need for multi-clouds has become more inevitable as companies seek to insure themselves against failure. At the same time, cloud offerings that enable organizations to extend their firewalls and networks directly into a hosted cloud are quickly coming to market. Altogether, these increase the requirement for 'Best Execution Venues' (BEV). End-user organizations are increasingly seeking ways to automate the delivery of workloads and applications to the most suitable cloud environments, be it internal or external, and whether that is determined by performance, risk, location and compliance or other SLA parameters.

OPTIMIS aims at optimizing IaaS cloud services by producing an architectural framework and a development toolkit. The optimization covers the full cloud service lifecycle (service construction, cloud deployment and operation). Optimis gives service providers the capability to easily orchestrate cloud services from scratch, run legacy apps on the cloud and make intelligent deployment decisions based on their preference regarding trust, risk, eco-efficiency and cost (TREC). It supports end-to-end security and compliance with data protection and green legislation. It also gives service providers the choice of developing once and deploying services across all types of cloud environments – private, hybrid, federated or multi-clouds.



Figure 1 - OPTIMIS Toolkit Features

OPTIMIS simplifies the management of infrastructures by automating most processes while retaining control over the decision-making. The various management features of the Optimis toolkit make infrastructures adaptable, reliable and scalable. These, altogether, lead to an efficient and optimized use of resources.

By using the OPTIMIS toolkit, organizations can easily provision on multi-cloud and federated cloud infrastructures and allows them to optimize the use of resources from multiple providers in a transparent, interoperable, and architecture-independent fashion.

2 What is OPTIMIS?

2.1 OPTIMIS Target Users

The OPTIMIS Toolkit comprises a set of tools to be used by Service Providers (SPs), Infrastructure Providers (IPs), Software Developers (SDs) and End-users to build and run cloud services in 'Best Execution Venues'. A definition for each target user is provided below:

- **Infrastructure Provider (IP):** An organization that provides pooled computing and storage services, including outsourced servers, OS, storage, datacenter space, network equipment, etc. to end-users or other service providers. For Infrastructure Providers, the Toolkit provides tools for managing services in compliance with service level agreements with SPs while using the infrastructure according to the IP's own policies regarding trust, risk, eco-efficiency and cost. The IP may use only its own infrastructure or subcontract resources from other IPs in a federation.
- **Service Provider (SP):** An organization that provides some kind of service or application to end-users or other service providers. It can be a Telco Provider, an Internet Service Provider, an Application Service Provider, or a Managed Service Provider, among others. For Service Providers, the toolkit provides tools for deploying and managing services on their own or external infrastructures provided by one or more IPs, possibly via a broker, based on parameters such as trust, risk, eco-efficiency and cost.
- **Application/Service Developer:** An individual or group of individuals developing applications or services for service providers or end-users, to be deployed on the infrastructure provider's infrastructure/resources. The Toolkit helps developers easily orchestrate cloud services from scratch and run legacy applications in the cloud.
- **End-user:** Person or organization that actually uses a product or service and does not resell it. OPTIMIS targets enterprises or organizations that want to deliver IT as a service to their own organizations using a combination of on-premise and hosted resources.

2.2 Positioning of the OPTIMIS Toolkit

OPTIMIS is a software toolkit that infrastructure and service providers and end-users deploy in their data centers. It encompasses enabling technologies that are required for consumers to easily and efficiently use cloud functionality to their best advantage. It is a complement to cloud management, orchestration and application lifecycle management platforms. It gives service providers the capability to easily orchestrate cloud services from scratch and run legacy applications on the cloud. It helps them make intelligent deployment decisions based on their preference regarding trust, risk, eco-efficiency and cost (TREC). It gives them the choice of developing once and deploying services on 'best execution venues' and across any type of cloud environments – private, hybrid, federated or multi-clouds. End-to-end security, data protection and compliance with green legislation are supported by the Toolkit.

Although the competition on the market is fierce, none of the identified competitor products cover the full range of OPTIMIS features. The most important attribute of OPTIMIS, the TREC framework, continues to be the key selling point along with the service orchestration features and catalog of deployment options. The figure below shows the position of OPTIMIS and its competitors. The open source cloud projects illustrated in the positioning chart are viewed as complementary products rather than competitors.

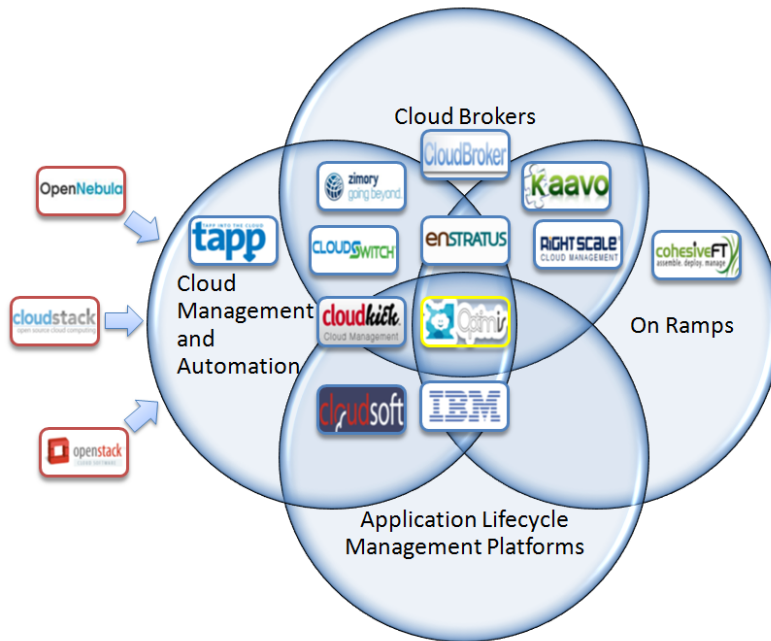


Figure 2 – Positioning of OPTIMIS

2.3 The OPTIMIS Toolkit

The OPTIMIS Toolkit can be broken down into three main groups of components: the **OPTIMIS Base Toolkit** with functionalities common to all components; the **OPTIMIS SP Tools** that enable service providers to implement, package, deploy and operate services, and finally the **OPTIMIS IP Tools** with functionality to manage the infrastructure (VMs, servers, data storage, etc.) required to operate services.

The **OPTIMIS Base toolkit** helps make optimal service deployment and operation decisions, and also provides fundamental services such as monitoring and security:

- The **Monitoring Infrastructure** gathers runtime information about physical hardware, virtualized hardware, energy consumption, and service performance. This information is aggregated and propagated to the TREC framework for analysis.
- The **Trust** framework allows IPs and SPs to determine each others' reputation and level of trustworthiness. Historical information is used to determine the trust level among two providers. However, if there are no historical values, the trust can be determined by applying social networking mechanisms and using the opinion of all providers who have previously worked together.
- The **Risk** tools enable IPs and SPs to define, assess, and manage risk, for instance, the likelihood of an event and its impact on operation, regarding key aspects of service provisioning. They support SPs in making informed, risk-aware decisions at service deployment when pre-selecting IPs, and help IPs manage well-balanced infrastructures, so they can maximize the Quality of Service (QoS) and minimize the number of SLA violations during service operation.
- The **Eco-efficiency** tools provide assessments and forecasts of relevant energy-related metrics for Cloud providers such as Energy Efficiency (Performance/Watt) and Ecological Efficiency (Performance/Kg CO2).
- The **Cost** tools allow IPs to both assess their costs and predict future costs for the provisioning of a service or component through comprised cost models which allow for cost related alternative options and configurations to be evaluated. Options for pricing are also included on the IP side. On the SP side, quotes can be obtained for a service or component during at deployment time, and costs can be monitored during run time. Cost can be optimized with the other three TREC factors.

- The **Security** services provide secure data communication and storage for the virtual machines constituting a cloud service, which may be deployed on multiple cloud providers. This is done by provisioning a virtual private network between the virtual machines, with the help of a peer-to-peer overlay network. The secure communication and storage of data is managed by the Cloud Broker or the Service Providers, by the administration of security agents deployed on the virtual machines whose behavior is controlled via security policies.

OPTIMIS allows IPs to effectively and efficiently manage infrastructure resources at a higher level of abstraction and enables SPs to create, deploy, and operate services with assessed and guaranteed TREC levels through the following tools:

- The **Programming Model** allows the creation of complex services by composing pieces of source code, licensed software, legacy applications and services. Although the composition is described as a sequential Java code, the programming model framework orchestrates the concurrent execution of its components on a pool of available resources which the programmer is not aware of.
- The **Integrated Development Environment** automates and simplifies the implementation of

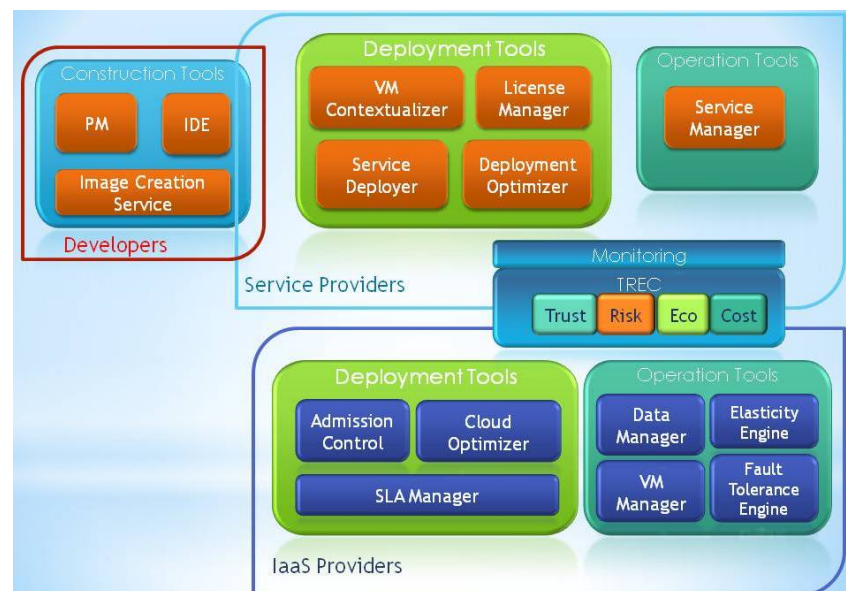


Figure 3 – Overview of OPTIMIS Tools

- new complex services by providing a GUI where these can be programmed (directly by writing Java code, or by using wizards to ease the porting of legacy applications). It automatically generates the VM images (using the Image Creation Service component) and the service manifest needed for deploying a service. It also includes options for starting the deployment of the service, interfacing with the OPTIMIS Service Deployer.
- The **Image Creation Service** automates the generation of VMs by implementing services and software pre-requisites.
- The **Virtual Machine Contextualizer** configures VM hardware at the Operating System level. It tailors the configuration of software within a VM image including other OPTIMIS components, dependencies of an application and applications themselves, for deployment into the IP's environment. This enables the dynamically changing environment of an IP to be abstracted away from the application deployed, enabling enhancements to scalability, performance, fault tolerance and other non-functional requirements such as security and license management.
- The **Service Deployer** manages the deployment of a new service, be it on private or multi-clouds. It also provides utility libraries that simplify bursting, federation, and cloud brokerage. For a service provider, the Service Deployer reduces the complexity of interacting with all these types of clouds to one simple call.
- The **Deployment Optimizer** is a decision-making tool used by the Service Deployer. It uses the TREC framework to negotiate with Infrastructure Providers in order to identify optimal execution venues for service deployment, for example by minimizing risk and/or service provisioning costs.

- The **License Manager** provides the mechanisms for using license protected software for cloud provisioned services while protecting the IPR of the software vendor.
- The **Service Manager** is used for service deployment and operation, and keeps track of all service runtime data. It also features functions for the automated management of service operation on the SP side (e.g. service redeploy according to pre-defined rules).
- The **Admission Control** component decides whether or not it is worth to admit a set of services into the cloud, and in case of acceptance, obtains the optimum allocation for each of the VMs that comprise the services using TREC-driven policies.
- The **Data Manager** creates on demand 'Data Analytics as a Service' clusters, which can be used as simple storage or added value data processing tasks, while meeting legal constraints regarding location and handling of data in a federated environment. The management of the 'DAaaS' clusters may be used on parameters such as risk and workload forecasting.
- The **Cloud Optimizer** is a global decision and management entity for IPs that combines the monitoring and assessment tools in the OPTIMIS Base Toolkit with the above listed management engines in order to create a policy-driven, self-managed cloud driven by the IP's overall business-level objectives.
- The **SLA Manager** negotiates, creates and monitors SLAs between SPs and IPs. It can also generate alerts in case some agreed terms are not fulfilled and evaluate the guarantees associated with the individual terms.
- The **VM Manager** provides basic plumbing mechanisms for starting/stopping/migrating VMs, and scheduling policies for the efficient allocation of VMs across the infrastructure and the management of physical hosts.
- The **Elasticity Engine** SLAs while reducing costs. It proactively allocates and de-allocates resources to a service. The number of provisioned VMs changes with changes in the service workload dynamics in terms of request arrival rate, and/or any other SLA metric of relevance for the service.
- The **Fault Tolerance Engine** ensures reliability by restarting failing VMs, recovering failures of physical hosts and initiating preventive actions when these failures are foreseen.

3 OPTIMIS Value

3.1 Value Proposition

OPTIMIS addresses the scenario where most companies use private and public clouds, on-premise and hosted resources in combination. OPTIMIS has tools for construction, deployment and operation of services. The key value propositions of the OPTIMIS Toolkit are:

- ✓ The Toolkit makes it easier for developers to orchestrate value-added composite cloud services and run legacy applications in the cloud. Developers do not need any cloud expertise. Service elements can be individually deployed on different venues. VMs are created automatically.
- ✓ The Toolkit helps service providers make informed decisions regarding the most suitable deployment venues across any type of cloud environments -private, hybrid, cloud-bursting, federated or multi-clouds- based on trust, risk, eco-efficiency and cost (TREC). It supports end-to-end security and compliance with data protection and green legislation.
- ✓ The Toolkit simplifies and automates the management of cloud infrastructures even in complex scenarios, such as cloud federation, based on business-level objectives such as trust, risk, eco-efficiency and cost.
- ✓ The Toolkit allows end users to become service providers and deliver IT as a service to their own organizations using a combination of on-premise and hosted resources.

3.1.1 Service Providers

The lack of trust and perceived risk related to cloud infrastructures are often identified as a barrier for widespread adoption of cloud computing. The overall goal is to foster the uptake of an open market by making providers' services transparent to the users.

In OPTIMIS, the SP plays the role of an intermediary between end-user organizations, that want to deploy their applications on the cloud, and the IPs, that offer cloud resources. The SP would inform the end-user about the overall reliability of IPs, thanks to a novel decision-making approach based on Trust, Risk, Eco-efficiency, and Cost factors. The SP acts as a negotiator and introduces '*TREC assessment in Service Level Agreements (SLA)*' and gains the ability to assess the reliability of various IPs.

The market will see clear benefits by incorporating the SP innovations into their structure and services. The business model is highly sophisticated and should attract interest from cloud service providers that want to avoid vendor lock-in and increase quality of service.

The SP provides the following capabilities:

- Find suitable IPs for the execution of a given service (described as a SLA request).
- Pre-negotiate a SLA with different candidate IPs on behalf of the end-user.
- Return a list of suitable IP offers (SLA offers) taking TREC information into account.
- Return provider rankings based on various criteria and tailored to the end-user's preference.
- When a suitable IP for deployment has been found, the SD triggers *contextualization* (configurations) of the VMs, and uploads the data to the repository of the chosen IP.
- Once the data transfer is completed, execution of the service is initiated by calling the local *Service Manager* component, which is responsible for service lifecycle management.
- Orchestrate entire services across a number of components.
- Act as a virtual IP, agreeing its own SLAs with end-users.

3.1.2 Infrastructure Providers

The OPTIMIS Toolkit simplifies and automates the management of the IP's infrastructure. IPs can specify business-level objectives (TREC factors), which will drive all management decisions. For instance, the IP can optimize the placement of VMs in order to reduce power consumption or to reduce the risk of SLA violations due to physical resource failures.

The OPTIMIS Toolkit allows gathering runtime information about physical hardware, virtualized hardware, energy consumption, and service performance, which can be used to track the variability inherent in cloud environments. This way, the IP can adapt its management decisions to changes and react to unexpected situations, for instance, restarting services when resource failures arise.

Using the OPTIMIS toolkit, IPs can work smoothly in complex scenarios, such as cloud federation. An IP can outsource resources to other IPs when local infrastructure is not enough to fulfill the SLA. The toolkit ensures the interoperability of OPTIMIS-based IPs by defining clear communication interfaces between them and standard protocols as well as data formats. These IPs can even interoperate with non-OPTIMIS-based IPs via simple adaptors included in the Toolkit.

3.1.3 Developers

Software developers can choose from many existing tools to program a service. However, existing tools fail to provide true cloud-unaware programming combined with easy service composition and orchestration. PaaS offerings such as Microsoft Azure or Google App Engine, require the programmer to use APIs for inter-process communication or for accessing cloud storage. Moreover, the deployment and execution of applications with these vendors are tied to their own infrastructure. Other tools such as WS-BPEL, SWPT, FAST, offer the graphical composition of service-based workflows, where data dependencies between services need to be manually and statically specified. Most of these graphical tools also lack the ability to specify conditions or introduce control flow statements in an imperative language. On a last note, programming models for distributed infrastructures adapted to the Cloud are either not flexible enough for all kinds of applications (i.e. MapReduce) or do not offer special support for service composition and orchestration (i.e. Aneka).

The OPTIMIS Programming Model improves current state of the art by providing a programming model and runtime system for developing and running added value composite cloud services. It goes a step further and addresses how cloud computing changes application architecture and development. In the OPTIMIS Programming Model, only sequential programming skills are required to write SaaS composites; the inner services are easily invoked as normal methods with no use of any library or new syntax. Service tasks can be combined with non-service ones and altogether can be automatically orchestrated by the runtime system. The runtime system is able to run in cloud environments and elastically reserve virtual resources depending on the number of requested tasks. The OPTIMIS programming model can potentially work on top of any cloud provider.

The use of the OPTIMIS programming environment can significantly reduce time-to-market needed to develop and deploy a new service on the cloud without any switching costs charged for changing cloud platforms.

More specifically, the OPTIMIS construction tools include the Integrated Development Environment (IDE), the Image Creation Service (ICS), and the Programming Model runtime. The IDE assists in the creation of the new service, as well as the service manifest that describes the service. The IDE is accessed through the Eclipse User Interface and extends the Eclipse basic functionalities. The ICS helps create an image of the service that will be later deployed in the cloud. The Programming Model runtime is in charge of the orchestration of the programmed service.

3.1.4 End-users

Originally seen as a service-provider play, the cloud orchestration and enablement opportunity is growing as enterprises increasingly seek to deliver IT as a service to their own organizations using a combination of on-premise and hosted resources.

OPTIMIS can help IT move from being a cost center to a service-delivery function. OPTIMIS gives IT departments the opportunity to become cloud service providers and deliver cloud services to their own organizations.

3.2 OPTIMIS Value in Action

3.2.1 A Typical Use Case

CRM software company 'SOFT' wants to become a SaaS provider. SOFT wants to move its CRM application to the cloud.

The programming component of the OPTIMIS toolkit allows the service developer at SOFT to define the service elements that include functionalities SOFT wants to provide to its clients. Once the elements are defined, the construction tools of the programming component create the service VMs and a 'service manifest' that will govern the deployment and operation of the created VMs/services.

Using the TREC optimization engine, the toolkit manages the deployment of the VMs to the service provider's preferred infrastructure.

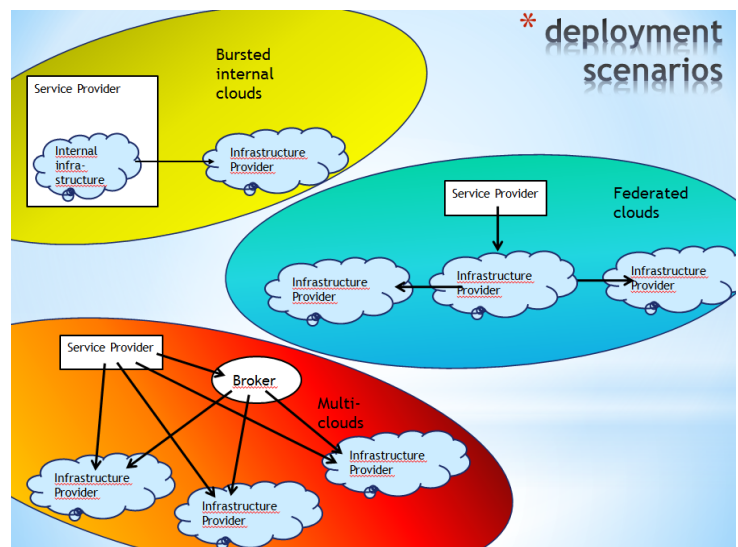


Figure 4 - Deployment Scenarios

Another component, the OPTIMIS broker, manages the infrastructure selection and service deployment by automatically evaluating infrastructure providers based on the service provider's TREC preference. These deployments can even occur across multiple cloud infrastructures if the TREC-based evaluation determines that would deliver the best result.

Once the service is live, OPTIMIS allows the service provider to continue to monitor, change and optimize the various configurations at runtime.

3.2.2 The Programming Model Use Case

Launching innovative applications requires significant investment of effort and money and their success on the market is uncertain. SMEs can't dedicate unlimited resources to do research and innovation. This use case demonstrates how the service construction capabilities of the OPTIMIS Toolkit help SMEs enter the global SaaS marketplace more easily.

More specifically, this use case is meant to validate the service construction capabilities of the OPTIMIS Toolkit. The Programming Model use case shows how to use the OPTIMIS service construction capabilities to easily develop innovative, value-added services, by focusing on two specific scenarios:

- a Genomic Computation Service, which combines different genomic services offered by Genomic Institutions with resource-intensive calculations,
- a Retailer Service, centered on the combination of current business services for reselling products offered by manufacturers in the wholesale market.

The Genomic Computation Service scenario implies a complex service composition and it is resource-intensive. Efficient execution and resource management are crucial for delivering good quality of service without shooting up operational costs. The OPTIMIS construction tools hide all the complexity and allow the service developer to intuitively implement the service without any Cloud-specific programming expertise.

In the Retailer Service scenario, the developer wants to combine existing 3rd party software in order to offer value added services. The OPTIMIS construction tools give software developers the ability to leverage existing services and easily orchestrate new ones that offer value-added features.

3.2.3 The Cloud Brokerage Use Case

At its simplest, OPTIMIS components can be used to enable a multi-cloud deployment of a service including several IPs. This in itself allows for a more dynamic and competitive ecosystem by enabling the SP to choose different IPs for hosting various components of a service.

The OPTIMIS Cloud Broker service provides two capabilities that build on each other:

- Ability to approach the Broker with a set of physical requirements of the service components (like memory, CPU, network bandwidth, etc.) as well as a set of constraints (like cost, trust levels, risk level and eco-efficiency certification) and be provided back with the best possible venues the various components can be deployed on, using one or more IPs that provide compatible services.
- Further intermediation between the SP and the IPs by receiving the service details and the associated VMs, data storage, etc. and doing the complete onboarding of the service across multiple IPs, on behalf of the SP.

The Cloud Brokerage use case is expected to bring out the following innovations and value:

- Matching of requirements and functionalities: providing standards-based SLA negotiation and agreement mechanisms to allow the broker to match the requirements of the SP with the functionalities provided by the various IPs. This is something that is lacking in the current offerings and forms the basis of the broker functionality.
- TREC based decision-making: The OPTIMIS toolkit provides means to make decisions on which IP and SP to engage with based on trust, risk, eco-efficiency and cost parameters. The use of this functionality allows the broker to make SP-IP matches based on TREC. While the cost parameter might be an obvious functionality, the other three metrics are a definite value-add.
- Value-added services: the cloud broker feature will provide value-added services to the SP (and even the IP), making it easier for the parties involved to move services to the cloud. For example, the ability to enforce corporate IT policies in a cloud environment as effectively as in a corporate data center is a much sought after capability, along with seamless identity management capability, ability to provide end-to-end encryption support across multiple IPs, ability to provide access to secure storage capability, etc., and the broker is in a key position to provide some of these capabilities.
- Federation capabilities: In the current offerings, the IP is able to leverage only its dedicated infrastructure resources to provide a service to the SP. However, the OPTIMIS toolkit allows the IP to internally use the services of (federate to) other IPs to cope with scaling demands. This capability can be brought into the brokerage scenario, allowing for a more efficient and resilient working of the IPs.

3.2.4 The Cloud-Bursting Use Case

The inherent advantages of on-demand resources within a virtualized Cloud environment are well documented, however, the ability to cloud-burst takes this on demand concept to new levels allowing the ability to create a so called eco-system of resources across previously autonomous, independent systems.

The term 'cloud-bursting' has been coined to describe the ability for an IP to externalize computing resources when a given situation and user requirements dictate so. The ability to do this provides many advantages such as:

- A new form of redundancy, where services can be configured to automatically start on an alternative provider when required.
- Dynamic scalability of resources to match service requirements.
- Cost management where externalization is only used when required.
- Enhanced QoS guarantees where greater user expectations can be supplied.
- A more solid expectation of SLA fulfillment can be given.
- An optimal platform can be supplied for user needs.
- Dynamic decisions based on active monitoring maintain the optimization of resources.

By a service provider utilizing the OPTIMIS toolkit, one organization can break the boundaries and limitations of its favored public or private Cloud environment and effectively include the resources of another provider based on trust, risk, eco-efficiency and cost.

IPs can become part of a Cloud-bursting scenario in different ways. An IP could for example incorporate the full OPTIMIS toolkit allowing the full stack of advantages such as offering monitored resource data, historical performance data and cost parameters to external parties, encouraging the use of their platform and allowing a means for platform users to gain similar resources externally. These external resources can be automatically selected using the parameters set by the service provider to choose the most apt IP for their needs or indeed incorporate multiple IPs into the scenario such as placing more important services to a higher QoS offering and less important compute needs to the cheapest of the available IPs.

Alternatively, an IP could simply agree to be part of a scenario where they do not have to incorporate these tools to their platform by allowing external OPTIMIS tools to rent resource from their platform in a standard way using a standard API. Anything in between these scenarios is also possible, for instance, an IP only incorporating some of the OPTIMIS features. This way an IP can be slowly introduced to the OPTIMIS ecosystem and conform to more of the OPTIMIS features over time and at its own pace.

4 Adopting the OPTIMIS Toolkit

As previously stated, the OPTIMIS Toolkit comprises a set of tools to be used by Service Providers, Infrastructure Providers, Software Developers and End-users to easily orchestrate cloud services and run them on the most appropriate execution venues.

Software Development Tools consist of the OPTIMIS Programming Model and IDE, as well as, the Image Creation Service. These tools automatize code parallelization, packaging, installation of application and software dependencies and generation of VM Images. The generated VM images can then be executed by the OPTIMIS Toolkit or any other Cloud fabric controller, VM Manager or hypervisor technology compatible with the hypervisor that generated the images. Consequently, Software Developers can adopt OPTIMIS Development tools independently of Service and Infrastructure Provider Tools, although in this case deployment and operational capabilities provided by the OPTIMIS Toolkit, such as BEV selection, Cloud self-management (automated elasticity and fault tolerance) and Data management (legal control on data placement), won't be applicable.

Service Provider Tools enable SPs to deploy and operate services in multiple private and hybrid Cloud scenarios (bursting, federated and brokerage). It is able to interact both with OPTIMIS enabled providers and

non-OPTIMIS providers by using a plug-ins developed for specific provider APIs. Therefore, the OPTIMIS Service Provider Tools are interoperable both with the OPTIMIS Infrastructure Provider's Tools and other infrastructures through plug-ins.

The OPTIMIS Infrastructure Provider Tools are a layer on top of cloud fabric controllers and storage services such as, eMotiveCloud used by default in the OPTIMIS Toolkit and OPTIMIS Data Management, or others such OpenStack Compute and Storage, Open Nebula or VMWare vSphere. Adoption by infrastructure providers aims to offer two

possibilities: "Full-OPTIMIS", complete adoption of the Toolkit including eMotiveCloud and OPTIMIS Data Management; and the so-called "OPTIMIS-Enhanced" option, in which parts of the toolkit interact or are substituted by other existing tools. Examples of OPTIMIS-Enhanced adoption are infrastructures adopting the entire OPTIMIS Infrastructure toolset but changing the cloud fabric controller, or even, adoption of a reduced set of capabilities such as improved Fault Tolerance through Risk assessment tools.

5 Toolkit Launch

The OPTIMIS project released the first version of its open source Toolkit on 1 June 2012. It is available for download on the OPTIMIS website (<http://www.optimis-project.eu/>). A next version will be released in October 2012 and further releases will be announced throughout 2013.