

Deliverable D2.2

Final inter-layer and inter-provider interface specifications for service-centric networking

Public report, Version 1.2, September 2014

Authors

UCL David Griffin, Miguel Rio
ALUB Frederik Vandeputte, Luc Vermoesen
TPSA Dariusz Bursztynowski
SPINOR Michael Franke, Folker Schamel
IMINDS Pieter Simoens, Piet Smet

Reviewers

Abstract This document specifies the components of the FUSION architecture for service-centric networking; the interactions between the identified functional components; and the interfaces defined between the components.

Keywords Service-centric networking, architecture, interface, API

© Copyright 2014 FUSION Consortium

University College London, UK (UCL)
Alcatel-Lucent Bell NV, Belgium (ALUB)
Orange Polska S.A., Poland (TPSA)
Spinor GmbH, Germany (SPINOR)
iMinds vzw, Belgium (IMINDS)



Project funded by the European Union under the
Information and Communication Technologies FP7 Cooperation Programme
Grant Agreement number 318205

Revision history

Date	Editor	Status	Version	Changes
11/02/2014	David Griffin	Initial Version	0.1	ToC
24/02/2014	David Griffin	Initial Version	0.2	Revised ToC and partner allocations
28/03/2014	David Griffin	Consolidated version	1.0	Full initial draft
22/05/2014	David Griffin	Revision	1.1	Updated contributions
10/06/2014	David Griffin	Revision 2	1.1a	Revised system architecture
09/09/2014	David Griffin	Stable version	1.1b	Consortium meeting review
19/09/2014	David Griffin	Final version	1.2	Final version

TABLE OF CONTENTS

TABLE OF CONTENTS	3
1. FUSION ARCHITECTURE OVERVIEW	4
2. INTERACTION SCENARIOS	6
3. INTERFACE SPECIFICATION	10
3.1 Service Registration and Deployment Interfaces	10
3.1.1.1 Registration interface	11
3.1.1.2 Deployment interface	13
3.1.1.3 Data-centre adaptor Interface	16
3.1.1.4 Inter-orchestrator interface	18
3.2 Service Resolution Plane Interfaces.....	20
3.2.1 <i>Instance reporting interface</i>	20
3.2.2 <i>Execution zone reporting interface</i>	21
3.2.3 <i>Execution zone advertisement interface</i>	21
3.2.4 <i>Interdomain advertisement interface</i>	22
3.2.5 <i>Forwarding table configuration interface</i>	23
3.3 Monitoring Interfaces.....	25
3.3.1 <i>Local and external network monitoring interfaces</i>	25
3.3.2 <i>Service resolution domain monitoring interface</i>	27
3.3.3 <i>Execution zone monitoring interface</i>	27
3.3.4 <i>Service monitoring interface</i>	29
3.3.5 <i>QoE reporting interface</i>	29
3.3.6 <i>Data-centre adaptor monitoring interface</i>	30
3.4 Service Query/Invocation Interfaces	31
3.4.1 <i>Client query/invocation interface</i>	32
3.4.2 <i>Interdomain query/invocation interface</i>	34
3.4.3 <i>Execution zone query/invocation interface</i>	36
3.5 Service Delivery Interfaces	38
3.5.1 <i>Service delivery/data-plane interface</i>	38
4. INTERFACE TECHNOLOGIES	38
4.1 Orchestration/zone management	38
4.2 Service resolution	39
4.3 DC specific	39

1. FUSION ARCHITECTURE OVERVIEW

The FUSION framework can be seen in Figure 1. Functionality is divided into 3 layers. At the lower layer IP routing forwards packets using traditional end-to-end protocols. At the upper layer the execution plane consists of all the execution zones where service instances will run. In the middle the service resolution layer forwards request from clients to service instances.

The basic operation of the FUSION system is that orchestration domains – consisting of a potentially large number of geographically distributed execution zones – deploy services on behalf of service providers in one or more execution zones according to the expected demand by service users. This is depicted in the upper layer of Figure 1. Service resolution domains are responsible for matching service requests to execution zones containing running instances of the requested service. This is depicted in the middle layer of Figure 1. Service resolution is anycast in nature – the user simply requests a service and it is the responsibility of the service resolution plane to find the “best” available instance for that request. Once a specific service instance in a specific execution zone has been selected for the user request, data plane communications take place in the data forwarding plane depicted by “IP Routing” in the lower layer of Figure 1.

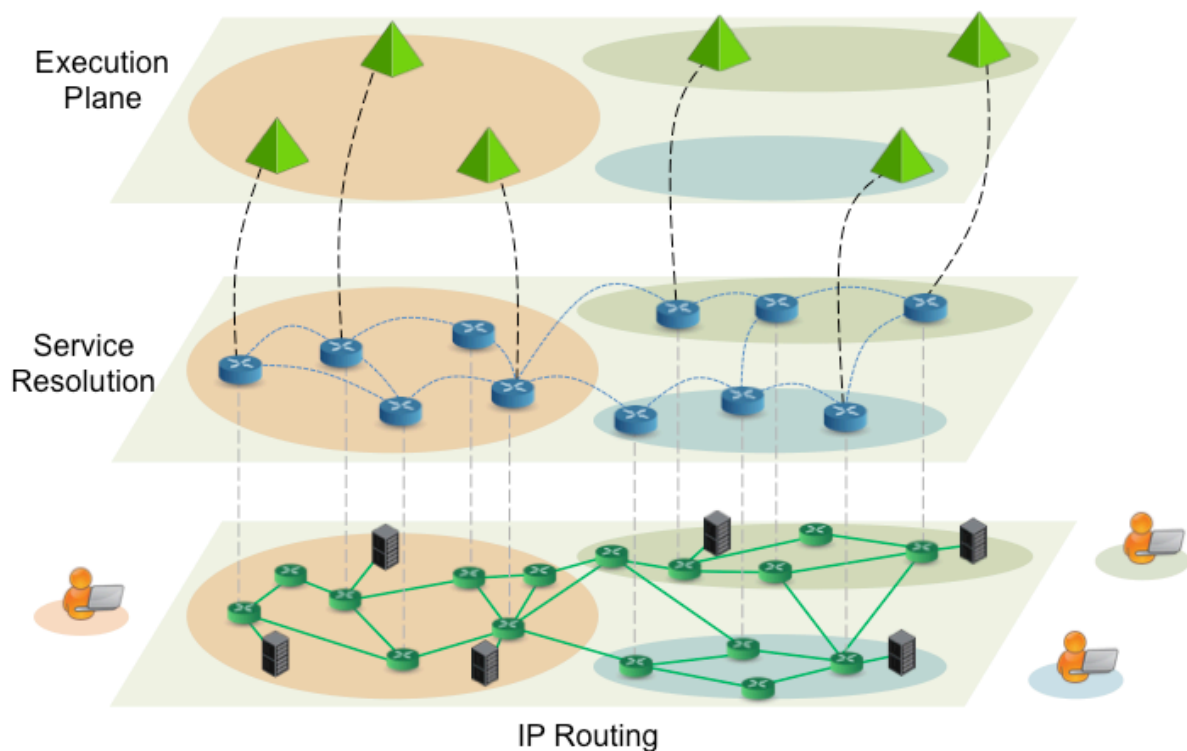


Figure 1: FUSION framework

The functional entities in the FUSION architecture are depicted in Figure 2. The three main entities are the orchestrator, execution zone and service resolver.

The *orchestrator* manages its orchestration domain resources including execution zones and services, which it manages on behalf service providers. The orchestrator is responsible for service management functions including service registration, server placement (selecting appropriate execution zones to execute service instances), service lifecycle management and monitoring.

The *execution zone* is the logical representation of the collection of physical computational resources in a specific location, such as a data centre, which is managed by an orchestrator. The orchestrator has an abstract view of an execution zone and the detailed internals are managed by a zone

manager. The zone manager is responsible for managing service instances within its zone but under the instruction of the orchestrator.

The *service resolver* is responsible for maintaining and managing service resolution information to create forwarding paths for queries to be resolved to execution zones containing available running instances of the specified service.

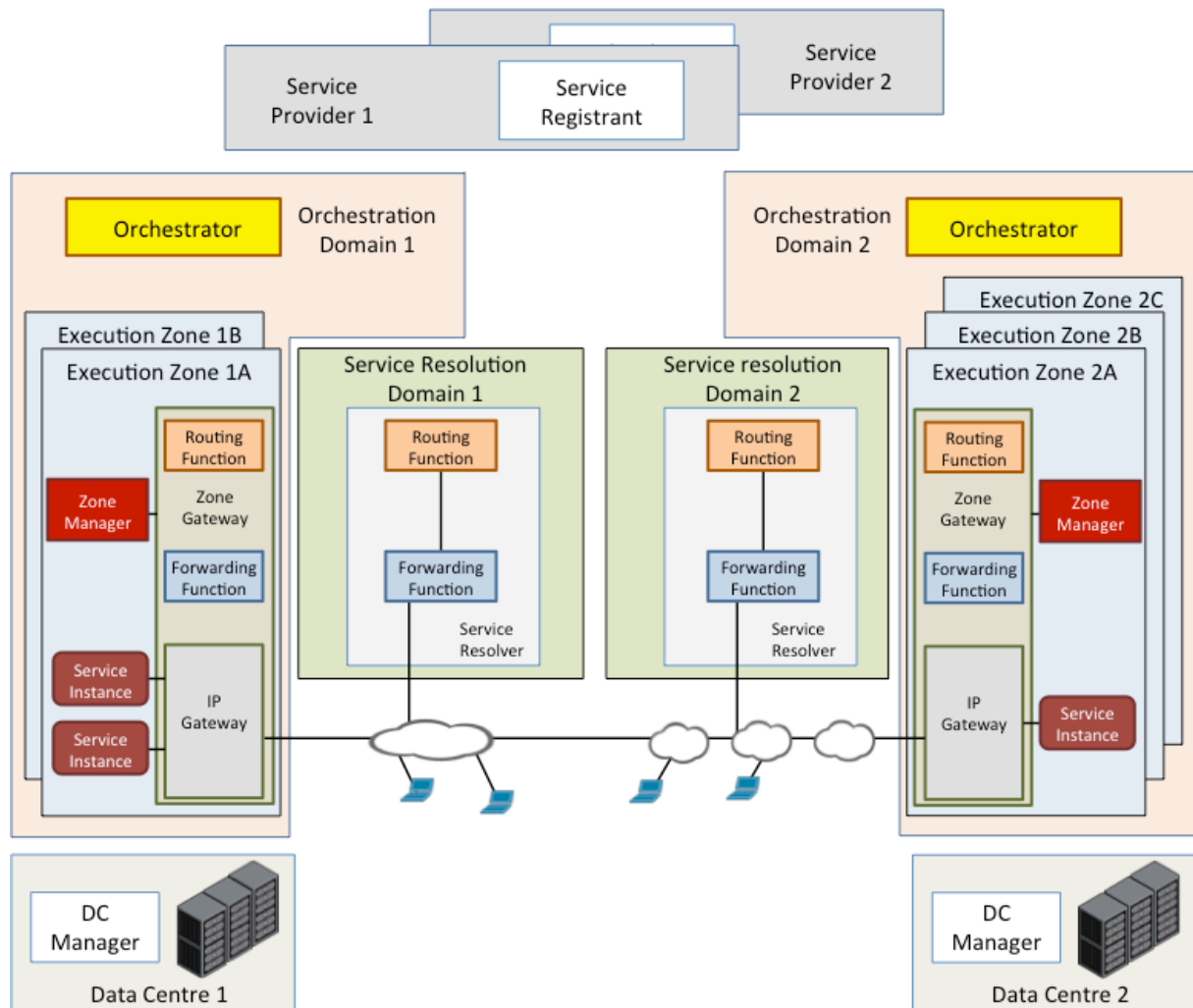


Figure 2: FUSION functional architecture

2. INTERACTION SCENARIOS

The following figures illustrate the interactions between the architectural components for the main scenarios involved in service deployment, advertisement and querying.

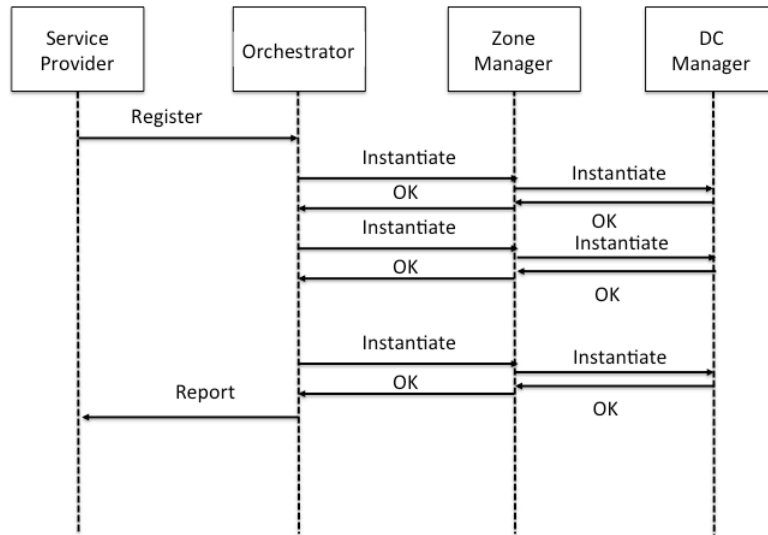


Figure 3: Service registration and deployment

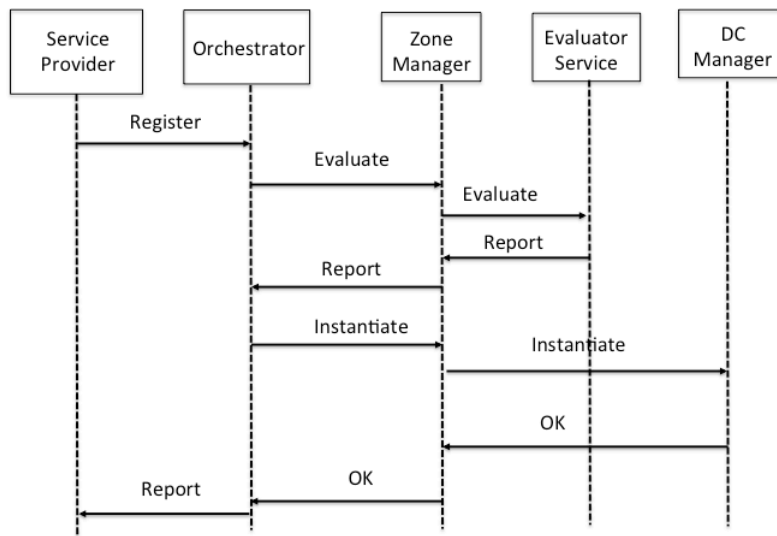


Figure 4: Service registration and deployment following evaluation

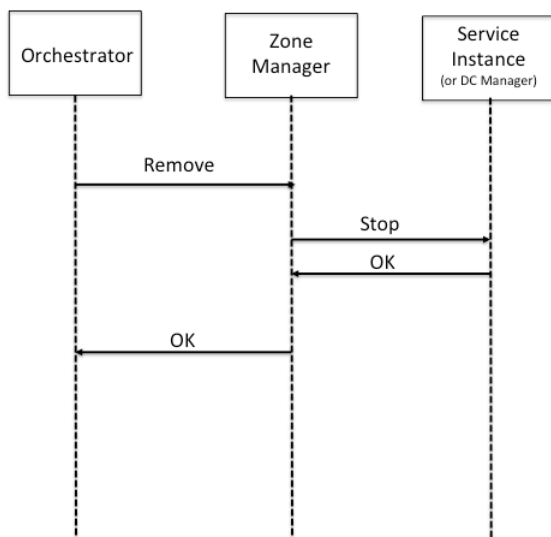


Figure 5: Service instance termination

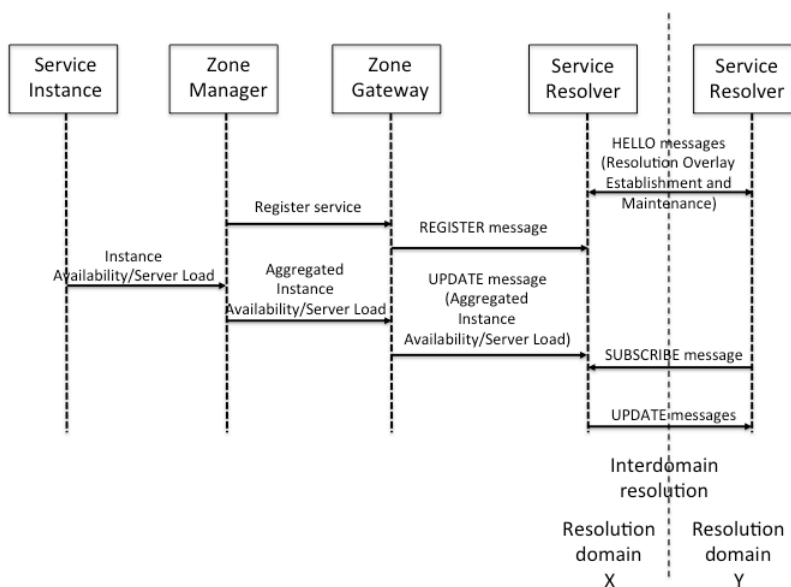


Figure 6: Service advertisement propagation

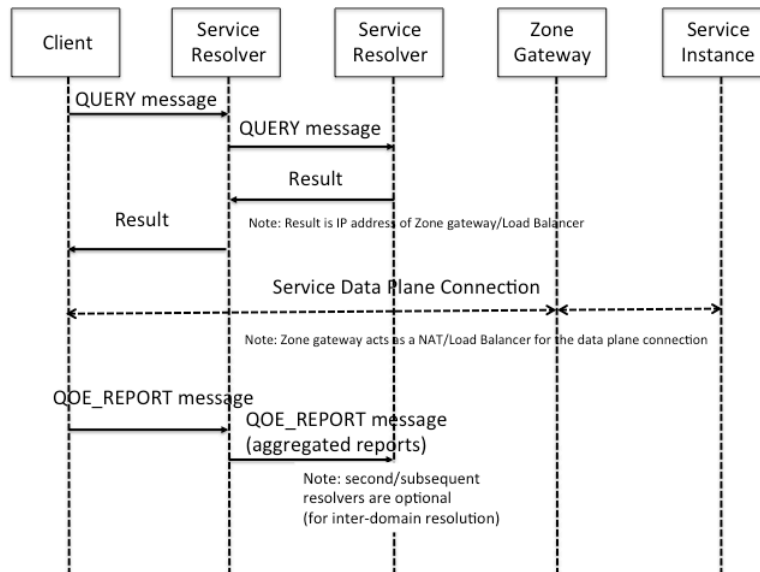


Figure 7: Service resolution

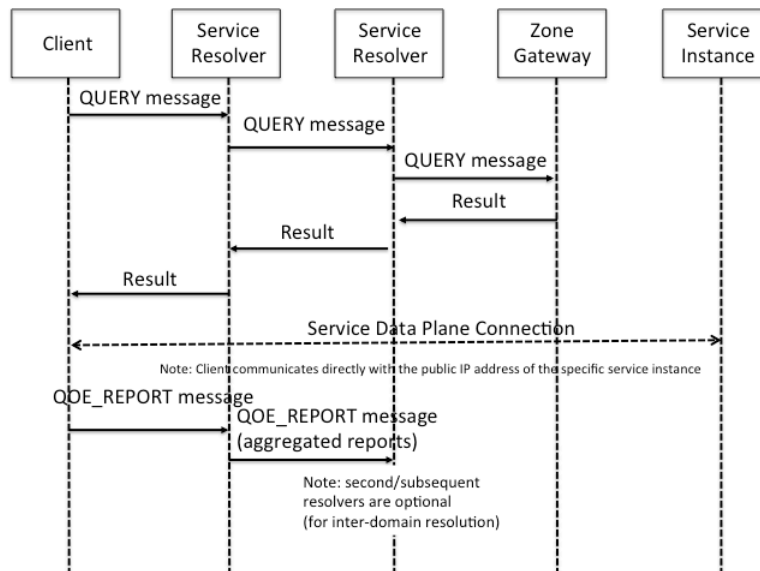


Figure 8: Service resolution, with optional involvement of destination zone gateway

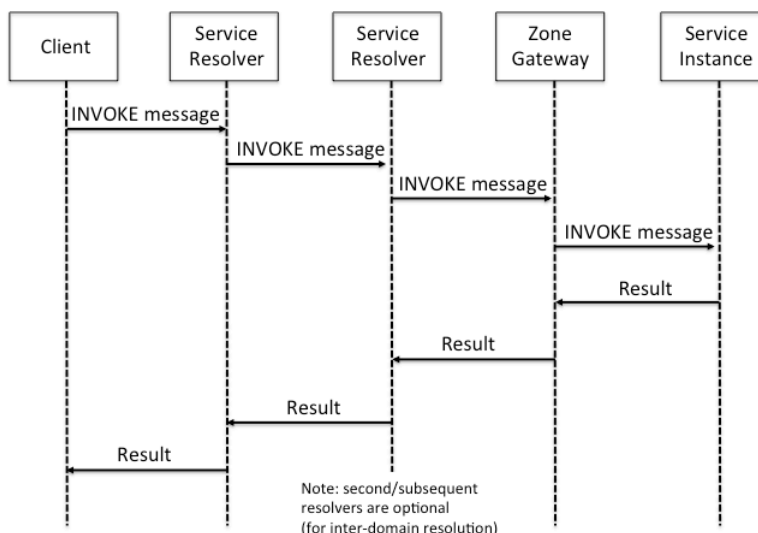


Figure 9: Optional service invocation through resolution plane

The following sequence chart shows service instantiation on demand. Rather than instances being pre-deployed in execution zones and available session slots being announced to the service resolution plane as in the previous cases, the *orchestrator* registers with a service resolver and announces session slots – with the semantic that the slots are potentially available on demand. In this way the service resolver is able to forward queries from clients and the fact that the service has not been instantiated is transparent to the resolution plane.

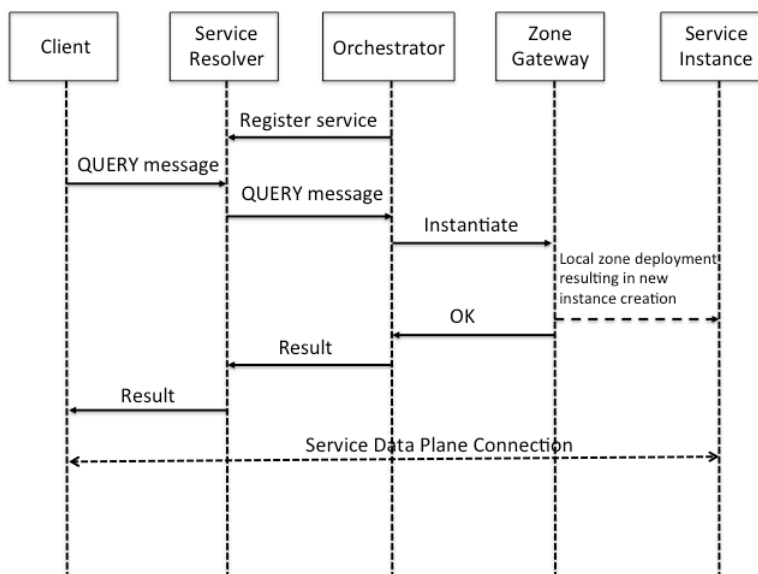


Figure 10: Service instantiation on demand

3. INTERFACE SPECIFICATION

This section identifies and specifies at a high level the interfaces between the functional components of the FUSION architecture. Implementation-specific details will be provided in FUSION deliverables D3.2, D4.2 and D5.2.

3.1 Service Registration and Deployment Interfaces

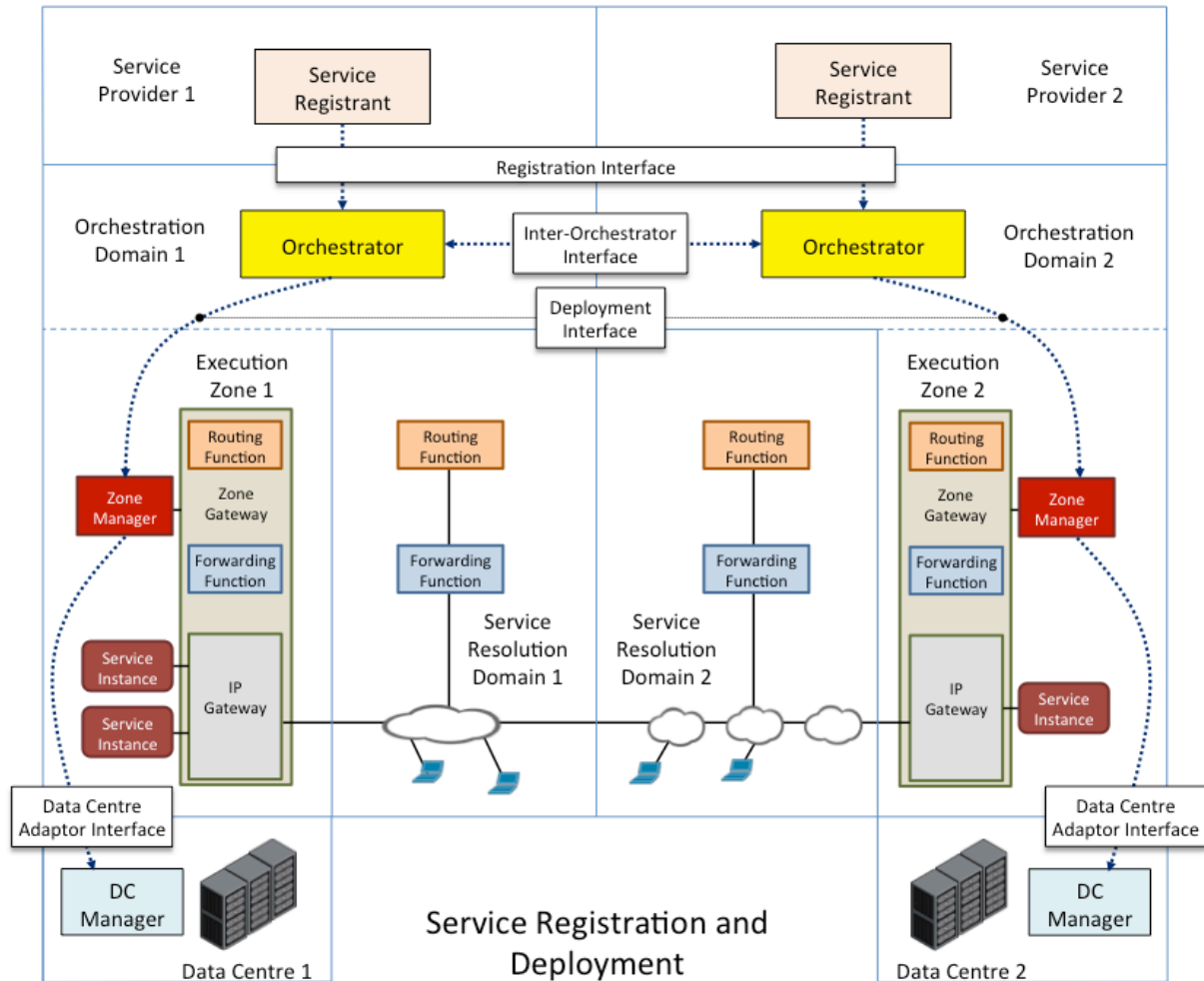


Figure 11: Service registration and deployment interfaces

3.1.1.1 Registration interface

FUNCTION NAME		
FUSIONRegisterService (service provider → domain orchestration)		
BEHAVIOUR		
<p>Register a new service type to an orchestration domain. In the current model, the manifest should contain all necessary information regarding the service, its policies and constraints and where to find the software packages and/or images.</p> <p>Upon registration, the domain orchestrator will (fetch and) parse the manifest file, fetch its external dependencies (e.g., the software VM image or container), assign a unique service ID to it and trigger the HEAT-like orchestration block to start provisioning and deploying a number of instances according to the service requirements. Obviously, it will also first verify whether all SLAs and cost models can be applied for that tenant (i.e., the registered service provider).</p>		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceManifest	Manifest/URL	Service manifest (or URL to manifest) containing all information
RETURN VALUES		
Name	Type	Description
Status	Int	Return code
ServiceID	FUSIONID	The registered FUSION service name

FUNCTION NAME		
FUSIONUnregisterService (service provider → domain orchestration)		
BEHAVIOUR		
<p>With this function, authorised users can remove an existing service type from an orchestration domain. This may include stopping all active service instances first. In the latter scenario, it may take a while before all instances are effectively stopped and removed from all execution zones, the service resolution plane, etc.</p>		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	Name of the service to unregister
StopServices	Bool	Immediately stop all existing instances
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONUpdateService (service provider → domain orchestration)		
BEHAVIOUR		
<p>This function can be used to provide updates to an existing service type. This may include changed requirements, policies or cost models, load patterns, etc.</p> <p>In the simplest scenario, the service provider simply uploaded a new manifest and the domain orchestrator first checks whether the new manifest is compatible with the old manifest. If this is not the case (e.g., in case of non-trivial or functional cases), the domain may decline the update request, forcing the service provider to create a new service type and remove the old one.</p> <p>For example, the latter approach is most likely preferable in case a new version of the service is available: to avoid versioning issues, the new version should best be registered as a new service type.</p>		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceName	FUSIONID	Name of the service to update
Manifest	Manifest/URL	New/updated manifest for the service
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONQueryServiceByName (service provider → domain orchestration)		
BEHAVIOUR		
<p>This function allows authorised users to query for particular information of a specific service type. If found, the domain orchestrator returns all active information of the requested service instance. This function for example allows for other service providers to use a particular service and incorporate it in another composite service.</p>		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	Name of the service under query
InformationType	Enum	Subset of what kind of information needs to be retrieved
RETURN VALUES		
Name	Type	Description
Status	Int	Return code
Description	Structured	Structured description of the requested information (XML, manifest, etc.)

FUNCTION NAME		
FUSIONQueryServiceByCategory (service provider → domain orchestration)		
BEHAVIOUR		
This function allows authorised users to query for particular services providing particular functionality or have particular requirements or features. This function could be used by for example a FUSION app store let users browse through all available and publicly accessible service types.		
PROPERTIES & PARAMETERS		
Name	Type	Description
Categories	List	List of features to look for
RETURN VALUES		
Name	Type	Description
Status	Int	Return code
ServiceNames	[ServiceName]	A list of all registered services that match the query

3.1.1.2 Deployment interface

FUNCTION NAME		
FUSIONRegisterZone (zone administrator → domain orchestration)		
BEHAVIOUR		
With this function, a new execution zone can announce and register itself to a particular FUSION domain.		
PROPERTIES & PARAMETERS		
Name	Type	Description
Zone	URI	Name of the execution zone
Parameters	Complex	A detailed description of the zone, its location and its capabilities
RETURN VALUES		
Name	Type	Description
Status	Int	Return code
ZoneID	ZoneID	A unique zone ID for this execution zone in the domain

FUNCTION NAME		
FUSIONUpdateZone (zone administrator → domain orchestration)		
BEHAVIOUR		
With this function, an execution zone can modify its registration parameters to the domain (e.g., change key capabilities, etc.)		
PROPERTIES & PARAMETERS		
Name	Type	Description
ZoneID	ZoneID	Execution zone identifier
Parameters	Structured	An updated version of the execution zone specifications
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONUnregisterZone (zone administrator → domain orchestration)		
BEHAVIOUR		
With this function, an execution zone can unregister itself from a domain.		
PROPERTIES & PARAMETERS		
Name	Type	Description
ZoneID	ZoneID	Execution zone identifier
StopServices	Bool	This flag indicated whether it will immediately shut down existing services (this request can be refused by the orchestrator).
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONZoneEvaluate (domain orchestration → zone manager)		
BEHAVIOUR		
This function allows an orchestration domain to request a zone manager to evaluate the request for deploying a number of instances of a particular service type inside the zone managed by a zone manager.		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceName	FUSIONID	Name of the service
ServiceParameters	Blob	Deployment parameters for the service
DeployParameters	Structured	FUSION-aware deployment parameters for the zone
Timeout	Time	Maximum amount of time for making the evaluation
RETURN VALUES		
Name	Type	Description
Score	KeyValueList	The multidimensional score

FUNCTION NAME		
FUSIONDeployService (domain orchestration → zone manager)		
BEHAVIOUR		
An orchestration domain can use this function for explicitly deploying a number of instances in the execution zone.		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceName	FUSIONID	Name of the service
ServiceParameters	Blob	Deployment parameters for the service
DeployParameters	Structured	FUSION-aware deployment parameters
Timeout	Time	Maximum amount of time for deploying the services
RETURN VALUES		
Name	Type	Description
Status	Int	Return code
InstanceIDs	[FUSIONID]	List of the service instance IDs of the newly created instances

FUNCTION NAME		
FUSIONTerminateServiceInstance (domain orchestration → zone manager)		
BEHAVIOUR		
With this function, the orchestration domain can explicitly terminate particular instances. Note that alternative functions will also be possible: reducing the available sessions, etc.		
PROPERTIES & PARAMETERS		
Name	Type	Description
InstanceID	URI	Name of the service
Timeout	Time	Maximum amount of time for terminating the sessions
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.1.1.3 Data-centre adaptor Interface

FUNCTION NAME		
FUSIONDCPrepareServiceEnvironment (zone manager → DC agent)		
BEHAVIOUR		
This function enables a zone manager to create and prepare an environment for a particular service to be deployed. This can range from physical to virtualised environments, consisting of a particular amount of resources that are allocated for that service.		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceName	FUSIONID	Name of the service
EnvParameters	Structured	Environment parameters (type, resources, VM location, etc.)
RETURN VALUES		
Name	Type	Description
Status	Int	Return code
EnvironmentID	FUSIONID	Reference to the newly created DC environment

FUNCTION NAME		
FUSIONDCResetServiceEnvironment (zone manager → DC agent)		
BEHAVIOUR		
This function enables a zone manager to reset an existing service environment, allowing to quickly recycle the environment for another service deployment.		
PROPERTIES & PARAMETERS		
Name	Type	Description
EnvironmentID	FUSIONID	Reference to the environment
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONDCTerminateServiceEnvironment (zone manager → DC agent)		
BEHAVIOUR		
This function enables a zone manager to clean up and terminate the enclosing environment that was used for hosting a FUSION service. It is assumed that the FUSION service was already stopped beforehand.		
PROPERTIES & PARAMETERS		
Name	Type	Description
EnvironmentID	FUSIONID	Reference to the environment
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.1.1.4 Inter-orchestrator interface

Interworking between orchestration domains is not considered to be part of the mainstream FUSION operations. Scenarios for interworking between orchestration domains are for further study, which may require updates to the initial interfaces specified in this section.

FUNCTION NAME		
FUSIONAnnounceServiceID (domain orchestrator A → domain orchestrator B)		
BEHAVIOUR		
Orchestration domain A announces the existence of serviceIDs available through its border gateway service resolver. These are either running in execution zones within the domain or in the case of transit domains they are available in remote domains.		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	The FUSION service name.
ServiceSlots	Int	Available service slots.
BorderResolverID	Locator	Locator of next hop border gateway service resolver that the border gateway service resolver in domain B should use.
RouteInformation	Structured	Route characteristics, hop count, domain path (in the case of transit, equivalent of AS path/set in BGP).
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONUpdateRoute (domain orchestrator A → domain orchestrator B)		
BEHAVIOUR		
Orchestration domain A updates the resolution information for a serviceIDs when the number of session slots or route information changes.		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	The FUSION service name.
ServiceSlots	Int	Available service slots.
BorderResolverID	Locator	Locator of next hop border gateway service resolver associated with this update.
RouteInformation	Structured	Route characteristics, hop count, domain path (in the case of transit, equivalent of AS path/set in BGP).

RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONWithdrawServiceID (domain orchestrator A → domain orchestrator B)		
BEHAVIOUR		
Orchestration domain A announces that a ServiceID or its route is no longer available through the specified border gateway service resolver.		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	The FUSION service name.
BorderResolverID	Locator	Locator of next hop border gateway service resolver.
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.2 Service Resolution Plane Interfaces

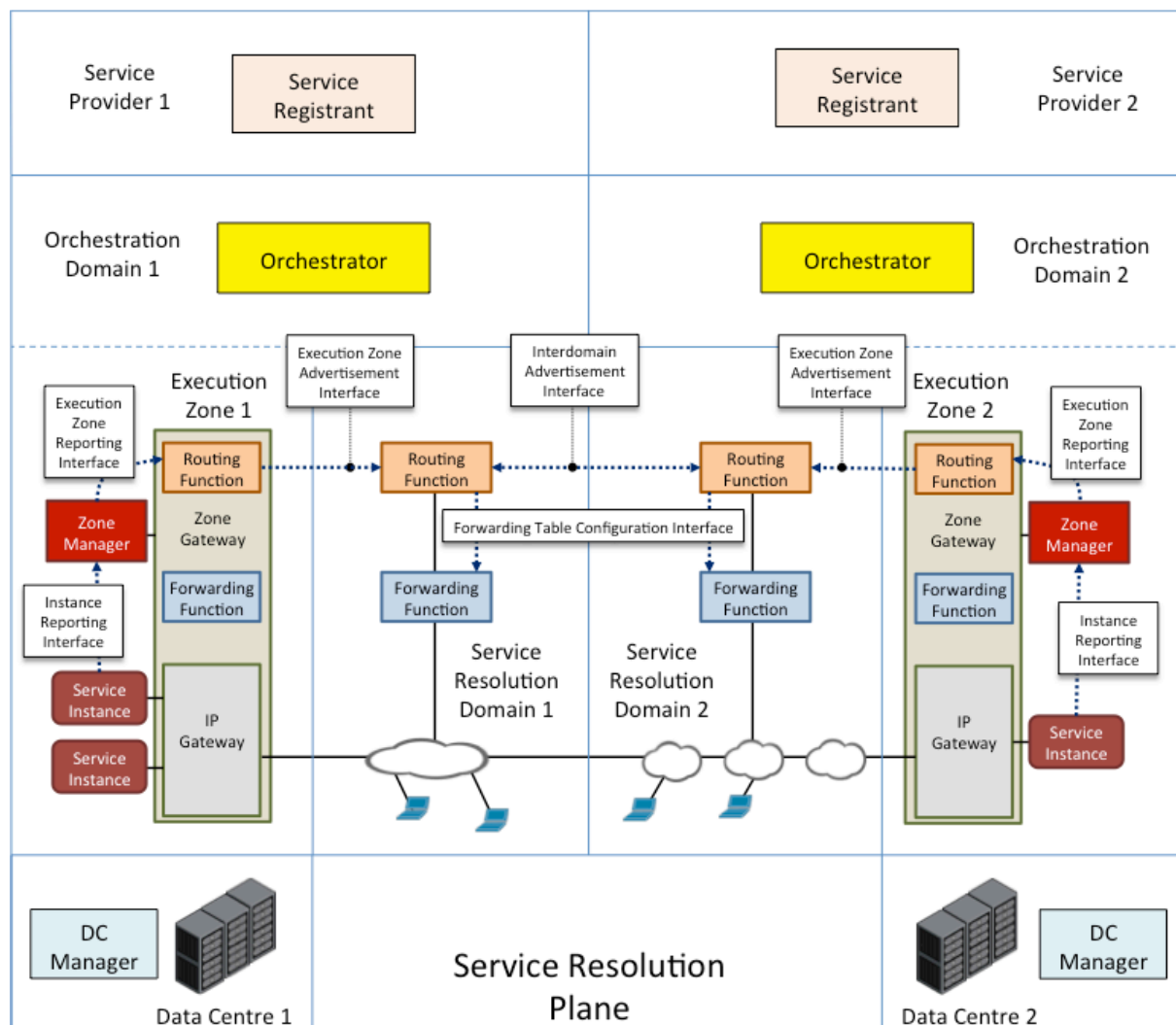


Figure 12: Service resolution plane interfaces

3.2.1 Instance reporting interface

FUNCTION NAME		
FUSIONServiceInstanceReport (service instance → zone manager)		
BEHAVIOUR		
Update service instance data		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	The FUSION service name
ServiceInstanceID	String	The service instance identifier
ServiceData	Structured	Information about the running service (total and used session slots, histogram of service duration).

Locators	Structured	Set of locators (type, value)
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.2.2 Execution zone reporting interface

FUNCTION NAME		
FUSIONServiceUpdate (zone manager → zone gateway)		
BEHAVIOUR		
Update aggregate service instance data		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	The FUSION service name
ServiceData	Structured	Information about the aggregate running service within the execution zone (session slots, histogram of service duration, utility function, authentication data).
Locators	Structured	Set of locators (type, value)
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.2.3 Execution zone advertisement interface

FUNCTION NAME		
FUSIONRegisterService (zone gateway → service resolver)		
BEHAVIOUR		
Register a service with the local service resolver		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	The FUSION service name
ServiceData	Structured	Information about the running service (session slots, histogram of service duration, utility function, authentication data).
SendToSource	Flag	Flag to indicate resolution must be made by the zone gateway
TimeToLive	Long	Time in seconds this message is valid for. If not refreshed

		information should expire. Zero means to deregister the service.
Locators	Structured	Set of locators (type, value)
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONUpdateService (zone gateway → service resolver)		
BEHAVIOUR		
Update service parameters with the local service resolver		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	The FUSION service name
ServiceData	Structured	Information about the running service (session slots, histogram of service duration, utility function, authentication data).
SendToSource	Flag	Flag to indicate resolution must be made by the zone gateway
TimeToLive	Long	Time in seconds this message is valid for. If not refreshed information should expire. Zero means to deregister the service.
Locators	Structured	Set of locators (type, value)
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.2.4 Interdomain advertisement interface

FUNCTION NAME		
FUSIONSubscribeToServiceData (service resolver → service resolver)		
BEHAVIOUR		
Subscribe to information from a remote service resolution domain		
PROPERTIES & PARAMETERS		
Name	Type	Description
SubscriptionID	Long	Identifier of the subscription
RequestedData	Structured	Type of information subscribed (service catalogue, serviceID).

TimeToLive	Long	Time in seconds the subscription is valid.
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONServiceData (service resolver → service resolver)		
BEHAVIOUR		
Update service parameters with remote service resolver		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	The FUSION service name
ServiceData	Structured	Aggregate information about the running service (session slots, histogram of service duration, utility function, authentication data).
SendToSource	Flag	Flag to indicate resolution must be made by remote resolution domain
TimeToLive	Long	Time in seconds this message is valid for. If not refreshed information should expire. Zero means to deregister the service.
Locators	Structured	Set of locators (type, value)
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.2.5 Forwarding table configuration interface

FUNCTION NAME		
FUSIONCreateForwardingEntry (service resolver, routing function → forwarding function)		
BEHAVIOUR		
Forwarding entries are created according to the routing decisions made by the resolution management functions.		
PROPERTIES & PARAMETERS		
Name	Type	Description
EntryID	EntryID	Forwarding table entry identifier

Information	Structured	Forwarding entry, containing ServiceID(s), load balancing parameters, locator(s) of next hop
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONUpdateForwardingEntry (service resolver, routing function → forwarding function)		
BEHAVIOUR		
Forwarding entries are updated according to the routing decisions made by the resolution management functions.		
PROPERTIES & PARAMETERS		
Name	Type	Description
EntryID	EntryID	Forwarding table entry identifier
Information	Structured	Forwarding entry, containing ServiceID(s), load balancing parameters, locator(s) of next hop
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONDeleteForwardingEntry (service resolver, routing function → forwarding function)		
BEHAVIOUR		
Forwarding entries are deleted according to the routing decisions made by the resolution management functions.		
PROPERTIES & PARAMETERS		
Name	Type	Description
EntryID	EntryID	Forwarding table entry identifier
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.3 Monitoring Interfaces

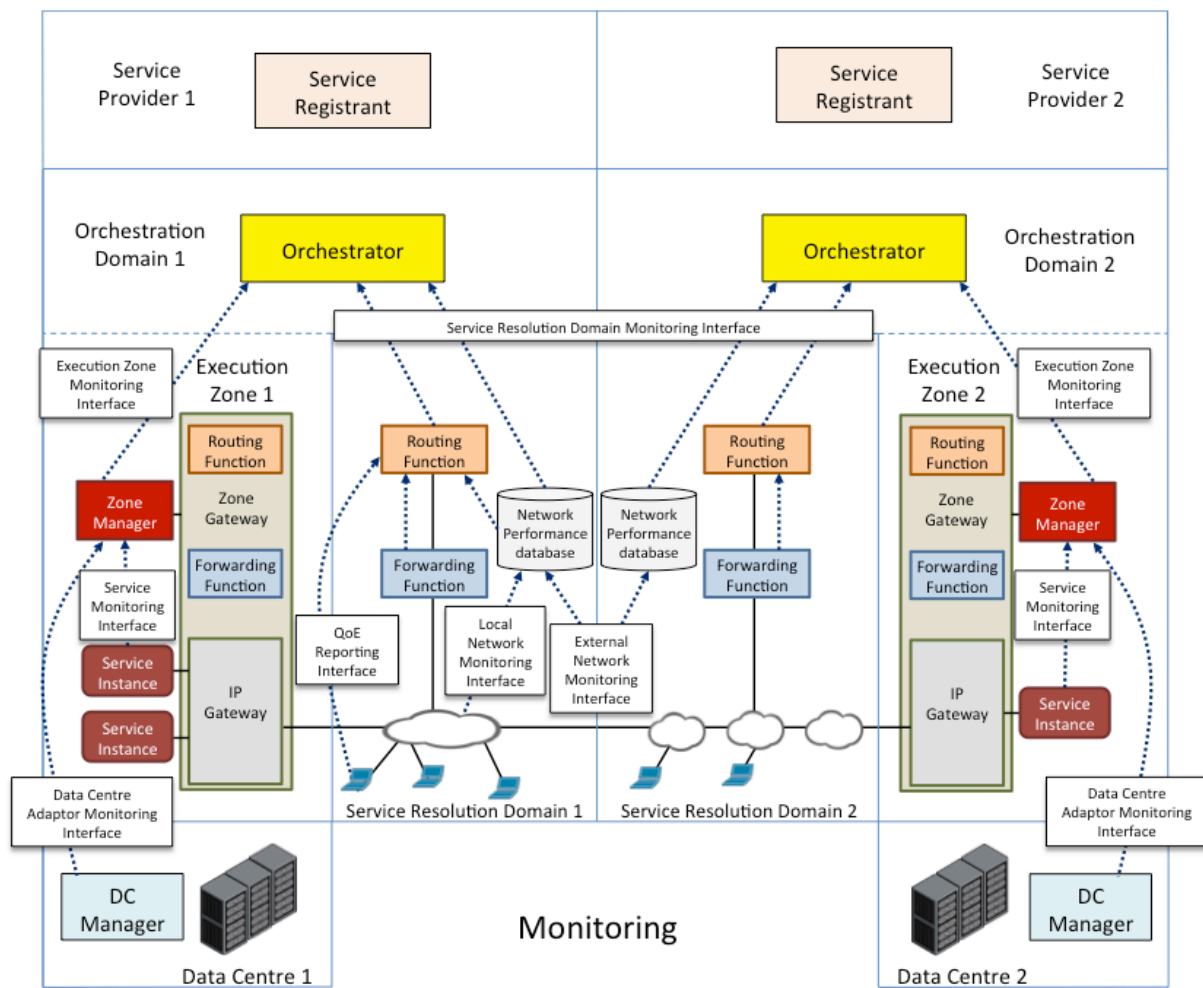


Figure 13: Monitoring interfaces

3.3.1 Local and external network monitoring interfaces

Service resolution, placement and deployment may require network information coming from the underlying IP infrastructure. Network related metrics that are currently considered for the application in FUSION are latency, jitter, bandwidth, and in specific cases topological information about the network.

Although FUSION does not impose any particular set of network monitoring tools to be used, given below is an abstract overview of the key function for gathering IP network level information. Given below is an overview of the key function for gathering service level information for resolution purposes by the resolution management function. Note this specification is conceptual and different renderings can be used in practice.

FUNCTION NAME		
FUSIONGetIPNetworkInformation (service resolver → non-FUSION-specific network monitoring system)		
BEHAVIOUR		
IP network element provides monitoring data concerning network performance statistics or and topological information collected from the perspective of that element (pull mode). This information may be aggregated and/or filtered.		
PROPERTIES & PARAMETERS		
Name	Type	Description
IProuterID	IPElementID	Identifier of IP network element (e.g., loopback address)
Information	Structured	Network monitoring information or topological information
RETURN VALUES		
Name	Type	Description
Information	Structured	The requested information

FUNCTION NAME		
FUSIONUpdateIPNetworkInformation (service resolver → non-FUSION-specific network monitoring system)		
BEHAVIOUR		
IP routers provide monitoring data to the resolution management function concerning historical network performance statistics and topological information (push mode).		
PROPERTIES & PARAMETERS		
Name	Type	Description
IPRouterID	IPElementID	Identifier of IP network element (e.g., loopback address)
Information	Structured	Network monitoring information or topological information
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.3.2 Service resolution domain monitoring interface

FUNCTION NAME		
FUSIONGetServiceResolutionInformation (domain orchestration → service resolver)		
BEHAVIOUR		
On orchestrator request, retrieve aggregated service resolution monitoring information for a particular service type or service instance (pull mode).		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceName	FUSIONID	Name of the service or service instance
Category	Enum	Specific subset of information (filter)
RETURN VALUES		
Name	Type	Description
Information	Structured	The requested information

3.3.3 Execution zone monitoring interface

FUNCTION NAME		
FUSIONGetServiceInformation (domain orchestration → zone manager)		
BEHAVIOUR		
Retrieve aggregated monitoring information for a particular service type (pull mode).		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceName	FUSIONID	Name of the service
Category	Enum	Specific subset of information
RETURN VALUES		
Name	Type	Description
Information	Structured	The requested information

FUNCTION NAME		
FUSIONGetZoneInformation (domain orchestration → zone manager)		
BEHAVIOUR		
Retrieve aggregated available information about the execution zone (pull mode).		
PROPERTIES & PARAMETERS		
Name	Type	Description
Category	Enum	Specific subset of information
RETURN VALUES		
Name	Type	Description
Information	Structured	The requested information

FUNCTION NAME		
FUSIONUpdateServiceInformation (zone manager → domain orchestration)		
BEHAVIOUR		
With this function, an execution zone provides the orchestration domain with up-to-date information concerning FUSION services running in the execution zone (push mode).		
PROPERTIES & PARAMETERS		
Name	Type	Description
ZoneID	ZoneID	Execution zone identifier
Information	Structured	Service monitoring information
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

FUNCTION NAME		
FUSIONUpdateResourceInformation (zone manager → domain orchestration)		
BEHAVIOUR		
With this function, an execution zone provides the orchestration domain with up-to-date information concerning the available resources and overall health of the execution zone (push mode).		
PROPERTIES & PARAMETERS		
Name	Type	Description
ZoneID	ZoneID	Execution zone identifier

Information	Structured	Resource monitoring information
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.3.4 Service monitoring interface

FUNCTION NAME		
FUSIONGetServiceInformation (zone manager → service instance)		
BEHAVIOUR		
Retrieve information for a particular service instance.		
PROPERTIES & PARAMETERS		
Name	Type	Description
Category	Enum	Specific subset of information
RETURN VALUES		
Name	Type	Description
Information	Structured	The requested information

3.3.5 QoE reporting interface

FUNCTION NAME		
FUSIONQoEReport (client → service resolver)		
BEHAVIOUR		
Clients report on quality of experience for a service invocation		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	Identifier of the service.
SourceID	Locator	Locator of the client
DestinationID	Locator	Locator of the invoked service instance
QoE observations	Structured	Set of observations (type, value) pairs
RETURN VALUES		
Name	Type	Description
Status	Int	Return code

3.3.6 Data-centre adaptor monitoring interface

FUNCTION NAME		
FUSIONDCGetInformation (zone manager → DC manager)		
BEHAVIOUR		
Request specific monitoring information for the underlying data centre.		
PROPERTIES & PARAMETERS		
Name	Type	Description
Category	Enum	Specific subset of information
RETURN VALUES		
Name	Type	Description
Status	Int	Return code
Information	Structured	The requested information

FUNCTION NAME		
FUSIONDCGetEnvironmentInformation (zone manager → DC manager)		
BEHAVIOUR		
Request specific monitoring information for a particular service environment.		
PROPERTIES & PARAMETERS		
Name	Type	Description
EnvironmentID	EnvironmentID	Reference to the environment
Category	Enum	Specific subset of information
RETURN VALUES		
Name	Type	Description
Status	Int	Return code
Information	Structured	The requested information

3.4 Service Query/Invocation Interfaces

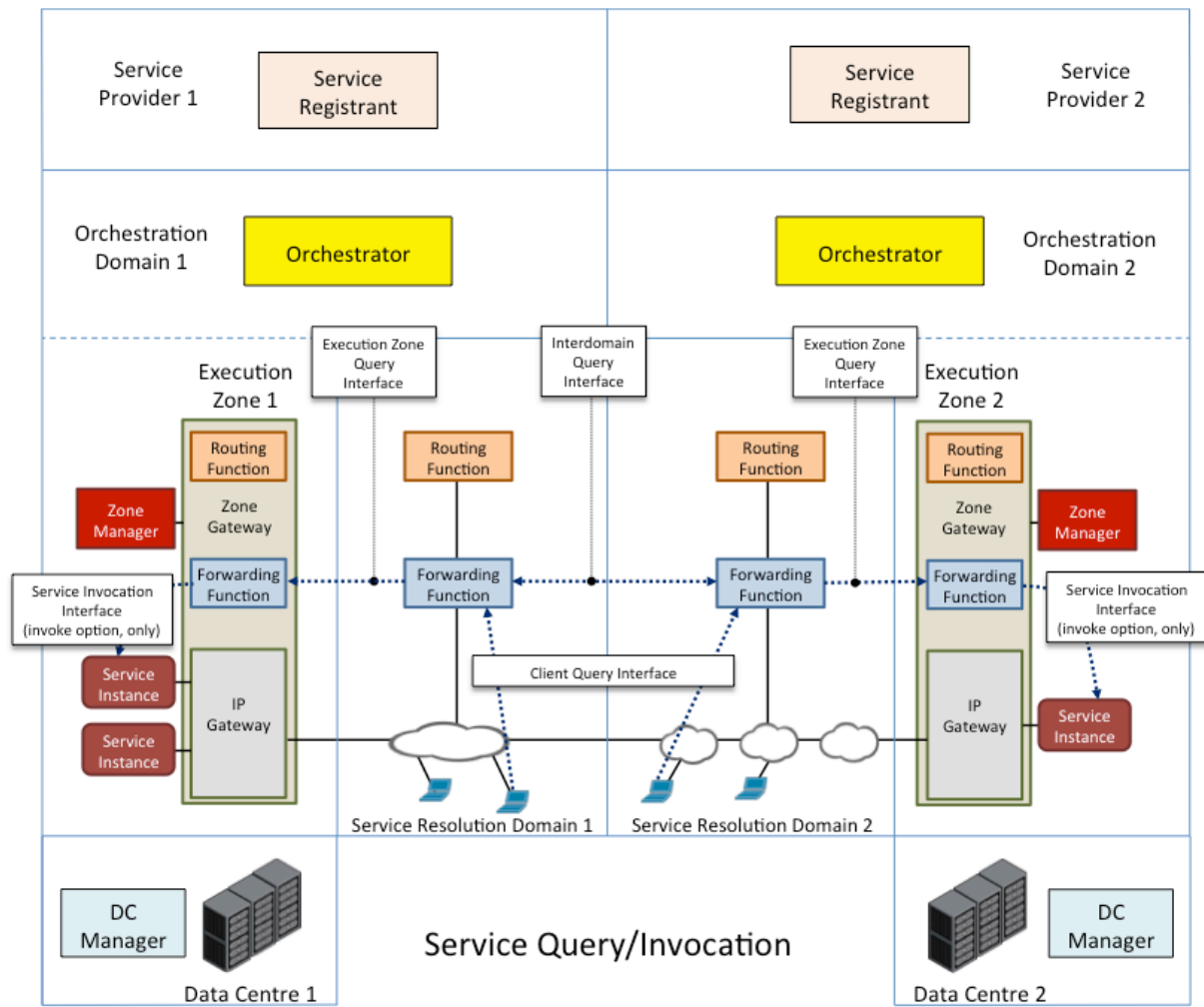


Figure 14: Service query/invocation interfaces

3.4.1 Client query/invocation interface

FUNCTION NAME		
QueryInstance (client->service resolver)		
BEHAVIOUR		
Resolve requested service identifier to instance locator		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	Identifier of the service.
Query Time to Live	long	Maximum time allowed to keep searching.
More than one	flag	Flag to signal whether more than one instance should be returned.
RETURN VALUES		
Name	Type	Description
RequestID	long	Identifier for the request.

FUNCTION NAME		
ReturnData (service resolver->client)		
BEHAVIOUR		
Return resolved service instance(s)		
PROPERTIES & PARAMETERS		
Name	Type	Description
RequestID	String	Identifier of the request.
Number of locators	Int	Number of locators being returned.
Locators	Structured	Set of locators (type of locator and locator value)

FUNCTION NAME		
InvokeInstance (client->service resolver)		
BEHAVIOUR		
Request the service resolution plane to select an instance and invoke the service with the conveyed payload data.		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	Identifier of the service.
Query Time to Live	Long	Maximum time allowed until reply is received
Payload	String	Payload to be delivered to the instance [application-specific]
RETURN VALUES		
Name	Type	Description
RequestID	long	Identifier for the request

FUNCTION NAME		
ReturnInvocationData (service resolver->client)		
BEHAVIOUR		
Return result of service invocation		
PROPERTIES & PARAMETERS		
Name	Type	Description
RequestID	String	Identifier of the request.
Payload	String	Payload of service invocation response [application-specific]

3.4.2 Interdomain query/invocation interface

FUNCTION NAME		
QueryInstance (service resolver->service resolver)		
BEHAVIOUR		
Resolve requested service identifier to instance locator		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	Identifier of the service.
Query Time to Live	Long	Maximum time allowed to keep searching.
More than one	Flag	Flag to signal whether more than one instance should be returned.
RETURN VALUES		
Name	Type	Description
RequestID	Long	Identifier for the request.

FUNCTION NAME		
ReturnData (service resolver->service resolver)		
BEHAVIOUR		
Return resolved service instance(s)		
PROPERTIES & PARAMETERS		
Name	Type	Description
RequestID	String	Identifier of the request.
Number of locators	Int	Number of locators being returned.
Locators	Structured	Set of locators (type of locator and locator value)

FUNCTION NAME		
InvokeInstance (service resolver->service resolver)		
BEHAVIOUR		
Request the service resolution plane to select an instance and invoke the service with the conveyed payload data.		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	Identifier of the service.
Query Time to Live	Long	Maximum time allowed until reply is received
Payload	String	Payload to be delivered to the instance [application-specific]
RETURN VALUES		
Name	Type	Description
RequestID	Long	Identifier for the request

FUNCTION NAME		
ReturnInvocationData (service resolver->service resolver)		
BEHAVIOUR		
Return result of service invocation		
PROPERTIES & PARAMETERS		
Name	Type	Description
RequestID	String	Identifier of the request.
Payload	String	Payload of service invocation response [application-specific]

3.4.3 Execution zone query/invocation interface

FUNCTION NAME		
QueryInstance (service resolver->zone gateway)		
BEHAVIOUR		
Resolve requested service identifier to instance locator		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	Identifier of the service.
Query Time to Live	Long	Maximum time allowed to keep searching.
More than one	Flag	Flag to signal whether more than one instance should be returned.
RETURN VALUES		
Name	Type	Description
RequestID	Long	Identifier for the request.

FUNCTION NAME		
ReturnData (zone gateway ->service resolver)		
BEHAVIOUR		
Return resolved service instance(s)		
PROPERTIES & PARAMETERS		
Name	Type	Description
RequestID	String	Identifier of the request.
Number of locators	Int	Number of locators being returned.
Locators	Structured	Set of locators (type of locator and locator value)

FUNCTION NAME		
InvokeInstance (service resolver-> zone gateway)		
BEHAVIOUR		
Request the zone gateway to select an instance and invoke the service with the conveyed payload data.		
PROPERTIES & PARAMETERS		
Name	Type	Description
ServiceID	FUSIONID	Identifier of the service.
Query Time to Live	Long	Maximum time allowed until reply is received
Payload	String	Payload to be delivered to the instance [application-specific]
RETURN VALUES		
Name	Type	Description
RequestID	Long	Identifier for the request

FUNCTION NAME		
ReturnInvocationData (zone gateway ->service resolver)		
BEHAVIOUR		
Return result of service invocation		
PROPERTIES & PARAMETERS		
Name	Type	Description
RequestID	String	Identifier of the request.
Payload	String	Payload of service invocation response [application-specific]

3.5 Service Delivery Interfaces

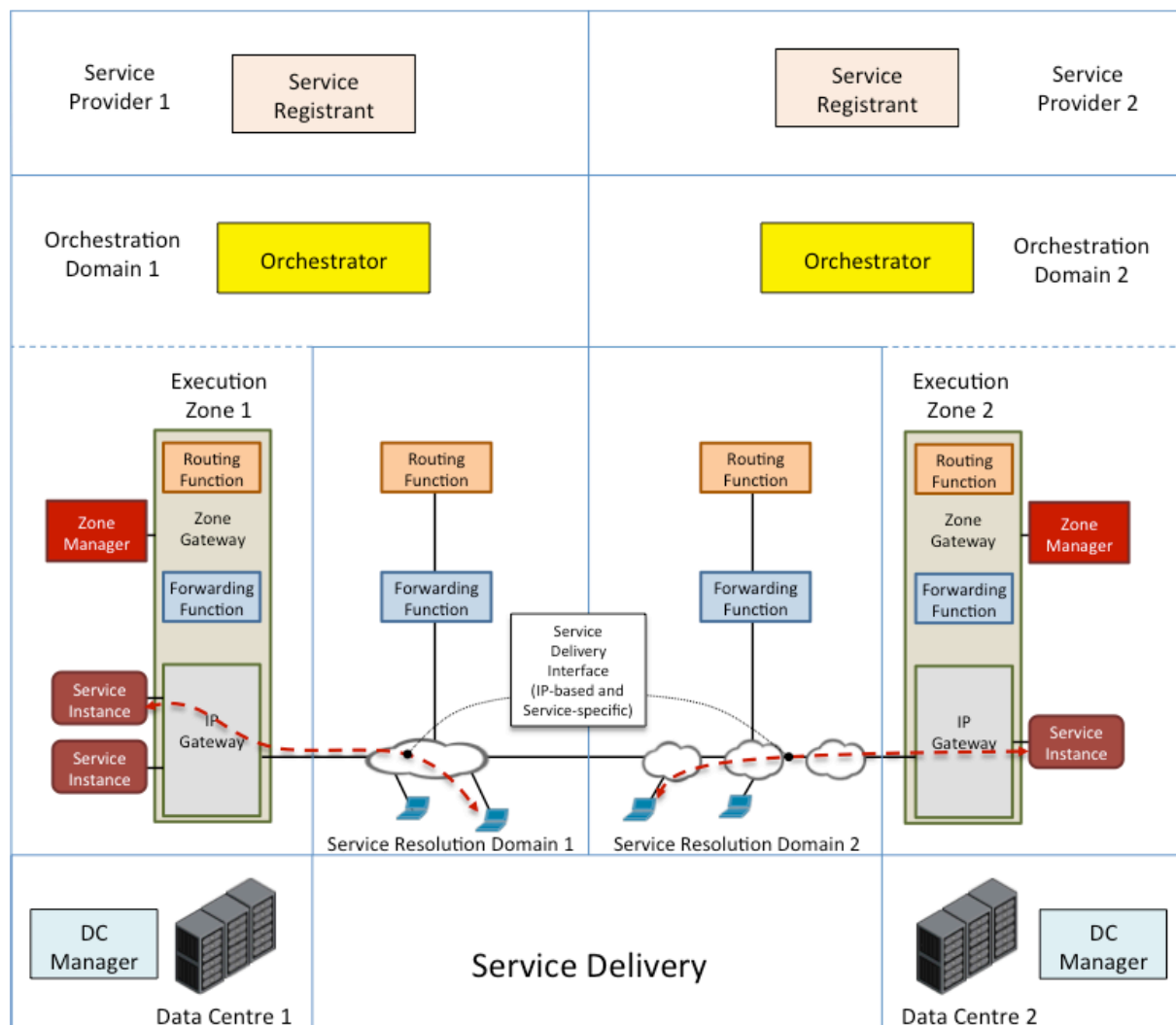


Figure 15: Service delivery interface

3.5.1 Service delivery/data-plane interface

The service data-plane interfaces are service-specific and not under the control of FUSION.

4. INTERFACE TECHNOLOGIES

4.1 Orchestration/zone management

Both for the intra-zone, intra-domain as well as in between a domain orchestrator and a zone manager, we intend to implement the interfaces via REST APIs in combination with JSON for a number of reasons:

- It is a proven light-weight protocol with a clear interface for querying and modifying resources;
- It facilitates running the various services in a distributed fashion inside a data centre or across the Internet;
- OpenStack and other orchestration and management systems successfully use REST for their inter-service management blocks;

- Due to its popularity, there are many frameworks available that allow for quick and efficient implementations, for example in the Python programming language;
- It allows for easy integration with an existing cloud management system like OpenStack;

As an example, we show how the service registration interface would look like using the REST/JSON API for a simple EPG service.

REST/JSON request:

PUT /provider/1.0/services/register

PUT Data {"manifest_uri": "http://exampleprovider.eu/epg/TOSCA.CSAR"}

REST/JSON reply:

```
{
  "status": 201,
  "service_id": "2c353530-a2e8-11e3-b2df-000c29c5da1c",
  "service_uri": "/provider/1.0/services/2c353530-a2e8-11e3-b2df-000c29c5da1c"
}
```

In this example, a service provider registers a new service type via the provider interface of the example fusion domain orchestrator using the 1.0 version of the REST/JSON API. Instead of uploading the TOSCA manifest archive directly, a reference to its location is passed along. The archive is expected to be fully contained and self-describing, containing all necessary information for the orchestration domain.

Upon successful registration, the domain responds with a status 201, the newly created unique service ID (here represented by a UUID), as well as the management endpoint in the FUSION domain where the service can be managed and updated (e.g., remove the instance, retrieve its monitoring data, update the manifest (e.g., its deployment and cost policies, etc.), etc.

4.2 Service resolution

A dedicated Networked Services Resolution Protocol for service, announcement, querying and resolution is being defined by the project. The specification details will be contained in deliverable D4.2.

4.3 DC specific

In case a FUSION execution zone is designed as a PaaS overlay on top of an existing data centre, the Data Centre Adaptor (DCA) zone management component needs to communicate with the underlying data centre management layer. As we assume that a FUSION execution zone may be deployed in different data centres using different APIs (cloud APIs, grid APIs, etc.), the FUSION functions need to be translated by the DCA into DC-specific API calls. For example, if the underlying DC supports the OpenStack APIs, the south-bound interface of the DCA can communicate with that DC using these APIs, either directly using the corresponding REST APIs of OpenStack, or e.g. by means of the corresponding Python client wrapper scripts that are available on many systems.

Below a simple example to deploy a new VM instance within OpenStack using the command-line tools:

```
$ nova boot myEPGservice --image "3afe97b2-26dc-49c5-a2cc-a2fc8d80c001" --flavor 6
```

The DCA is responsible for translating and handling all commands in between FUSION and the DCA, which needs a different backend driver for each DC management API on which a FUSION zone and corresponding FUSION services could be deployed.