

**SEVENTH FRAMEWORK PROGRAMME  
CAPACITIES**



**Research Infrastructures  
INFRA-2007-1.2.1 Research Infrastructures**

**DRIVER II**

**Grant Agreement 212147  
“Digital Repository Infrastructure Vision for European Research II”**



**The DRIVER Service Activities**

Deliverable Code: D5.4

## Document Information

### Project

Title:	DRIVER, Digital Repository Infrastructure Vision for European Research II
Start date:	1st December 2007
Call/Instrument:	INFRA-2007-1.2.1
Grant Agreement:	<b>212147</b>

### Document

Deliverable number:	D5.4
Deliverable title:	The DRIVER Service Activities
Contractual Date of Delivery:	1st of February 2009
Actual Date of Delivery:	
Editor(s):	CNR
Author(s):	Paolo Manghi, Franco Zoppi
Reviewer(s):	Wojtek Sylwestrzak
Participant(s):	
Workpackage:	WP5
Workpackage title:	Infrastructure Sustainability and Maintenance
Workpackage leader:	CNR
Workpackage participants:	ICM, NKUA, CNR, DTU, UMINHO, NUK, UON, UGENT, UNIBI
Distribution:	Confidential
Nature:	Deliverable
Version/Revision:	1.5
Draft/Final:	Draft
Total number of pages: (including cover)	
File name:	The DRIVER Service Infrastructure (DSI) V1.5.doc
Key words:	Service, Monitoring, Sustainability, Policies

## Disclaimer

This document contains description of the DRIVER-II project findings, work and products. Certain parts of it might be under partner Intellectual Property Right (IPR) rules so, prior to using its content please contact the consortium head for approval.

In case you believe that this document harms in any way IPR held by you as a person or as a representative of an entity, please do notify us immediately.

The authors of this document have taken any available measure in order for its content to be accurate, consistent and lawful. However, neither the project consortium as a whole nor the individual partners that implicitly or explicitly participated in the creation and publication of this document hold any sort of responsibility that might occur as a result of using its content.

This publication has been produced with the assistance of the European Union. The content of this publication is the sole responsibility of DRIVER-II consortium and can in no way be taken to reflect the views of the European Union.

The European Union is established in accordance with the Treaty on European Union (Maastricht). It is based on the European Communities and the member states cooperation in the fields of Common Foreign and Security Policy and Justice and Home Affairs. The five main institutions of the European Union are the European Parliament, the Council of Ministers, the European Commission, the Court of Justice and the Court of Auditors. (<http://europa.eu.int/>)



**DRIVER-II is a project partially funded by the European Union**

# Table of Contents

Document Information.....	2
Disclaimer.....	3
Table of Contents .....	4
Figures.....	5
Tables .....	6
Log of Changes .....	7
Summary.....	8
1 Introduction.....	9
1.1 Purpose and Intent .....	9
1.2 Terms and Abbreviations .....	10
2 The DRIVER Service Infrastructure Organization.....	12
2.1 Summary view .....	12
2.2 DRIVER Resource Model .....	13
2.3 Resource Management Actions.....	15
2.3.1 External requests handling .....	15
2.3.2 Infrastructure maintenance .....	16
2.4 Administrator Users: Roles and Responsibilities.....	17
3 The DRIVER Service Infrastructure Operation.....	20
3.1 Actions and procedures .....	20
3.1.1 External requests handling .....	20
3.1.1.1 Add an Application .....	20
3.1.1.2 Add a Site .....	20
3.1.1.3 Add a Node .....	21
3.1.1.4 Add a new service resource .....	21
3.1.1.5 Add a new Software release of a service resource .....	21
3.1.1.6 Add a new resource typology .....	21
3.1.1.7 Add a Data Source.....	22
3.1.1.8 Access an Application, a Service or a Data Source .....	22
3.1.2 Infrastructure maintenance .....	22
3.2 Service Level Agreement and Quality of Service in DSI .....	23
3.3 Instruments.....	27
4 References .....	29

## Figures

Figure 1 – The DRIVER Service Infrastructure architecture .....	13
---	----

## Tables

Table 1 – Terms and Abbreviations .....	11
Table 2 – Responsibilities (Actions vs. Roles) .....	19
Table 3 – Resource Functionality .....	23
Table 4 – Resource Reliability, Availability and Serviceability (RAS) .....	24
Table 5 – Resource Usability .....	24
Table 6 – Resource Efficiency .....	24
Table 7 – Resource Maintainability .....	24
Table 8 – Resource Portability .....	25
Table 9 – User Support .....	25
Table 10 – Operational Support .....	25
Table 11 – Data Source QoS .....	26

## Log of Changes

<b>Deliverable N° &amp; Version N°</b>	<b>Date (dd/mm/yyyy)</b>	<b>Changes</b>
D5.4 V0.3	18/06/2008	First draft sent to Manghi for review.
D5.4 V0.4	27/06/2008	Review of concepts and terminology (partial).
D5.4 V0.5	11/07/2008	Reworked according to Manghi's feedback.
D5.4 V0.6	18/08/2008	Additional re-organisation by Zoppi – Sent to Manghi for review.
D5.4 V0.7	10/10/2008	Reworked according to Manghi's feedback.
D5.4 V0.8	16/10/2008	Further additions by Zoppi. Some contents moved to DCC. Comments from ICM included.
D5.4 V1.0	25/11/2008	First issue released for Review to ICM.
D5.4 V1.3	05/01/2009	Version reviewed and edited by Manghi
D5.4 V1.4	09/01/2009	Version reworked by Zoppi
D5.4 V1.5	12/01/2009	Version reworked by Manghi

## Summary

The goal of this document is to define the methodologies driving the DRIVER Service Activities (DSA), whose aim is to structure and organize the activities of the Responsible Organization (RO) administering the DRIVER Service Infrastructure (DSI) so as to support the Participating Organizations (POs), i.e. *data providers* and *service providers*, willing to join or use the infrastructure run-time environment to serve the needs of their communities of *end-users*.

# 1 Introduction

## 1.1 Purpose and Intent

The DRIVER Service Infrastructure (DSI) is administered by one Responsible Organization (RO) and offers the tools for building and maintaining multiple, coexisting service-based applications for Participating Organizations (POs). The DSI application framework gives life to a run-time environment whose actors and participation rules are described in this document. In particular, PO can be *data providers*, who are interested in sharing and contributing their valuable content resources to the DSI, or *service providers*, which are interested in building so called PO applications, i.e. DRIVER applications obtained by combining available or newly deployed DRIVER resources.

The DSA classifies the RO and PO *users*, the *actions* the RO have to perform to support POs and the *rules* and *policies* both should obey for the DSI to achieve optimal *Quality-of-Service* (QoS). In particular, this document focuses on two main aspects:

- The definition of the *DSI organizational model* in terms of PO applications, the resource categories involved in application construction, the actions and related procedures that can be executed over such resources, and user roles w.r.t. the execution of such actions.
- The definition of the *DSI operation policies* in terms of the administrative activities and instruments needed to support the integration of new POs and to guarantee infrastructure QoS over time.

The operation of DSA is carried out by a number of RO *administrators*, who collaborate with PO *administrators* to ensure PO satisfaction and DSI quality QoS. RO administrators exchange information and report about DSI organization and operation through a dedicated MediaWiki website [6], namely *DSAWiki*, which gathers all relevant information related with:

- DSI policies, e.g. PO supervision and support, resource management actions and procedures, users involved per actions;
- DSI status , e.g. available resources per PO, resource current status, PO users involved, PO participation agreement;
- DSI administration history, e.g. log of management operations.

The website is expected to evolve during the project lifetime so as to reflect the progresses of DSI maintenance. While this document illustrates DSA main concepts, the website, due to its role in the project, represents the actual ongoing deliverable.

To summarize, this document is organized as follows: Section 2 describes the *organizational structure* of the DSI, while Section 3 focuses on the operation of the DSI introducing the *resources and actions* of the DSI, along with the respective administrator user *roles*, and the *instruments* provided to operate the DSI.

## 1.2 Terms and Abbreviations

Term / Abbreviation	Meaning
DCC	DRIVER Control Centre An administrative application, providing utilities needed by RO administrators to support POs at performing their operations according to the assigned SLA.
DRP	DRIVER Release Plan A structured description of the resources and actions required to satisfy the objectives described in a PO DTD request.
DSA	DRIVER Service Activities The organization and the operation of the activities to be performed by RO and PO administrators in order to maintain an active DSI.
DSI	DRIVER Service Infrastructure The running environment of the DRIVER Infrastructure.
DSAWiki	A website which gathers all relevant information related to the management of the DSI; e.g. users involved, policies established, resources allocated. It contains also the details of the procedures needed to operate the DSI according to a defined QoS.
DTD	DRIVER Target Description A structured application form to ease POs at describing their needs and identify the proper solution in the DRIVER Infrastructure. As a result of the DTD, a POs is assigned a DRP or a DUP and a relative SLA.
DUP	DRIVER Usage Profile A structured description of a DRIVER resource typology, intended to describe its purposes and uses. Resources, to belong to a certain resource typology, must match the DUP prescriptions.
PO	Participating Organization An organization willing to act as DSI resource provider and/or consumer to satisfy the application needs of the PO end-users.
QoS	Quality of Service A general, structured description of the level of service that can be offered by the DRIVER Infrastructure, based on a number of quality parameters.
RO	Responsible Organization The organization in charge of the administration of the DSI running environment, that is supporting POs at their activities and continuously monitoring the DSI so as validate the POs' SLA.
SLA	Service Level Agreement A statement of the specific level of QoS to be supplied by a PO willing to join the DRIVER Infrastructure. In the same statement, the RO must guarantee an optimal DSI QoS for the core services, i.e. Enabling Services, so as to ensure operation of all PO applications.



## Table 1 – Terms and Abbreviations

## 2 The DRIVER Service Infrastructure Organization

This section gives an overall description of the architecture of the DSI, of the *resources* managed by the DSI, of the actions implemented to support the operation of the DSI according to a defined QoS, and of the roles and responsibilities assigned to the different actors in charge of the execution of these actions.

### 2.1 Summary view

Infrastructure systems are maintained by one responsible organization (RO), which provides support to other participating organizations (POs) willing to provide and integrate their resources and constructing their applications with such resources. In particular, resources may belong to the following categories: system resources (e.g. sites, basic software, nodes), functional resources (e.g. services) and content resources (e.g. data sources, in turn managed by special functional resources).

The DRIVER Infrastructure adopts the component-oriented approach of Service Oriented Architectures (SOA) [7], where services are implemented as Web Services [8]. In SOA, applications consist of a set of distributed services that interact to deliver the functionalities expected by their users. Services support specific sets of functionalities in isolation and can be combined into workflows of actions to model arbitrary complex data computation processes. Most importantly, Services can be shared between different applications coexisting in the same SOA.

The DRIVER infrastructure enriches the principles of SOA with the notion of “orchestration”. More specifically, an application is not defined as a predetermined set of Services, but as a “declaration” of all service functionalities needed for the application to work. At run-time, infrastructure orchestration mechanisms will form PO applications by enabling and guiding the interaction between those Services that are sharable and match the application declaration constraints. POs can therefore contribute in augmenting the quality-of-services of all running applications by offering new hardware and deploying DRIVER service instances; the orchestration framework will automatically exploit the new resources by reusing them in the context of needing applications.

The infrastructure framework governs all applications by means of special *Enabling Services*, available 24/7 and administered by the infrastructure RO. All services need to register to the *Information Service* their *profile*, i.e. compound of information about their (web) location, the functionality they expose and their current status. Services must continuously update their profile to the latest status to the Information Service, so that the latter becomes the keeper of the “infrastructure resource map”. Service orchestration is delegated to other Enabling Services, named *Manager Services*. Such Services are entitled of the execution of actions, as a reaction to the occurrence of certain events. The Manager Service orchestrates services by dynamically discovering through the Information Service where the services it requires are located and by combining their interaction.

As shown in Figure 1, we can distinguish between data source resources, which are integrated through special services into the DSI so as to exploit their content, *functional resources*, which are services running at some DRIVER *node* in turn operative at some DRIVER *site*, and *applications*, which are dynamic combinations of such services, automatically orchestrated by a dedicated Manager Service. Application resources are provided by POs and can be consumed by PO end-users or by PO external applications, which can access the DRIVER web services through the standard web service access protocols; e.g. the OAI-PMH Publisher Service can be accessed by harvesting applications.

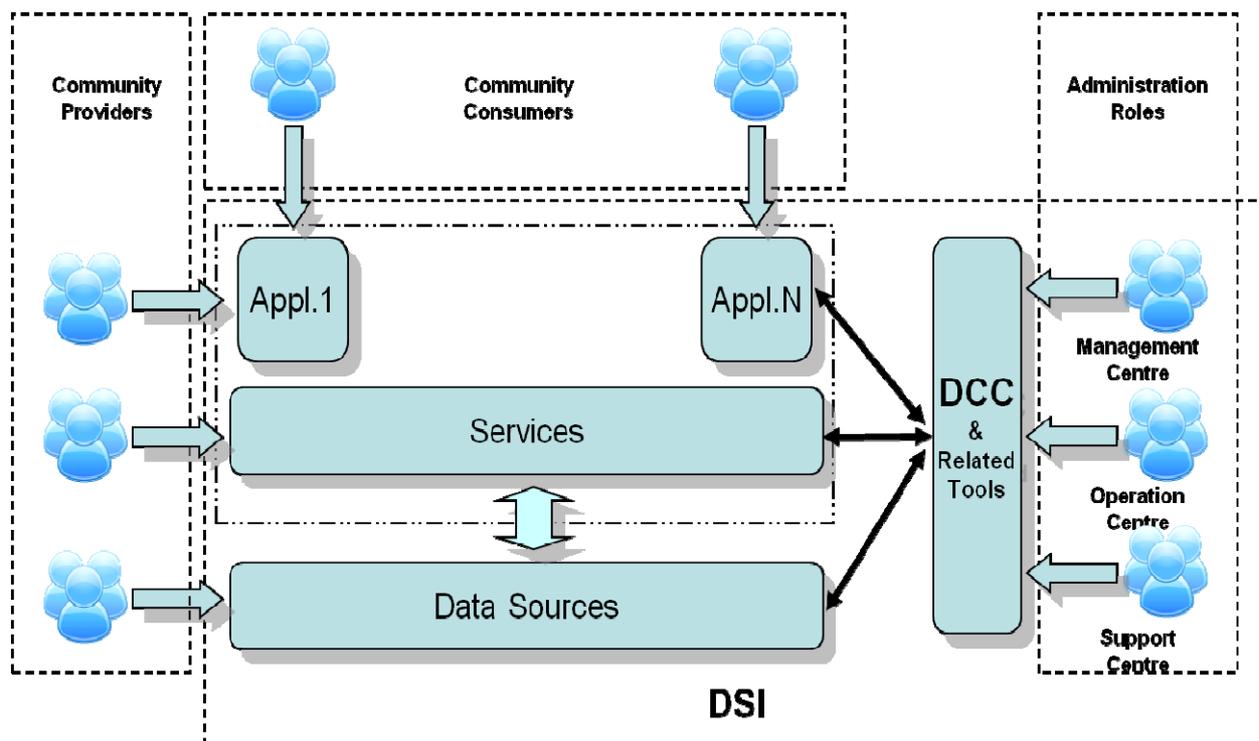


Figure 1 – The DRIVER Service Infrastructure architecture and organization

## 2.2 DRIVER Resource Model

DRIVER resources, of all kinds, are implemented as Web Services or as special entities managed by such services. In particular, the DRIVER resource model differentiates between *resource typologies* and *resource instances* (simply *resources*), where resources are entities *available* to the DIS, i.e. present and usable in the running environment, which *conform* to one specific resource typology. Resource typologies describe “classes” of homogenous resources. Such definition is formalized by means of a *DRIVER Usage Profile* (DUP), to be released and updated by the RO Management Centre. A DUP provides the following information about the resources of the given typology:

- Registration: *resource profile* schema definition, to be introduced below, and registration methods (i.e. manual or automatic);
- Deployment: *actions* required to make a resource of this typology available to the DIS running environment;
- Usage (access): if it is a functional resource typology, i.e. a service, usage is given by means of the relative web service API, together with the informal semantics of the methods; if it is a content resource typology, the API of the service required to manage it and what operations can be performed, through the service, over this typology of resources;
- Implementation: many resource implementations may exist for the same resource typology; as for usage, service APIs and informal semantics provide a contract for resource developers or providers willing to deliver a new resource software release.
- Quality of Service: the minimal *expected SLA* (to be defined in the next section) of the resource, i.e. the minimal QoS thresholds to be supported by any resource of this typology to be validated and certificated.

Resource typologies can be added any time to the DIS, together with the relative DUP, and after *validation* and *certification* from the RO Manager Centre.

Any resource available to the DSI run-time environment has to match the relative DUP requirements so as to guarantee the expected functionality and content usage under the expected QoS. In particular, the actions applicable to a resource are:

- Validation: prior integration into the DSI running environment, resources must be *validated* against their DUP;
- Certification: a validated resource can be *certified*, i.e. integrated into the DSI;
- Registration: to be available for usage by other resources, resources must *register* their *profile* to the Information Service, i.e. the DSI registry. Profiles are formal, machine-processable resource descriptions (i.e. XML files in the DSI) of the resources defined by the relative DUP, that contain information on their unique identity, typology, location, function, configuration and current status. By querying the Information Service, consuming services can search and discover, among the available resources, the resources they need for their operation.
- Quality of Service agreement: to be deployed into the DSI, resources should declare a *SLA* (defined in the Section 3), i.e. the QoS the PO is willing to guarantee for that resource.
- Monitoring: resources should be monitored to check their actual SLA, i.e. the current QoS, in order to match it against the resource *SLA* declared at registration time.

More in detail, the DSI resource model offers the following resource kinds:

### System resources

- *Site*: a web accessible physical machine, thus a combination of network, hardware and software resources capable of hosting a DRIVER *Node*.
- *Node*: a set of Service(s) sharing a local "communication layer", i.e. that do not need to communicate through network web service interfaces but can rely on efficient API interaction.

### Functional resources

- *Service*: an active DRIVER service, i.e. a piece of software that implements DSI participation rules (e.g. registration to the Information Service, implementation of subscription and notification APIs) and is installed and running at some DRIVER node.
- *Data structure*: resources managed (i.e. created, updated, operated over, deleted) by services.

Important instances are:

- ◆ *Application*: the declaration of a DRIVER application, together with its configuration, to be managed by Manager Services.
- ◆ *User*: any of the PO and RO individuals in the need of authenticated and/or authorized interaction with the DSI services.

### Content resources

- *Data Sources*: data sources are collections of information objects owned by POs and integrated into the DSI through appropriate DRIVER services by means of standard or proprietary access protocols. For example Institutional Repository data sources expose through OAI-PMH protocols their metadata records, which are imported into the DSI through DRIVER OAI-PMH Harvesting Services.

### User resources

The DSI main *actors* are the RO and the POs, where POs can be of two kinds:

- *Consumers*: consumers are POs whose end-users use PO applications or whose developers devise external (non-DRIVER) applications that need to interact with

DRIVER services; e.g. OAI-PMH harvesters that interact with DRIVER OAI-PMH Publisher Services.

- *Providers*: providers are POs owning resources of the kinds mentioned above and willing to integrate/share them within the DSI and/or willing to build PO applications within the DSI.

Users are individuals “belonging” to the RO or to the POs, with special usage rights over the DSI resources. Users are DSI resources, in particular *data structure resources* to be managed by User Services. Users can be of two kinds:

- *Administrators*: we shall differentiate between PO and RO administrators. The former are PO individuals in charge of monitoring, controlling and maintaining the PO applications; the latter are the RO individuals in charge of monitoring, controlling and maintaining the DSI environment.
- *End-users*: end-users are the PO individuals whose needs are satisfied by the PO by acting as consumer and/or as provider.

## 2.3 Resource Management Actions

### 2.3.1 External requests handling

Consumer and provider POs can benefit from a DSI instantiation in a variety of ways. More specifically, their application needs may be satisfied by a number of properly “orchestrated” DSI resources. On the other hand, what resources are required, how to find, construct or deploy them, as well as the associated orchestration scheme, are all aspects not necessarily familiar to the POs. In order to help POs at formulating the proper enquiry and be instructed on how to achieve their applicative expectations, the RO Management Centre provides PO administrators with a web form (DRIVER Control Centre instrument, DCC) guiding POs at specifying a structured and high-level description of their applicative needs, called *DRIVER Target Description* (DTD). DTDs are devised so as to be interpreted by the RO to produce:

- a *DRIVER Release Plan* (DRP), i.e. a description of the typology and number of *resources* needed to fulfil the application needs and the *actions* to be executed by both the PO and the RO to provide, if not available, and deploy such resources; and
- a number of resource *Service Level Agreement* (SLA), one for each resource involved in the DRP: SLAs are statements of commitment agreed between the RO and the PO, specifically devised to ensure both resources and DSI Quality-of-Service.

DTDs support POs at the identification of the resource typologies and relative usage they require; e.g. the addition of a specific application resource, or the addition of a content resource. While generic resources come with a well defined SLA, i.e. installations of service resources at a node, application resources require POs to specify the expected SLA. Application resource DRPs are calculated by the RO Management Centre in order to match the application resource SLA, by considering the DUPs of the resources required by the application, the DSI registered and reusable resources with their SLAs. The result is a list of actions whose execution from the RO and the PO optimizes the deployment of the application resource w.r.t. the resources available to the DIS and those that are still missing.

In the following, we summarize the individual actions that might result as part of DRPs obtained from PO DTDs, divided in providers and consumers.

#### Provider POs

POs can perform a variety of actions related with resource provision, which all end with the definition of the resource SLA.

- Add a site;

- Add a node to a site;
- Add a new service instance to a node (install/deploy);
- Add a new software release of a service resource (implement);
- Add a new typology of service (implement);
- Add an application;
- Add a data source.

For each of such actions, in order to support the PO in the setup, certification and operation activity, a DARP and the relative SLA will be provided to the PO by the RO. The DRP establishes an agreement between the PO and the RO, and describes the requirements and procedures they should both meet and accomplish to complete the action.

### **Consumer POs**

For each of the following, a DRIVER Usage Profile (DUP) will be provided to the PO by the RO:

- Access an application;
- Access a service;
- Access a data source.

### **2.3.2 Infrastructure maintenance**

For the implementation of maintenance *actions* refer to [1].

## 2.4 Administrator Users: Roles and Responsibilities

The DSI operation is organized in different *areas of work*, each of them involving different DSI resources, actions and users. Specifically the areas are:

- Resource Installation and Upgrade: the area deals with the installation and upgrade of the software implementing the functional resources, i.e. services, running in the infrastructure.
- Resources Compliance and Integration: the area deals with the compliance verification process that system, functional and content resources have to go through to be classified and integrated in the DSI.
- Resource Monitoring: the area deals with the tools used to monitor the status of any resource of the infrastructure.
- Resources Security: the area deals with the implementation of the security model of the infrastructure, to be applied to any resource.
- Usage and Operation Support: the area deals with the aspects of providing support to the PO administrators.

PO and RO *actors* have *administrator users*, who play different *roles* in the different areas. These roles are:

*Management Centre administrators (RO)* The Management Centre is responsible for the management and the operation of the DSI. The main tasks of the Management Centre are:

- Process *DTD requests from POs*, to produce the relative DRP and SLAs;
- For the integration of new resource typologies:
  - Definition of the relative *resource compliance test*;
  - Definition of the *resource installation, operation and support actions*;
- Validation of SLAs by *monitoring DSI resources* through the appropriate DSI instruments;
- When SLAs are not respected: *execution of repairing actions* on the RO side or *sending SLA violation notifications* to the PO Operation Centres;
- Plan service software *upgrades and deployment*;
- Collect requirements from Support Centres and Operation Centres so as to add new resource typologies.

*Support Centre administrators (RO)* The Support Centre is responsible for providing support to PO Operation Centre administrators executing DRP *actions*. The main tasks of the Support Centre are:

- Provide *support* for DSI issues:
  - Application framework principles: how DRIVER services operate and interact, how applications can be orchestrated by Manager Services, etc.;
  - Resource usage, installation/deployment.
- Gather new administration requirements for the improvement of Management Centre activities and instruments.

*Operation Centre administrators (PO)* All system, functional and content resources belonging to the same PO are under the responsibility of the relative Operation Centre. The main tasks of these administrators are:

- *Application of DTD requests* and formal commitment to the relative DRP and SLA;
- *Execution of DRP actions*:
  - In the case of integration of new resources into the DSI (e.g. new data source resource, new service instance, new service instance for new service typology, new service instance for service implementation), execution of resource *compliance test actions*;
- *Guarantee SLA conditions*:

- Accepting SLA violation notifications from the Management Centre and *execution of repairing actions*;
- Monitoring and administration of PO application resources; e.g. end-user management, data flows, workflows, etc.
- Gather administrator and end-user requirements for the improvement of Management Centre activities and instruments.

Different users are assigned different responsibilities over certain actions, depending on their role. Table 2 illustrates the resulting distribution, where:

- *Request, Plan and Execute*: the requested action, welcomed by the support centre, corresponds to a sequence of *procedures* to be planned by the Management Centre and executed by Operation Centre of the requesting PO.
- *Approve*: some action imply access to existing running services or the introduction of new services. In this case, the Management Centre needs to evaluate and approve the impact of the request so as to ensure DSI quality-of-service.
- *N/A*: the user involved has no rights to execute the relative action.

Action	Role				
	Management Centre	Operation Centre	Support Centre	PO Op. Centre as Provider	PO Op. Centre as Consumer
Create a site from scratch	Plan	Execute	Support	Request	N/A
Add a node to a site	Plan	Execute	Support	Request	N/A
Add a new service instance	Plan	Execute	Support	Request	N/A
Add a new software release of a service instance	Plan	Execute	Support	Request & Execute	N/A
Add a new typology of service	Approve & Plan	Execute	Training	Request & Execute	N/A
Add an application	Plan	Execute	Support	Request	N/A
Add a data source	Plan	Execute	Support	Request	N/A
Access a service	Approve	Execute	Support	N/A	Request
Access a data source	Approve	Execute	Support	N/A	Request

Access an application	Approve	Execute	Support	N/A	Request
-----------------------	---------	---------	---------	-----	---------

**Table 2 – Responsibilities (Actions vs. Roles)**

## 3 The DRIVER Service Infrastructure Operation

POs define DTDs as high-level descriptions of sequence of actions, namely a DRIVER Release Plan (DRP), over DSI resources. A is generated based on the DTD submitted by the interested PO to the Management Centre. such as addition or access of a resource, but are mostly defined to help POs at identifying the *DRIVER application template* that might best fulfil the needs of the PO communities. An application template identifies a specific application resource typology, i.e. a set of target functionalities together with the “typology” of resources and “typology” of actions required to deploy the application and support it. Application templates are defined and managed by the Management Centre, which can always add new ones or modify the existing ones; e.g. POs in the need of applications not yet described by any template can define and add a new application template to the DSI by interacting with the Support Centre.

Once the required application template is identified, the PO completes the DTD by supplying the application resource SLA, i.e. the QoS parameters the application is supposed to offer. The information available from the complete DTD, i.e. application template and SLA, is used by the Management Centre to produce the application resource DRP and the SLAs of the individual resources required by the application. In particular, RO Management Centre administrators are in charge of:

- Interact with POs in the execution of the DRP *actions* required to accomplish their DSI participation requests; and
- Monitoring the DSI running environment so as to control the validity of SLAs and ensure the expected QoS.

In the following, *actions* are described in detail as procedures, while the notion of SLAs and QoS are presented, together with the *instruments* specifically devised to ease the Management Centre activities.

### 3.1 Actions and procedures

This section describes each *action* as a number of *procedures* to be executed by the RO and the involved PO so as to finalize the action. In the following, only the main steps of the procedures are defined, whilst the details are described in the Wiki's Section “The DRIVER Deployment and Operational Support Procedures”, accessible on-line at [6].

#### 3.1.1 External requests handling

##### 3.1.1.1 Add an Application

Adding an application resource is a consequence of the PO having identified and selected a specific DRIVER application template through the relative DTD. The template, together with the application resource SLA parameters, identifies kind and number of resources involved and the list of actions to be executed to deploy the application.

- The RO generates DRP and SLAs for the DTD: assessment of the dimensioning, extension and configuration of the DRIVER infrastructure in order to host the new application resource.
- The RO supports the PO to complete DRP so as to respect the SLAs, which constitute the basic agreement on top of which all further actions regarding that PO will be built.
- Once certified, the application resource is registered to the DSI.

The typology of the individual actions constituting DRPs are described in the following sections.

##### 3.1.1.2 Add a Site

The creation of a site encompasses the following procedures:

- The RO performs an assessment of the dimensioning and configuration of the different DRIVER resources to be part of the site, in order to return an evaluation of the network, hardware and software pre-requisites of the site.
- The RO generates the DRP and the SLAs relative to this action.
- The RO supports the PO in the site certification and operation start-up against the defined DRP and SLA.

### 3.1.1.3 Add a Node

POs providing site resources to the DIS can add new node resources according to their needs. The addition of a new node encompasses the following procedures:

- The RO perform an assessment of the dimensioning and configuration of the new node to be added to an existing site.
- The RO generates the DRP and the SLAs relative to this action.
- The RO supports the PO in the node certification and operation start-up against the defined DRP and SLA.

### 3.1.1.4 Add a new service resource

POs providing node resources can add service resources according to their application needs. The procedures include:

- The RO performs an assessment of the dimensioning and configuration of the node to host the new service instance.
- The RO generates a new DRP and a SLA covering the requirements related to the new service, or updates the existing ones, if a service of that type had been already installed.
- The RO supports the PO in the service certification and operation start-up against the defined DRP and SLA.

### 3.1.1.5 Add a new Software release of a service resource

Since services can evolve during their life time, an update procedure is provided based on the following:

- The RO perform an assessment of the dimensioning and configuration of the node to host the new release of the service instance.
- The RO updates the DRP and a SLA related to the service, to cope with the new service's characteristics and requirements.
- The RO supports the PO in the service update and operation start-up against the updated DRP and SLA.

### 3.1.1.6 Add a new resource typology

In order to extend the functionalities available to the DSI application resources, PO can add new resource typologies, typically together with their first service resource implementation. In the case of new content resource typologies, i.e. new data structure resource typology, new service resource typologies will have to be introduced as well, so as to provide the services for managing resource instances of the former. The procedures required for the addition of a new resource typology are the following:

- The RO evaluates the effectiveness of the proposed resource within the DRIVER infrastructure.
- The RO and the PO collaborate to define the new DUP and SLA covering the characteristics and requirements of the new typology, to be adopted as usage, implementation and validation patterns for the new users to come.
- The RO supports the provider PO during the design, implementation and test of the service software.

### 3.1.1.7 Add a Data Source

Data sources are Data Structure Resources and are added to the DSI through the service resources entitled of handling their integration. The procedures required for their addition are:

- Assessment of data source software, metadata, supported standards for interoperability, etc..
- If the Data Source is “DRIVER compliant”, i.e. there exist services capable of managing their content: definition of the DRP and the SLA relative to this action.
- If the Data Source is not “DRIVER compliant”, i.e. no service resource currently available is capable of handling its content: consider the design, implementation and test of such services (action: “addition of a new service resource typology”).
- PO entitle of data source registration.
- RO entitled of data source certification and validation.

The DSI can handle a number of different data sources. In the current set of service resources, DRIVER compliant data source are limited to OAI-PMH repositories.

### 3.1.1.8 Access an Application, a Service or a Data Source

Existing applications can be accessed by consumer POs whose resources need to reuse and access one or more of the application resources, be it a service, a data source or another form of data structure. The DRP of the consuming PO will include the resource, together with the related SLA.

- The consumer PO submits a DTD with the structured description of its requirements for the application.
- The RO perform an assessments of access rights for approval or denial.
- If approved, a DRP and a SLA are generated, and planning or training for access is provided by the RO.
- The RO supports the PO in the application access start-up against the defined DRP and SLA.

### 3.1.2 Infrastructure maintenance

For the structure and execution of these *actions*, please refer to the DRIVER Control Centre documentation [1].

### 3.2 Service Level Agreement and Quality of Service in DSI

This section introduces the concepts of Service Level Agreement (SLA) and Quality-of-Service (QoS). For each POs DTD, specific DRP and SLAs will be defined, describing duties of the former based on the requests they enquiry for. SLAs constitute “service contracts” between RO and POs and represent the term against which the QoS, i.e. the level of service supplied by each POs, must be monitored and guaranteed. The RO Management Centre has the duty to verify that DSI resources are respecting their SLAs and alert and support PO Operation Centres to adjust the situation when this is not the case.

Given the complexity of the DSI environment, QoS has to be defined taking into account a number of different parameters, ranging from the quality of the software deployed on the nodes to the ability of the Support Centre to guarantee an adequate response time to users’ support requests, as well to ensure “good quality” content, etc..

The following Tables define the “quality parameters” that have been taken into account to define the DSI QoS along with the related monitor and control activities, the metrics used to measure such parameters, and the thresholds adopted to assure a defined level of QoS. Such parameters will be used to define specific SLA for different resources with different PO users.

Most of the following parameters (points 1 through 6) are tailored on the ISO/IEC 9126 standard [ISO9126] and apply to any resource. Parameters at points 7 and 8 are specifically provided for the monitoring of the support activities, whilst parameters at point 9 are devoted to the monitoring of the quality of data sources’ content.

When a parameter is associated to more than one indicator and related metric, the table will also specify to what extent each metric contributes to the parameter evaluation. In the following “weighted trees” for QoS evaluation are given for each of the defined parameters. Then corresponding thresholds are suggested to establish a quantitative acceptable limit for each metric. Values for “Weight” and “Threshold” are defined as examples for some parameters only. The actual values have to be changed/confirmed or specified when instantiating these on real cases.

Parameter	Weight	Indicator	Weight	Metric	Threshold
Functional coverage	0.4	Requirements coverage	1	% (Satisfied vs. required)	0.90
Security	0.2	Resource access security	1	% per M (# of intrusions vs. # of accesses)	0.99
Interoperability	0.1	Interoperability	1	% (Satisfied vs. required)	0.80
Accuracy of results	0.2	Accuracy	1	% (# of right vs. # of expected)	0.90
Compliance to standards	0.1	Standards coverage	1	% (Complied vs. required)	0.95

**Table 3 – Resource Functionality**

Parameter	Weight	Indicator	Weight	Metric	Threshold
Availability	0.33	Uptime	0.33	% per M	0.90
		Scheduled	0.33	% per M	0.06

		downtime			
		Failure downtime	0.33	% per M	0.04
Fault Tolerance	0.33	Minimum Guaranteed QoS	0.33	% of standard QoS (Mix(s) of the provided thresholds)	
		Robustness level	0.33	% per M (# of stop vs. # of failures)	0.60
		Time to restore	0.33	Avg time (Time of failures vs. # of failures)	
Recoverability	0.33	Time to restart	1	Avg time (Tailored on different stop severity levels)	

**Table 4 – Resource Reliability, Availability and Serviceability (RAS)**

Parameter	Weight	Indicator	Weight	Metric	Threshold
Ease of training	0.3	Time to learn	1	Avg time	
Ease of use	0.3	Time to use	1	Avg time	
Operability	0.4	Time to install	0.4	Avg time	
		Time to manage	0.6	Avg time	

**Table 5 – Resource Usability**

Parameter	Weight	Indicator	Weight	Metric	Threshold
Response time	0.4		1		
Resources usage	0.3		1		
System load	0.3		1		

**Table 6 – Resource Efficiency**

Parameter	Weight	Indicator	Weight	Metric	Threshold
Analysability		Time to analyse a problem	1	Avg time	
Changeability		Time to production	1	Avg time	
Stability		Stability wrt changes in the infrastructure	1	% per Y (# of resource changes vs. # of infrastructure changes)	0.9
Testability		Time to test a solution	1	Avg time	

**Table 7 – Resource Maintainability**

Parameter	Weight	Indicator	Weight	Metric	Threshold
Adaptability		Code invariance wrt different targets			
Installability		Easy of installation on different targets			
Conformance		Conformance to technical standards			
Replaceability		Replaceability of resources with new versions			

**Table 8 – Resource Portability**

Parameter	Weight	Indicator	Weight	Metric	Threshold
Response time	0.6	Tickets duration	1	Avg time	
Solution cost	0.4	Tickets opened	0.33	Avg # per M	
		Tickets closed	0.33	Avg # per M	
		Tickets interactions	0.33	Avg #	

**Table 9 – User Support**

Parameter	Weight	Indicator	Weight	Metric	Threshold
Response time	0.4	Tickets duration	1	Avg time	
Solution cost	0.6	Tickets opened	0.33	Avg # per M	
		Tickets closed	0.33	Avg # per M	
		Tickets interactions	0.33	Avg #	

**Table 10 – Operational Support**

Parameter	Weight	Indicator	Weight	Metric	Threshold
Reputation					
Sustainability					
Security Enforcement					
Interoperability Support					
Scalability					
Authenticity					
Integrity					

Provenance					
Freshness					
Preservation Performance					
Size					
Scope					
Trustworthiness					
Fidelity					
Viability					
Metadata Evaluation					

**Table 11 – Data Source QoS**

## 3.3 Instruments

### DSI organisation

The DSIWiki gathers all relevant information related to the management of the service infrastructure:

- DSI policies, e.g. PO supervision and support, resource management actions and procedures, users involved per actions;
- DSI status , e.g. available resources per PO, resource current status, PO users involved, PO participation agreement;
- DSI administration history, e.g. log of management operations.

The website is expected to evolve during the project lifetime so as to reflect the progresses of DSI maintenance. For a complete description of the Wiki refer to [6].

### Issue Tracker

The issue tracker is an indispensable tool especially in a geographically distributed and highly heterogeneous development team. Its uses can be extended to front-office mode interaction, in order to distribute the enquiries and requests from external PO users among the DRIVER Support Centre administrators.

### Statistics Monitoring

Special Monitoring tools have been developed to support DRIVER Management Centre administrator activities. Full description of such tools is given in Deliverable 7.3 [9].

### Data Sources Validation

The DRIVER Consortium has developed a set of Guidelines for Content Providers, which, when implemented, improve available metadata and assist DRIVER in simplifying the use of this metadata by service providers. Furthermore, the use of the DRIVER Guidelines provide orientation for managers of new data sources to define their local data-management policies, for managers of existing data sources to take steps towards improved services and for developers of data source platforms to add supportive functionalities in future versions. The following guidelines are available:

- DRIVER Guidelines Overview [DGO]
- DRIVER Guidelines FAQ [DGFAQ]
- DRIVER Guidelines Implementation Specifics [DGIS]

The development and application of the DRIVER guidelines is an on going effort, and thus they are frequently updated. The most current versions can be found at:

- <http://validator.driver.research-infrastructures.eu/>

Moreover, the DRIVER team has developed a web based tool, whose task is to check the degree of conformance with the guidelines. The Validation Tool is accessible at:

- <http://validator.driver.research-infrastructures.eu/>

Implementing the Guidelines is straightforward and assistance is available from the DRIVER Guidelines helpdesk at:

- [helpdesk@driver-support.eu/](mailto:helpdesk@driver-support.eu/)

### DSI management

The DRIVER Control Centre (DCC) [1] delivers functionalities for centralized management of the whole DRIVER infrastructure, i.e. for the interaction between RO and POs and the establishment

and sustainment of the relative SLAs. In particular, the DCC aims at supporting the interaction between POs and RO from the initial contact, to the integration of their data sources to the deployment and/or operation of the relative PO applications. DCC offers a broad set of administration, configuration management, provisioning, monitoring, and security capabilities functionalities in a structured way, in order to reduce the cost and complexity of managing DRIVER environments. Resources management functionality within DCC improves the DSI service levels through performance monitoring and diagnostics for managed resources.

DCC can be used to:

- By POs, to submit to the DRIVER Management Centre a request to the RO with a DTD form (structured and grounded description of the PO needs). The DCC should return the relative DUP profile, which identifies the proper type and mix of DRIVER resources to be deployed by the PO Operation Centre, hence the actions to be accomplished by the PO and the RO to achieve this result. Such operations by turn out to be:
  - Install the DRIVER resources as an autonomous node within the DRIVER infrastructure (closed-loop).
  - Install the DRIVER resources as a node within the DRIVER infrastructure (open-loop).
  - Develop a new Application on the DRIVER infrastructure.
  - Develop a new Service for the DRIVER infrastructure.
  - Plug a Data Source into the DRIVER infrastructure.
- Define the proper dimensioning and configuration of the target HW&SW environment with respect to the requested DRIVER resources.
- Control of compliance of new resources with respect to the DRIVER infrastructure policies.
- Simplify deployment of DRIVER infrastructure by implementing automated standard procedures or even by cloning fully (patched and) tested DRIVER nodes to multiple hosts.
- Quickly identify available DRIVER SW for a node and bring this SW up to the latest patch level.
- Apply consistent, proactive management practices across multiple nodes using policies and management templates.

## 4 References

- [1] *D5.Y The DRIVER Control Centre*  
<http://reflink/>
- [2] *The DELOS Digital Library Reference Model*  
[http://www.delos.info/files/pdf/ReferenceModel/DELOS\\_DLReferenceModel\\_0.98.pdf](http://www.delos.info/files/pdf/ReferenceModel/DELOS_DLReferenceModel_0.98.pdf)
- [3] *DRIVER Guidelines FAQ,*  
<http://www.driver-support.eu/faq/guidelinesfaq.html>
- [4] *DRIVER Guidelines Implementation Specifics,*  
[http://www.driver-support.eu/documents/DRIVER\\_guidelines\\_annexes\\_1\\_1\\_1.pdf](http://www.driver-support.eu/documents/DRIVER_guidelines_annexes_1_1_1.pdf)
- [5] *DRIVER Guidelines Overview,*  
[http://www.driver-support.eu/documents/DRIVER\\_guidelines\\_1\\_0.pdf](http://www.driver-support.eu/documents/DRIVER_guidelines_1_0.pdf)
- [6] *The DRIVER Service Infrastructure User Guide Wiki,*  
<http://driver.research-infrastructures.eu/ServiceActivities/ServiceInfrastructure>
- [7] Ali Arsanjani, *Service-oriented modeling and architecture*, IBM developerWorks,  
<http://www-128.ibm.com/developerworks/webservices/library/ws-soa-design1>
- [8] *W3C Web Services Activity web site,*  
<http://www.w3.org/2002/ws>
- [9] *D7.3 DRIVER Monitoring Tools Specification*, DRIVER-II Project Deliverable