ECP-2007-DILI-517006

EFG – The European Film Gateway

Functional Analysis of DRIVER customisation

Deliverable number	*D 4.1*
Dissemination level	*Confidential*
Delivery date	*30 January 2009*
Status	***Draft (22-january-2008)***
Author(s)	*Georg Eckes (DIF), Paolo Manghi (CNR-ISTI), Pasquale Savino (CNR-ISTI)*

e**Content*****plus***

This project is funded under the *e*Content*plus* programme,
a multiannual Community programme to make digital content in Europe more accessible, usable and exploitable.

TABLE OF CONTENT

1 Introduction

The aim of this deliverable is explaining how the main applicative functionality requirements identified in the Description of Work (DoW) [2] can be implemented as a service-oriented application within the DRIVER infrastructure application framework **Error! Reference source not found.**. The resulting system will consist of an EFG infrastructure hosting one EFG application, which will be capable of maintaining the European Film Information Space and making it accessible through a Web Portal.

Section 2 introduces the functionalities to be implemented in EFG, while Section 3 provides a description of the DRIVER application framework and illustrates how the technology available to EFG partners can be integrated into the EFG infrastructure and used to compose the EFG application.

2 EFG functional specification

From the DoW, the main EFG technical objectives are the construction and maintenance of a European Film Information Space, the provision of a Web Portal to search over its content and the injection of such content into the Europeana Project database [3], i.e. the "cauldron" of European Cultural Heritage digital information. In this section we analyse these objectives in order to "formalize" their implementation issues in terms of functional requirements.

2.1 European Film Information Space

EFG's main purpose is that of constructing and populating a uniform Information Space of film-related metadata information by gathering and "harmonizing" content available from reliable European film information data sources. The resulting *European Film Information Space* (*EFIS*) will serve world-wide user communities interested in the movie industry with an unique official source of catalogued information.

The current scenario features a number of autonomous European Film data sources gathering and organizing film metadata information according to local data models and schema, i.e. metadata format. The integration of such data sources is strongly hindered by both their schema heterogeneity and their possible redundancy at the content level. Indeed, although the different data models typically share a common core of information, e.g. film, artists, movie companies, these tend to include peculiar pieces of information and, more generally, not to conform to a common agreed structure; besides, being independently administrated, different data sources may replicate film information, which could be identical or clash or be instead complementary in some aspects; e.g. two data sources may contain information about the same film, but one with the original title and the other one with the title translated in another language.

From this analysis, the project envisages the definition of a common data model describing the features considered *relevant* to the film domain and of the relative schema. Starting from that, the plan is that of ***transforming*** content ***published*** by local data sources into the corresponding common schema representation. Such import action should consider ***cataloguing*** operations, for example to blend or eliminate information originating from different sources but relative to the same entities. The following functional requirements can therefore be identified:

- ***Content Publishing***: not all data sources offer web/internet accessible interfaces and protocols to read, i.e. harvest, their content. Where missing, this functionality should be provided.
- ***Content Transformation***: data conforming to one format, possibly harvested from a data source, need to be transformed through a well defined format-to-format mapping into another format. Mapping definition and mapping application functionalities should be provided.
- ***Content Cataloguing***: some entities of the data model, e.g. film, artists and movie companies, need functionalities for duplication control, information blending and post-transformation cleaning.

2.2 EFIS Web-portal application and integration to Europeana database

The remaining two technical objectives of the project, namely providing an EFG Web Portal over the EFIS and providing the EFIS to Europeana database, fall under the same general issue of ***accessing content in the EFIS from applications***. Applications can be portals, such as the EFG Web Portal, or third-party consuming services, such as Europeana, willing to interact with EFIS to serve their user communities. The following functional requirements can be thus identified:

- ***OAI-PMH interfaces***: OAI-PMH protocol should be supported to access the whole EFIS or a define the most appropriate OAI-PMH Sets.
- ***OAI-ORE interfaces***: OAI-ORE protocol to export compound film objects should be supported.
- ***Information Retrieval Search***: EFIS should enable Information Retrieval searches over its content and offer access to such functionalities through open protocols, such as SRU/SRW over HTTP.

Implementing such functionalities would enable demanding organizations to develop applications on top of the EFIS according to their needs. An example of this is the EFG Web Portal itself, which will exploit the open search functionalities to offer end-users the functionalities needed to discover the information they seek for. In particular, the EFG Web Portal should consider the following functionalities:

- ***Collection management***: end-users should be able to query portions of the EFIS, called collections, whose content is characterized by common constraints; e.g. “films of American production”, “films before 2000”. The portal should enable administrator users to define and organize collections into a hierarchy so as to facilitate end-users.
- ***Multilingual searches***: film information may be provided in different languages and end-users queries should be executed taking this factor account.

3 EFG architectural specification

In order to achieve its functional objectives, the project will adopt the DRIVER infrastructure technology and shape its system over the application framework it offers. The technological outcome of the DRIVER-II project is an Open Service Infrastructure capable of (i) supporting the construction of Information Spaces of digital objects collected from a number of heterogeneous Repositories and (ii) the construction of Service-oriented applications, whose services can be shared and reused in the context of others, on top of such Information Spaces.

The DRIVER software is conceived to be open and flexible and tailored to "glue" and integrate different technologies as parts of the same application. Accordingly, EFG plans to rely on the DRIVER infrastructure so as to exploit the functionality services it currently offers, i.e. harvesting, transformation, storage, indexing and collection services, and extend it with the services it currently misses to the purpose of the project, i.e. the multilingual-oriented MILOS IR system (ISTI-CNR) will be integrated as a service and new web portal services will be built so as to interact with the resulting Information Space.

In the next sections, we shall describe the principles lying behind the DRIVER application framework then explain how the functional requirements introduced in the previous section can be fulfilled by using existing technology or by developing/integrating new technology. To conclude, we draw a model of the resulting EFG infrastructure and application.

3.1 The DRIVER application framework

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/or 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) **Error! Reference source not found.**. 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. set of information about their location (access point), 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 (*i*) data source resources, which are integrated through special services into the infrastructure so as to exploit their content; (*ii*) functional resources, which are services running at some DRIVER node in turn operative at some DRIVER site; (*iii*) and applications, which are dynamic combinations of such services. Application resources are provided by POs and can be consumed by PO end-users or by PO external applications, which can access the DRIVER services through standard access APIs.

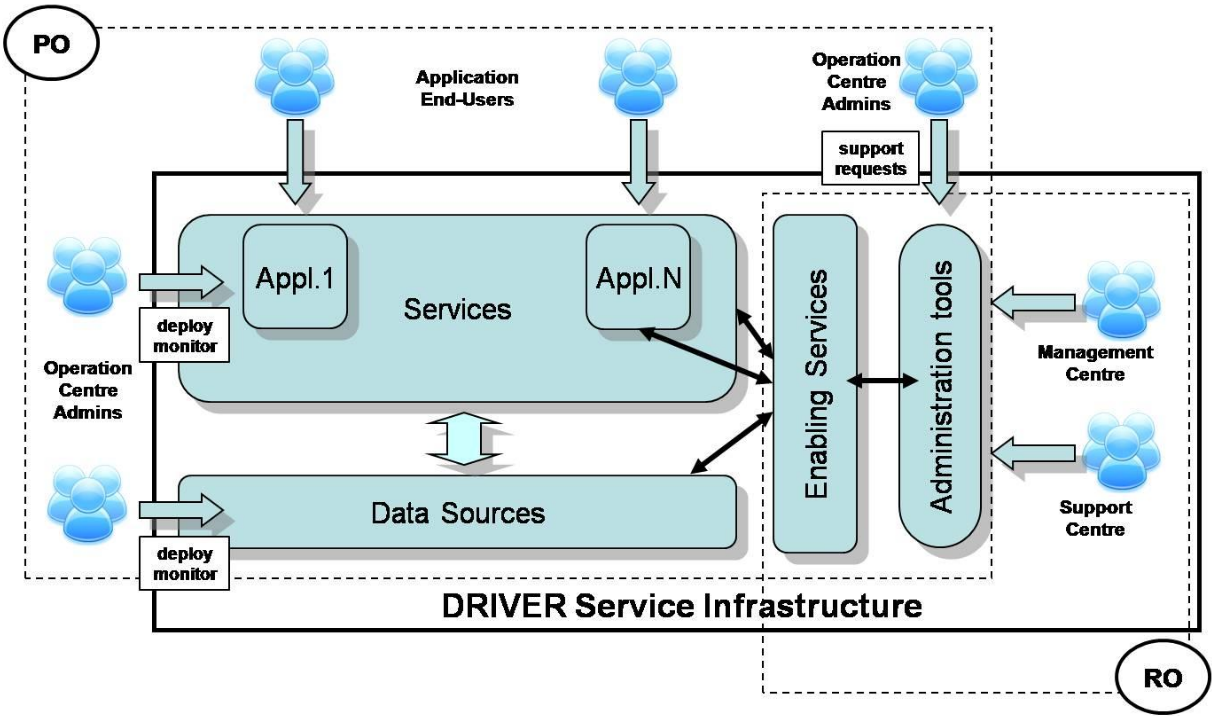

Figure 1 – The DRIVER Service Infrastructure architecture and organization

The RO operates the Management Centre and the Support Centre, respectively dedicated to the administration of the infrastructure (orchestration and monitoring of the resources running in the infrastructure) and to offer technical support to POs Operation Centres willing to partake the infrastructure environment.

3.2 Existing Technologies

In this section we introduce the technologies that will be used to support the EFG application, together with the customization steps required for their integration in the EFG infrastructure.

3.2.1 Infrastructure Service Kits and middleware

The DRIVER Enabling Services will be used to give life to the EFG production infrastructure, on top of which the EFG Portal application will be constructed. The infrastructure service-oriented application framework will allow for the exploitation of the functionality services delivered within the DRIVER project and their integration with new services, to be designed and developed in the EFG project. In particular, the DRIVER Service Kit offers tools for the definition of data aggregation workflows, for Information Space organization/access and for User Management:

- Harvesting Services: accessing external OAI-PMH conformant data sources to retrieve the metadata records they expose;
- Transformation Services: transforming metadata records of arbitrary complexity from one format into another;
- Store Services: storing metadata records of arbitrary complexity;
- Indexing Services: indexing content, i.e. metadata records of arbitrary complexity, extracted from store services; informally, a set of indices fed with metadata of the same format forms a DRIVER Information Space;
- Collection Services: managing a hierarchy of named collections partitioning the EFIS; collections are implemented as CQL queries with a unique name,;
- Search Services: reply to CQL queries [7] over an Information Space by selecting the Index Service of that Space that can best perform the query;
- OAI-PMH Publishing Services: implementing the OAI-PMH protocol to publish the content of an Information Space;
- OAI-ORE Publishing Services: implementing the OAI-ORE protocol to publish the content of an Information Space;
- User Management Services: managing user profiles;
- Authentication and Authorization Services: managing user authentication and authorization on behalf of services.

The DRIVER infrastructure and the data aggregation workflow service kit in operation are illustrated in Figure 2. Figure 3 depicts user functionality services (user, collection, recommendation, community services) and their possible interaction to support features of the Web Portal.

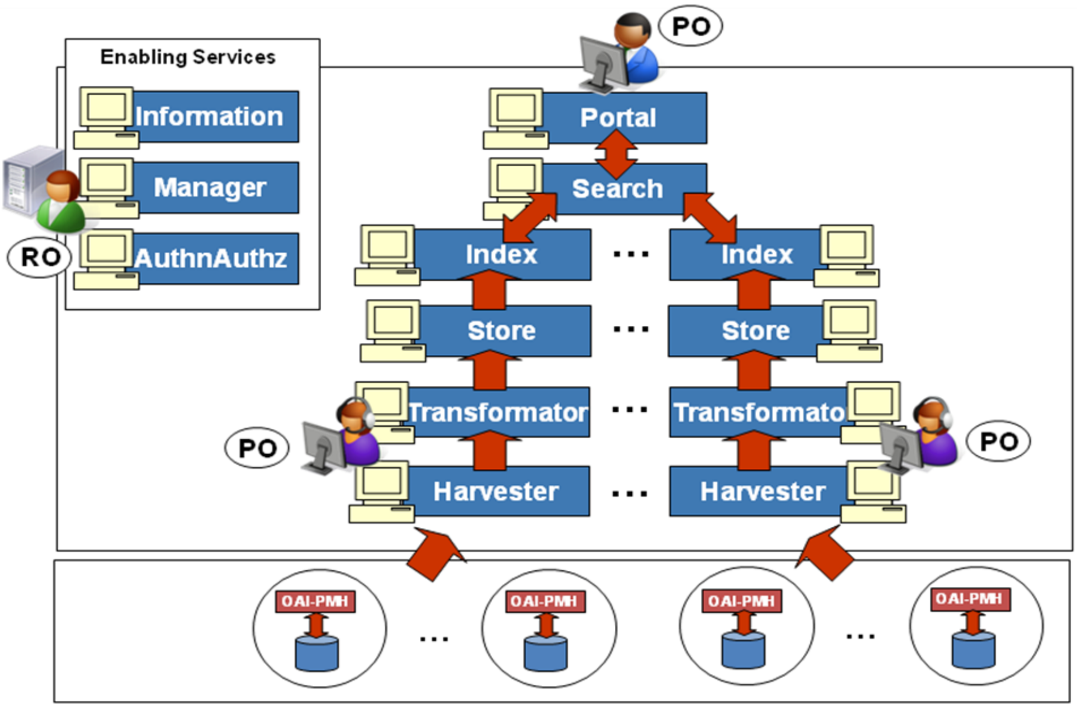

Figure 2 – DRIVER Architecture: data aggregation workflow application

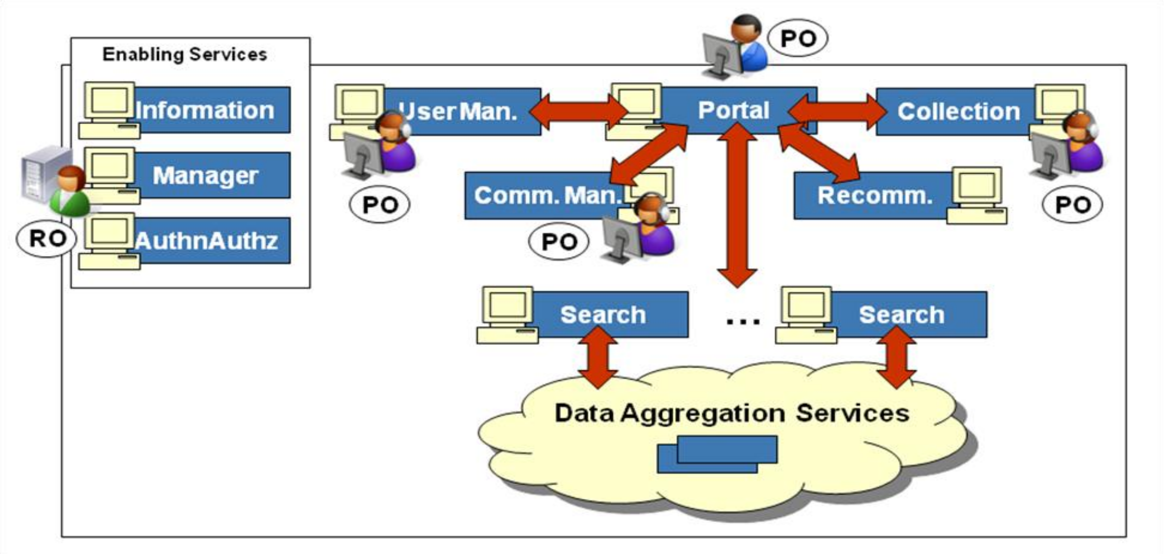

Figure 3 – DRIVER Architecture: user portal application

DRIVER customization

Services in the DRIVER Service Kit not only provide support for the aside but required functional aspects of dataflow management, such as persistent storage and authorization/authentication mechanisms, but also match the following EFG functional requirements:

- ***Content Transformation***;
- ***OAI-PMH interfaces***;
- ***OAI-ORE interfaces***;
- ***Collection management***.

Content Cataloguing functional requirements are instead not comprised in the DRIVER Service Kits. However, transformation services, which are capable of operating changes and modifications to metadata records, could be naturally extended with functionalities for authority file management. Metadata records harvested from data sources and mapped into the common EFG schema should be organized and maintained by EFG administrator users to form authoritative directories for films, artists and movie companies. Due the peculiarity of this functionality, a new Authority File Management Service might be considered as an alternative to Transformation Service's extension.

3.2.2 MILOS (YAPI and XML SEngine)

- Functional requirement to be fulfilled: ***Information Retrieval Search*** and ***Multilingual searches*.**

- DRIVER indices do not provide support for multilingual issues.

To be completed...
DRIVER customization

Wrap service in to Web Service Suite and satisfy the minimal requirement for infrastructure integration; e.g. registration of the service profile, update of the service profile status.

3.2.3 OpenDLib Repository

The OpenDlib Repository (ISTI-CNR) is a distributed, component oriented system capable of supporting the construction of customized digital library applications. The system provides a number of functionalities that can be properly configured and combined so as to match the different expectations of the DL user communities. Among other features, OpenDLib includes a framework for metadata transformation that can be easily adapted to import metadata from files or through a variety of standard access protocols. Not only, metadata imported/stored within the system can be accessed through OAI-PMH interfaces by any consuming service. As such, OpenDLib can also play the role of OAI-PMH gateway for external data sources not equipped with such functionality.

DRIVER customization

As mentioned in the description of the EFG application scenario, some of the data sources may not provide an OAI-PMH publisher service. In these cases, the EFG project aims at supplying the underlying organizations with tools, i.e. services, that can serve their data source technologies with OAI-PMH publishing mechanisms.

Due to its stand-alone installation and its aggregative and publishing tools, OpenDLib could be adopted to satisfy ***Content Publishing*** functional requirements. The system can be used to import metadata records according to the format and protocol available at the partner site and then to convert the records into the common EFG schema. Architecturally, two solutions are possible for OAI-PMH publishing: different instances of the system could be installed at the data source sites to provide independent OAI-PMH interfaces for the EFG infrastructure; or instead one single instance could be installed to centrally import content from all data sources and thus provide EFG with a single point OAI-PMH access to all of them.

3.3 EFG Infrastructure and application

The EFG infrastructure will therefore consist of an instance of the DRIVER infrastructure, appropriately integrated and customized with the services required by the EFG application and still missing in the DRIVER Service Kit. Figure 4 illustrates the architecture of the EFG application, where the red colour highlights the required customizations.
The MILOS System technology will be integrated to offer Information Retrieval and Multilingual search functionalities, while Transformator Services will be endowed with authority file management features. Finally, in order to enable OAI-PMH harvesting of all data sources available to EFG, OpenDLib technology can be supplied to those sites that do not implement such access protocol. Finally, the Web Portal can be built on top of a search service supporting SRU/SRW APIs.

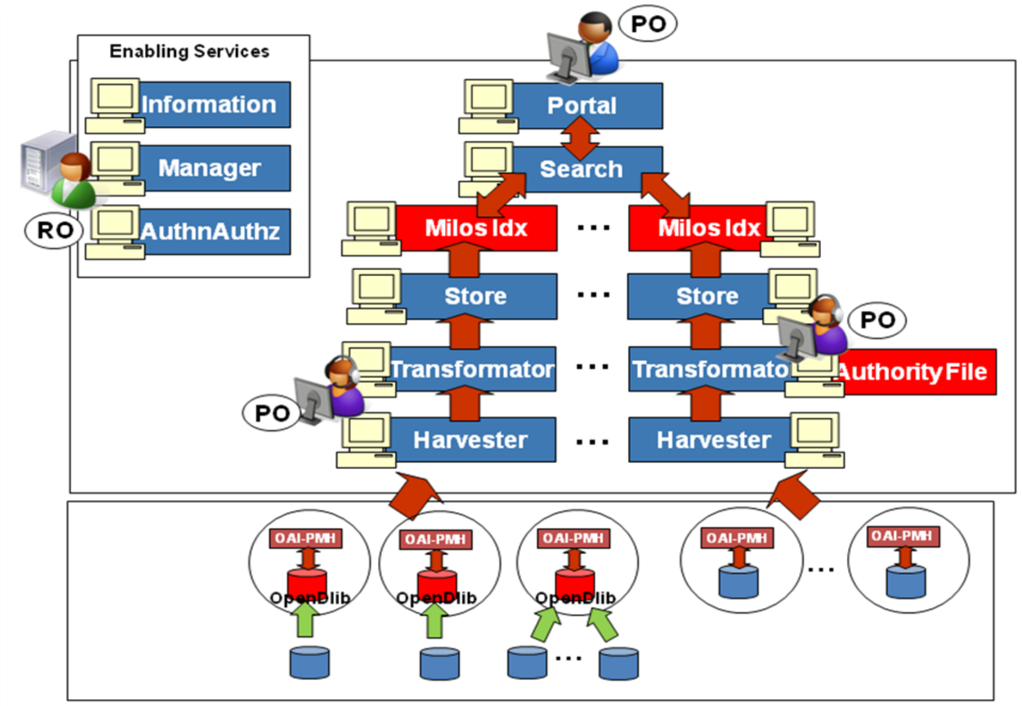

Figure 4 – EFG Infrastructure and application

4 Bibliography

[1] The DRIVER Infrastructure. http://www.driver-community.eu

[2] EFG Description of Work.

[3] Europeana Project. http://www.europeana.eu

[4] OAI-PMH. http://www.openarchives.org/pmh

[5] OAI-ORE. http://www.openarchives.org/ore

[6] SRU/SRW. http://www.loc.gov/standards/sru

[7] CQL. http://www.loc.gov/standards/sru/specs/cql.html

[8] OpenDLib. http://www.opendlib.com