A Knowledge Management Toolkit based on Open Source

Roberta Mugellesi Dow[1], Hugo Marée[2], Raúl Cano Argamasilla[3], Jose A. Martínez Ontiveros[4],
Juan F. Prieto[4] and Diogo Bernardino[5]

[1]European Space Agency, Darmstadt, Germany
[2]European Space Agency, Noordwijk, Netherlands
[3]Terma GmbH, Darmstadt, Germany
[4]Immedia IT, Madrid, Spain
[5]Serco, Darmstadt, Germany

Abstract: The positive experience with the knowledge capture project at the European Space Agency (ESA) Automated Transfer Vehicle (ATV) mission has opened a path worth following by the whole Agency. The ATV team has a privileged insight of what to preserve and how to present it so, to enable the user to get the best out of the mission's knowledge, a portal was created and tailored to the specific needs of this mission (ATVCAP project). In order to capitalize from this experience, a spin-off has been developed in the form of a generic system, the KM Toolkit, that can be instantiated in as many business units as necessary. The idea is to offer to other areas a ready-to-go solution that can be tailored to the particularities of the interested party. This toolkit has been designed upon an architecture that integrates its services in several layers (shared services, integration, knowledge, access) and it provides to the end user a comprehensive set of features that use Open Source at their core: a portal as a single entry point (Drupal), wiki functionality (Drupal & MediaWiki), search engine (Apache Solr and ManifoldCF) and Competency Management (an in-house development based on Open Source). In addition to the KM Toolkit, it was developed as well a mechanism that allowed an easy content population, contributing with this to offer a content-rich solution as efficiently as possible. This paper describes how the Knowledge Management team at the European Space Agency has approached this endeavour from the conception and design to its implementation, based on past experiences (e.g. ATVCAP) and describing as well some lessons learned for the future.

1 INTRODUCTION

When designing a Knowledge Management System for a global organization with more than 2200 employees and international presence, it is a big challenge to find a solution that respects both a highly federated structure and that also enables the organization to capitalize on the knowledge and innovation from its different areas.

At the European Space Agency, two approaches have been used so far. On the one hand, local projects for knowledge preservation and sharing, started and managed at a business unit level. These activities, while offering highly customized solutions, lack the global integration required for a corporate KM system (Mugellesi et al., 2011). That is, knowledge is very frequently not made visible to the rest of the organization. On the other hand, the opposite approach has also been followed by some initiatives that offered solutions of the type one-fits-all, such as ESAnet or ESA Connect, both feature-rich social networks. All the efficiency gained by the centralization of this solution is countered by the lack of flexibility when it comes to customizations requested at local level.

A way to complement the previous scenario is the KM Toolkit, a set of tools ready to be installed in whatever business unit requires it, containing a set of features considered of added value by the users – usually extracted from previous pilots, interviews, lessons learned of previous activities, etc. (Guerrucci et al., 2012). This toolkit must be flexible enough to offer quick response times on its installation and as powerful as to disseminate corporate-wide all knowledge it handles.

2 BACKGROUND

For some years already, ESA's KM team vision has been of building one "ESA brain" from individual "neurons". That is, creating a network of interconnected systems that offer access to corporate knowledge regardless of its location. Each "neuron", together with the ability to connect to other ones, should offer a set of features that have been selected as the most useful by the users in the organization.

The project "ESA KM Strategy Testbed Implementation", which took place between 2010 and 2012, recommended a selection of the best tools and methods supporting KM (KM Portal, Competency Management, KM Officer, etc.) after gathering user input and comparing it with the organizational strategy. Therefore, if a KM Toolkit was to be built, it was more than adequate to form it from the mentioned output.

A good example of a project that uses the preliminary version of the KM Toolkit is the ATVCAP project (Guerrucci et al., 2014; Mugellesi et al., 2014). Here, a set of KM tools developed originally as pilots together with other concepts recommended by the KM team (e.g. taxonomy), were put together to fill the needs of the ATV mission, which decided to start an activity for knowledge preservation. The successful result of this project led to the development of a solution similar to the ATVCAP system, ready to be replicated in any other business unit. This is how the idea of the KM Toolkit was born.

3 A REUSABLE KM TOOLKIT

In order to be able to provide a timely response to directorates' requests in terms of KM, a reusable toolkit is a pertinent answer. Having a generic platform that can be swiftly instantiated in the environment of the requester would help the central KM body to build trust and sense of reliability in the organization.

It is, moreover, a solution of compromise. Business units using the KM Toolkit will have administrative rights over it and ownership of the contents, with the power of customization to their own needs. At the same time, these business units are required to maintain certain interface and a minimum set of requirements that would ensure future connectivity and compatibility with other instances of the product.

Furthermore, three out of four of the selected platforms for the KM Toolkit (Drupal, Apache Solr, MediaWiki) are Open Source, supported by a community large enough to provide information and support on the necessary inquiries. In this way, this activity minimizes licensing costs while optimizing openness and flexibility.

4 REFERENCE ARCHITECTURE

The KM Toolkit is an integrated set of tools with a service oriented approach, where each tool provides one or more services or functions supporting activities of the KM processes.

The definition of a reference architecture helps establish the principles and guides to implement the toolkit, deploy it in different scenarios, and evolve in the future adding or replacing tools without strong design modifications.

Figure 1 shows the reference architecture defined for the toolkit, which is based on the experience from previous KM systems and follows common practices in the field. As depicted in the figure, the set of services and knowledge management functions have been organised in layers.

At the bottom, the Shared Services Layer provides basic infrastructure services which are shared by two or more higher level ones. They are general purpose IT services, such as databases, security and search. Some others such as Document and Content Management systems could be added later.

On top of the previous layer, the Integration Layer plays a crucial role for the toolkit, and as an evolution from past initiatives, it is where most of the effort have been spent. Its purpose is twofold:

- Services integration, allowing KM services to interact with infrastructure services, whether they are common ones provided by the toolkit or they are leveraged from the corporate infrastructure

- Content integration, which provides a single view of the contents generated and used by the KM services, and that normalizes the descriptive information of those contents, given by a taxonomy and metadata.

Next, the Knowledge Layer consists of the high level KM tools that interface directly with the users through the portal. These tools can be organised in different domains related to knowledge management processes. Figure 1 shows some domains proposed:

- **Knowledge Capture and Dissemination.** Tools that enable creating, classifying and making contents available, contents that make the knowledge more explicit and reusable for new
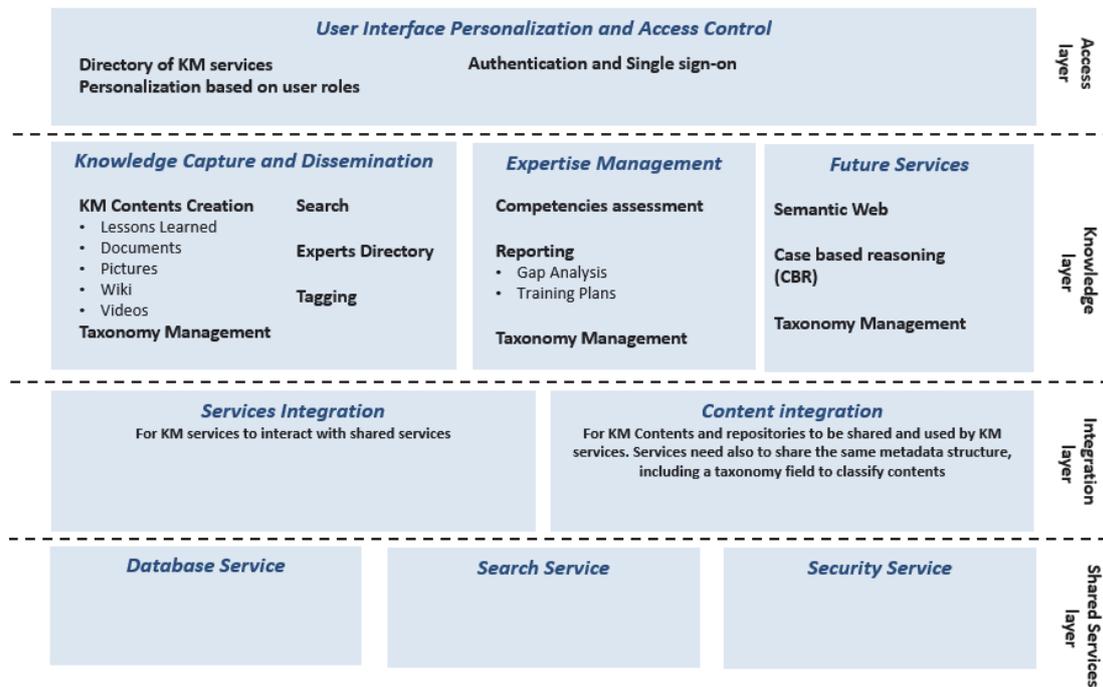
Figure 1: KM Toolkit reference architecture.

purposes: Wikis, Discussion Forums, Articles, Documentation.

- **Expertise Management.** Tools oriented to manage people's expertise, providing ways to detect gaps between actual and required knowledge, and linking this information with proposed training.

Also some future areas could be incorporated in the Knowledge Layer, such as data mining for automatic knowledge discovery, Case Based Reasoning and Semantic Web.

Finally, the Access Layer covers the concepts of authentication, authorization and interface components that would allow the user not only to operate within the KM services, but also to make them agile and usable enough to achieve the aims of each KM initiative.

5 IMPLEMENTATION

Figure 2 depicts how this architecture has been implemented in the first release of the generic KM Toolkit, with the different systems and components, and they are described more deeply in the following subsections.

5.1 Knowledge Management Portal

The KM portal (see Figure 3) is the unifying element of the toolkit. Not only it provides functionality of its own, but also serves as the main entry point to the KM services.

It is based on Drupal, an open source PHP web content management system, it implements the knowledge taxonomy (see section 0) and provides access to all the contents. It also constrains the access to the information by checking users and permissions.

The KM portal uses various Drupal's modules to manage and display the information, so it is possible to interact with other systems like the CMT in order to present the Expert Directory, or to show source code and links to documents in the repository behind the portal. The following are the main functionalities offered by the KM portal:

- Storage and retrieval of knowledge assets: documents, lessons learned, pictures, videos, etc..
- Collaborative content creation.
- Internal search (Drupal's repository).
- External search, within repositories not directly related to the toolkit and stored elsewhere.
- Taxonomy definition, to categorise the information for better searching and retrieval
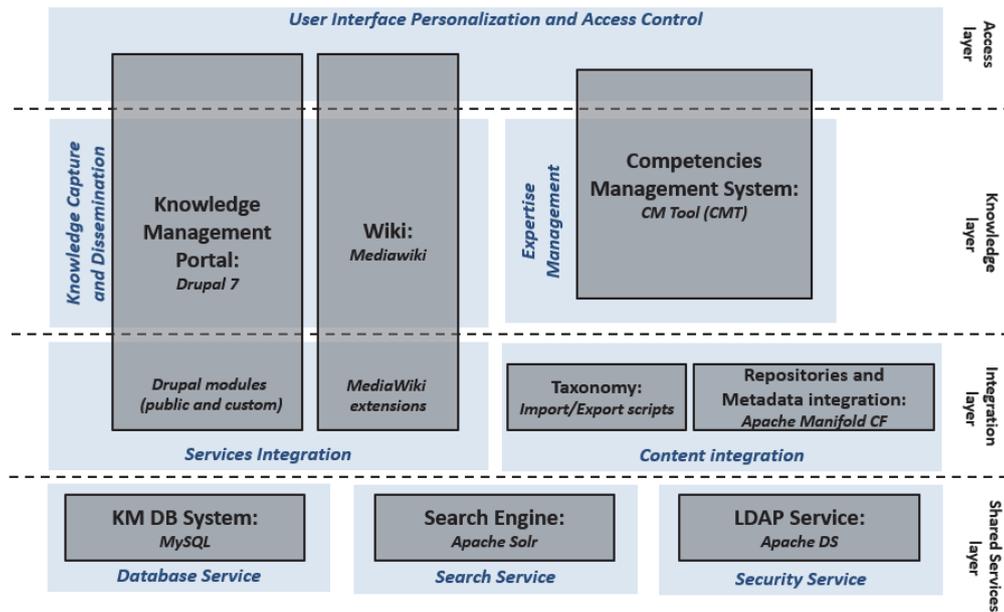
Figure 2: KM Toolkit current implementation.

- Customized presentation of search results, according to content types and their metadata
- Authentication and authorization, based on the ESA corporate LDAP service.

The specific functions that the portal provides through the user interface have been developed leveraging a Drupal flexible mechanism called "features" which allows to encapsulate related functionality and configurations into independent modules that can be easily activated or deactivated depending on the business unit requirements (i.e. video contents and particular functions related to video managed by a single module).

Figure 3: KM Portal homepage.

The design of the Drupal code is as follows:

- A common layer including the fields applicable to different features of the system, and the common visualisation elements.
- A set of independent features for the different content types.

This type of design allows to tailor the KM Portal to the different use cases, by enabling or disabling the content types to be used.

The full portal has been implemented following standard coding conventions, to easily identify and maintain the source code.

5.2 Wiki

The Wiki functionality is provided by MediaWiki, the software upon which Wikipedia is based. Although this is part of the KM Toolkit, the MediaWiki operates in a standalone fashion and is not integrated within the portal, due to the large effort necessary to such an endeavour. Hence, currently it is an independent tool which can be accessed from the KM portal, but running on a different platform. It must be considered that:

- It does not share configuration related to other tools with the KM portal and provides a different user interface and navigation.
- It uses a different user/login and permission schema from the portal (though a single sign-on approach is planned in the tools composing the toolkit).

- It uses a separate database from the rest of tools (this will happen with most tools in the toolkit).

In order to work around the fact that MediaWiki and the KM portal are working separately, two alternatives have been considered and implemented, letting the business unit chose the most fitting one:

1. Use the lightweight wiki functionality from the KM portal to allow users to create Wiki-like content. This is a valid solution for units which do not need complex article creation lifecycle, so everything would be supported by the KM portal and the MediaWiki would not be necessary.

2. If the use of a MediaWiki platform is required, the search engine can be used as the link between it and the KM portal. That is, the search engine would index wiki entries and present them as results from the KM portal providing a mid-way integration between platforms that may satisfy most of its users.

5.3 Competency Management System

This system is implemented by the Competency Management Tool, a tailored platform for a comprehensive and validated process that provides ESA an insight into the current and future required competencies of a particular business unit (Mugellesi et al., 2013). Such a repository can guarantee to an organisation the availability of necessary skills and abilities to carry out the unit's work. This would also provide a management support for identifying competency gaps, training opportunities and development plans on both short and long terms.

The CMT allows the different actors to interact through the tool along the many different steps until the current picture of the organisation has been completed. At the end of the process a summary report will be provided by the tool for an overall analysis (see Figure 4).

The design of the CMT is follows a layered approach:

- Data layer: To access the information in the database.

- Business layer: Java components running on a web application server, in which the actual application code runs, implementing the logic.

- Presentation layer: Web browser related technologies, producing the static and dynamic elements of the web pages visualized in the web browsers.

The web application has been developed making use of Java Servlets and JSP pages, following the MVC design pattern (Model-View-Controller). The model is implemented using simple Java objects (POJOs) linked in JAR libraries. The view is implemented using mainly JSP (Java Server Pages), and the associated HTML and JavaScript. The controller is implemented in servlets, with the responsibility to coordinate the processing and implement necessary checks. Also, the CMT uses Pentaho Business Intelligence libraries, for reporting and MDX queries. These technologies are very suitable to meet the intensive reporting requirements of the competency management process.

From a functional point of view, the CMT can be decomposed in the following main modules:

- Editing functionality, to populate data.

- Data analysis and reporting.

- Process dashboard (see Figure 4).

- General purpose functions, e.g., to implement security, configuration, and others.

5.4 Search Engine

The integrated search engine is based on Apache Solr and it is a key functionality of the toolkit for two reasons. First, it enables the user to find information stored both within and outside the portal. Second, due to its decoupling from the core KM Portal, it could potentially provide with search results to different instances of the KM Toolkit, having a centralized index feeding a decentralized network.

One of the critical elements of a search engine is its schema definition, that is, the set of metadata fields that the engine will index and store from the incoming contents. A custom schema has been created in order to meet the different needs of the components in the Integration Layer (see section 4).

The search engine offers the following features:

- Indexing of portal contents: that is, a direct integration with the search engine, and to ensure synchronization between portal contents and search indexes.

- Indexing of non-portal contents: it is important to be able to index information from other sources; contents from other toolkit components (Wiki, Experts, etc.) and contents outside the toolkit (documents, Web, RSS, etc.). Most of these contents carry a metadata set whose field names do not match the ones in the basic schema, and therefore some metadata manipulation is required (through the metadata integration system that is described later).
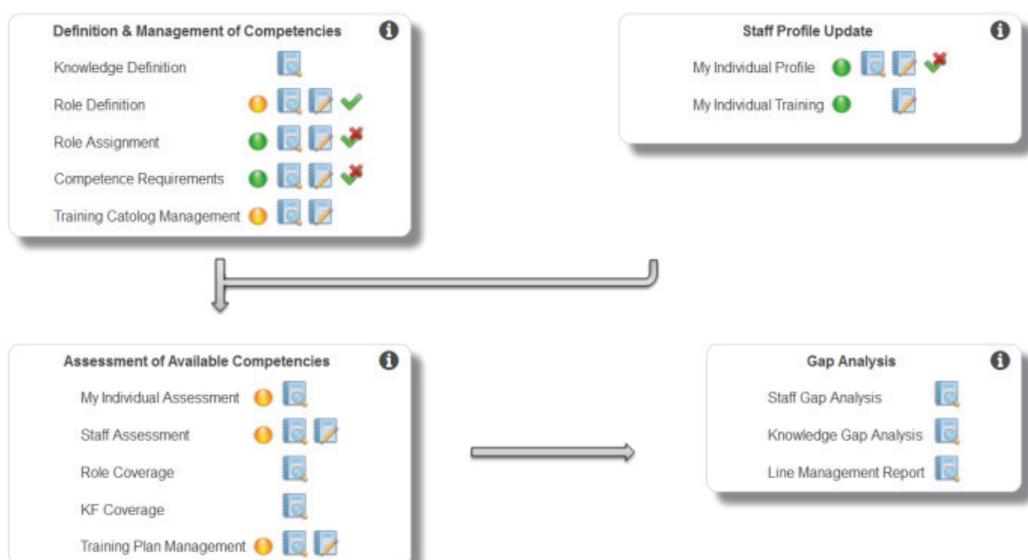
Figure 4: CM Tool homepage.

- Searching capabilities from the portal: one single query to get results from portal contents and external contents.
- Security on results: the system shows results according to the authorisation level that the current user has. Because of results can belong external contents, the initial schema has been extended to include specific security fields for them, with the consequence of having two different field sets containing security information. The Drupal-Solr search module has also been customized, in order to add some additional security filters whenever a search is triggered.
- Advanced search: to search for specific values in metadata.
- Faceted search: allows a user to explore a collection of results by successively applying predefined filters.

5.5 Repositories and Metadata Integration

The capabilities of the toolkit to inject contents from internal or external repositories into the search engine is achieved by Apache ManifoldCF, an integration and synchronization framework whose core ability is to move contents between different systems and to manipulate related metadata.

As depicted in Figure 5, ManifoldCF basically manage jobs, each one consisting of an input repository, an output target (usually Solr), and an authority connector that deals with access privileges that contents have in their origin. This information is important to ensure that search results are properly filtered according to the original privileges.

ManifoldCF also provides features for metadata transformations, allowing to adapt original metadata to the normalised schema in the search engine, by providing operations for changing field names and adding new fields. The capability to work with regular expressions is very useful for this purposes.

Jobs are managed through a web application, and a basic setup has been needed to work with the toolkit database server. As configuring jobs from scratch may be time consuming the toolkit provides some basic pre-set jobs, for web, CMT and shared folders contents integration.
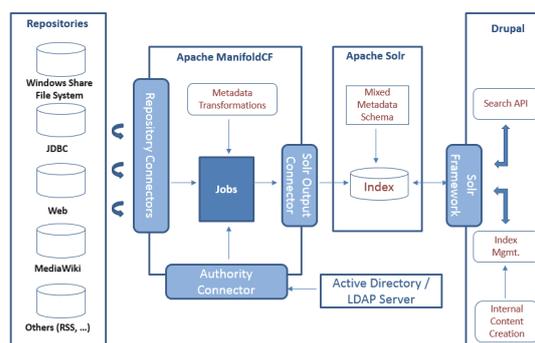

Figure 5: Contents integration with ManifoldCF.

5.6 Taxonomy

The taxonomy is a hierarchical structure in which

the knowledge assets are classified and can be considered as the backbone of content categorization within the platform.

Each of the KM Toolkit's components (KM Portal, MediaWiki, Competency Management Tool) is expected to house a copy of the taxonomy to organize the knowledge they handle. Since they are not automatically synchronized, in the initial versions of the system the administrator had to take care to keep the right version in all platforms. One of the improvements provided within the version we describe in this paper is that the KM Toolkit provides a set of scripts that setup the same taxonomy in all of the components, keeping consistency of the knowledge structure and hence diminishing the risk of content misclassification.

5.7 LDAP Service

LDAP services are normally used to support authentication mechanisms in the access layer, such as the sign-in process within the toolkit, and single sign-on, to avoid having different credentials to access different KM services. This service is often implemented by a corporate service (such as Microsoft Active Directory), but as the toolkit must be flexible enough to adapt to different environments, where the access to that corporate service could not be ensured or there could be some management issues, it has also implemented a light weight LDAP based on Apache Directory Server (DS).

The LDAP service is particularly relevant in the KM Toolkit because it plays the role of integrating security for the different tools. It is therefore a way to ensure consistency across the tool, from the data integration components up to the applications.

ManifoldCF uses the LDAP groups as well, to inject authorization tokens associated with contents when they are indexed in Solr.

The CMT and KM Portal use LDAP to identify users, authenticate them, and apply security restrictions based on their role. Groups are synchronized with the role-based authorization system of the KM portal, through an integration module, and users validating against LDAP are automatically restricted to see the internal contents which are relevant to their group membership.

5.8 KM DB System

Finally, it is worth mentioning that for the sake of simplicity and in order to reduce maintenance costs, all tools have been configured to use the same database service as the support system to store their contents and setup information.

MySQL and MariaDB relational database servers have been tested and can be used with the toolkit.

6 CONTENT POPULATION

The KM Toolkit also provides a way to facilitate the content population in mostly an automated way, both for the KM Portal and the Wiki.

6.1 Content for the KM Portal

Any knowledge asset to be hosted in this platform (document, pictures, etc.) can be uploaded manually, but this would be time consuming if we are speaking about thousands of documents. Using the aforementioned feature of automatic uploading, the process is made easier. More precisely, for this to work, two elements are needed:

- Common repository to share between the KM Toolkit administrator and the sponsor (i.e., the project instantiating the tool).
- Import feature in the KM Portal: a mechanism to read CSV files with file descriptions and locations and to upload each of them into the portal, as a content item. This is done via the Feeds module for Drupal.

Having that, the project sponsor has to make sure that they populate the shared drive with the files to keep, under the corresponding categories. When this is done and upon notification, the KM Toolkit admin will process the folder structure and upload it, saving an enormous amount of time to both parties.

6.2 Content for Wiki

Another lesson learned from the ATVCAP experience is that the creation of articles in the MediaWiki format can be cumbersome at first. Therefore, in order to ease the creation of MediaWiki articles, users are allowed to do it off-line and within a regular Microsoft Word document. Then, the MediaWiki administrator processes them with a customized content import mechanism and uploads them to the platform. This simple procedure improves the level of user collaboration in the platform by eliminating the MediaWiki learning curve and by using automated conversion and upload mechanisms.

7 WAY FORWARD

In order to guarantee scalability and sustainability, further work is recommended. In particular:

- *Connection module:* To provide a robust and complete connection module that would allow access to all the information of a network of interconnected instances of the KM Toolkit.

- *Improved taxonomy management system:* To keep taxonomies aligned and avoid duplications within each element of the toolkit..

- *Network monitoring:* To guarantee the network's health it wise to active monitor the system.

- *Access restrictions and content types:* Customizations must guarantee the compatibility of the tailored system. Therefore, it is a need analyse and implement the best way to allow this incremental developments within the network.

- *Corporate single sign-on:* For an even better integration of the toolkit in the corporate technological ecosystem.

- *Distributed software update mechanism:* To synchronize the updates in different instances.

- *Key performance indicators (KPIs):* It is of a big added value having measurements on performance, usage, and other relevant actionable data from which improve the system.

8 RELATED WORK

Arriving to the KM Toolkit has been a gradual and incremental process that started several years ago with different independent pilots (Guerrucci et al., 2012; 2014). The following solutions have been especially influential in the final result.

8.1 KM Portal

The KM Portal is a gateway to information and tools created after user feedback requesting a single entry point. It is a platform that enables users to self-organize in communities, publish contents of different nature (articles, videos, etc.) and in general, it is a feature-rich environment for many content related activities.

It was implemented using Drupal 6, integrated to the corporate sign-on with LDAP connectors and partially integrated with an external general purpose search engine.

8.2 XDot Search Engine

This is a commercial search engine based in Lucene technology that has been for some time used as a stand-alone product to search among a variety of repositories (e.g. web sites or file systems) and also integrated as the "external search" feature from the KM Portal. Its search capabilities have proven to be very intuitive and useful, though it not being an open source solution has in certain cases been a barrier to adaptability in a changing environment.

8.3 ATV Portal

Built upon the KM Portal, the ATV Portal uses the same technology, Drupal 6, and customizes the interface adding the following main changes:

- Content is accessed by a double filter of "content type" plus "knowledge area".

- Communities function disappears. It was considered that a complex community management system was not necessary.

- Batch import. To automatize the transfer and categorization of documents from the original repositories to the ATV Portal.

8.4 ATVpedia

The ATVpedia is a MediaWiki instance, working as a stand-alone platform and loosely connected to the ATV Portal. The goal of this platform, as part of the ATVCAP project (Guerrucci et al., 2014), was to include a detailed description of the different knowledge areas of the ATV mission, that is, to provide one article per element in the ATV taxonomy. Further work has been planned to integrate the ATVpedia content in the ATV Portal's search results, so the user experience is more similar as the one provided by the KM Toolkit.

9 CONCLUSIONS

The ATVCAP project set the first stone in the road to reuse KM tools in an unified platform. Building upon this experience, the KM Toolkit offers a general solution that is adaptive to local environments, which can tailor it to their needs. A KM system that is supported by a replicable toolkit to achieve fast and progressive deployments of KM across the organisation seems to be a promising option. From the technical point of view, open source can help with this strategy by enabling teams

to provide continuous enhancements.

The initial KM toolkit encompasses a search engine, a repository for knowledge assets (videos, lessons learned, documents, etc.), a taxonomy, access control, functionality to manage competencies and an integration framework to act as a glue to put them to work together.

Some requirements are to be imposed though: any modification to the original instance shall keep the connectivity, so a network of knowledge platforms is always in place. Also, it is important to follow a service oriented approach and a reference architecture, in order to have clear criteria when adding new components or replacing existing ones. Although integrating different technologies takes significant effort at the beginning, it is later compensated with faster deployments and the involvement of the end user.

REFERENCES

Mugellesi Dow, R., et al., 2014, Knowledge capture in ESA projects: The ATV case, *IAC-14, D5,2,11, Toronto, CA.*

Guerrucci, D., et al., 2014. Phased Approach to a Knowledge Management Network, *KMIS 2014, Rome, IT.*

Mugellesi Dow, R., et al., 2013, Knowing More About Knowledge Management At ESA, *KMIS 2013, Algarve, PT.*

Guerrucci, D., et al., 2012. Technological Aspects of the KM System in ESA, *4th International Conference on KM, Toulouse SpaceShow 2012,* Toulouse.

Mugellesi Dow, R., et al., 2011. Corporate Knowledge Management, *IAC-11, D5,2,1,* Cape Town *, SA.*

Mugellesi Dow, R., et al., 2010. Managing Knowledge for Spacecraft Operations at ESOC, *Journal of Knowledge Management, Vol.14, No.5.*