FRACTAL SYSTEM & PROJECT SUITE: ENGINEERING TOOLS FOR IMPROVING DEVELOPMENT AND OPERATION OF THE SYSTEMS

A. Pérez-Calpena[a], E. Mujica-Alvarez, J. Osinde-Lopez [a], M. García-Vargas[a]
[a]FRACTAL SLNE. C/ Castillo de Belmonte 1, bloque 5 Bajo A. E-28232 Las Rozas de Madrid (Spain);

ABSTRACT

This paper describes the FRACTAL Systems & Projects suite. This suite is composed by several tools (GECO, DOCMA and SUMO) that provide the capabilities that all organizations need to store and manage the system information generated along the project's lifetime, from the design phase to the operation phase.

The amount of information that is generated in a project keeps growing in size and complexity along the project's lifetime, to an extent that it becomes impossible to manage it without the aid of specific computer-based tools. The suite described in this paper is the solution developed by FRACTAL to assist the execution of different scientific projects, mainly related with telescopes and instruments, for astronomical research centres. These tools help the system and project engineers to maintain the technical control of the systems and to ensure an optimal use of the resources.

GECO eases the control of the system configuration data; DOCMA provides the means to organise and manage the documents generated in the project; SUMO allows managing and scheduling the operation, the maintenance activities and the resources during the operational phase of a system. These tools improve the project communication making the information available to the authorized users (project team, customers, Consortium's members, etc). Finally and depending on the project needs, these three tools can be used integrated or in an independent manner.

Keywords: Systems, projects, suite, system engineering, configuration control, documents, operation, maintenance.

1. INTRODUCTION

This paper provides an overview of the FRACTAL System & Projects Suite, which is currently composed by the following applications:

- GECO (Configuration Management Tool)
- DOCMA (Documentation Management Tool)
- SUMO (Operation and Maintenance Tool)

First, in section 2, we are going to explain why the FRACTAL System & Project suite is being developed. Later, in section 3, we will describe each tool identifying the initial requirements and outlining the capabilities achieved. Finally, in section 4, we will conclude about the status of the current suite and its future development.

2. WHY THE FRACTAL SYSTEM & PROJECT SUITE IS PRODUCED?

The amount of information that can be generated and must be maintained in a company, project or organization keeps growing, to an extent that it becomes virtually impossible to manage such information without the aid of specific computer-based tools. This is a very well known situation in all organizations and the FRACTAL starting point for beginning to develop the suite described in this paper.

FRACTAL is a company founded in 2005 and dedicated to carry out engineering and scientific projects mainly related with telescope and professional astronomical instrumentation, whose main customers are universities and research centres. It is also important to mention that most of the people working or collaborating in our company is

geographically distributed, which makes even more necessary to keep the project's information in an electronic format, well organized and easily accessible to all people involved.

We have identified the need of having computer tools to provide help in the following areas:

- **System engineering and configuration control.**

  System Engineering is defined as the interdisciplinary effort that governs the global technical effort done in a project framework to transform the initial requirements into the final system. Therefore, Systems Engineering provides the basis to establish a good organization during the technical development of a project in order to fulfil both the system requirements and the project schedule and budget. This discipline is always essential for the success of any project, especially the more complex ones, which include different professional skills and whose partners and working groups are often geographically distributed.

  The System Engineering group must produce the System Engineering plan for the development of a system. During the implementation of such plan, the system configuration data are generated: Product Tree (PT) elements, requirements, interfaces, specification documents, verification matrix, configuration control records, non-conformities records, etc.

  A Configuration management tool helps to store the configuration data and should allow establishing links between the data, which will help to control these data and to track correctly the system development in order to ensure that the system will be successfully integrated.

  A Configuration management tool also easies engineers to access the configuration data. Sharing the system information between all people involved in a project is quite important for improving the project communication and essential when the members of the project are working at different locations.

- **Project documentation.**

  All organizations produce and store a good amount of documents. Normally, the number of documents is high enough to make a documentation management tool always a need. When several companies collaborate in a multi-discipline, complex project in a geographically distributed environment, a documentation tool starts to be even more important. In such a case, the number of documents generated and stored in the diverse work centers, with different versions, written and reviewed by several individuals gets unmanageable very fast and the need for a specific computer-aided tool becomes essential.

  Taking into account the geographically distribution of our personnel and the complexity of the projects we develop, we quickly identified the need to use a Documentation tool which allows us to keep and organize documents and to improve the internal communication among the different members involved in a project.

- **Operation and maintenance management.**

  A good Operation and Maintenance Plan guarantees the expected return of a system and optimizes the human and economic resources that must be used. The amount of activities and resources that must be considered in the operation and maintenance of a system makes us to be aware of the importance to develop a specific computer tool, which would easy the implementation of such a plan.

  Recently, FRACTAL identifies the possibility to be involved in the organization phase of a system. Considering also that FRACTAL has enough expertise in this area, as part of the FRACTAL team has been working in Systems Operations Engineering for several scientific projects, we decide to start by defining the needs of an Operation and Maintenance tool.

An additional requirement, common to all the applications of the suite, is that they must run on different environments and at different locations. In order to minimize the maintenance costs and the complexity, we should choose a platform independent technology, provide simple ways to install the software in a distributed environment and implement a user access policy and encryption features to protect the projects data.

3. FRACTAL SYSTEM & PROJECT SUITE OVERVIEW

In the following sections, we are going to describe the three tools that currently constitute the FRACTAL System & Projects Suite.

3.1 GECO: Configuration Management Tool

GECO is a Configuration Management Tool that provides the means to manage the configuration data generated in all phases of a project, i.e., not only during the design phases of a system but also during its integration, verification and, even later, when the system enters into operation.

The application has been developed with two objectives: firstly, to assist the System Engineering Group and Configuration Control Group to control and maintain the configuration items of a system and, secondly, to make this information available to other groups within the organization or the consortium in charge of developing and operating the system.

The initial requirements of the tool can be summarized in the following ones:

1. Provide a configuration data repository, where the entire configuration data generated during the project life-cycle is kept and can be accessed according to the access rights of each user. The configuration data that we desire to manage with the application are the following:
    - The Product Tree elements and parts.
    - The table of interfaces.
    - The requirements, interface requirements and the relationships between parent and child requirements.
    - The requirement verification matrix.
    - The configuration changes.
    - The non-conformities.
    - The operation anomalies.
    - The project documentation.
2. Provide the means for implementing the configuration data traceability. GECO shall allow establishing cross-relationships among the data stored in the tool: Product Tree element, Interfaces, Document, Requirements, Verification Matrix, etc. These links are critical for keeping the system's coherence.

    In particular, the tool must be able to manage the requirements traceability (parent-child relationships). This traceability can allow the engineer to analyze correctly how the changes requested in one part of the system may affect other areas and to ensure that:
    - The final system meets the high level requirements.
    - The system can be integrated.
3. Provide the means for implementing the system configuration management and the project quality control. At least, the tool shall provide the following capabilities:
    - To manage the lifecycle of the configuration changes, non-conformities and anomalies.
    - To notify about their state transitions by email to the related users.
    - To introduce automatically the technical requirements into the verification matrix.
4. Generate automatically the requirements documents. So, in order to maintain a coherent set of requirements and to prevent duplicated information, the tool must generate the requirement document directly from the requirements introduced in the tool.

    We consider that a requirement document is needed because it is an important view of the requirement set, mainly, to interface with external parties, such as contractors and customers, and to comply with the intended project milestones and reviews.

GECO uses a public relational database to store the data and manage the backups. It provides a graphical user interface developed in Java, which makes the application platform independent, i.e. it runs on UNIX, Linux and MS-Windows operating systems.

Taking into account the requirements, GECO has been developed as described in the following lines:

- **Data organization:** The data is hierarchically distributed following the Product Tree of the systems. Each Product Tree element must have a code that identifies it uniquely. Using a consistent, manageable coding convention is a paramount, as this code is the base for assigning codes to the rest of the configuration items: documents, requirements, interfaces, etc. The user gives the Product Tree element code, and then the tool generates automatically the codes of the remaining configuration items.
- **Data views:** The tool allows the users to visualise the data in two ways (see Fig. 1):
  - A summary data view which displays the most relevant record fields of the selected item in a table. In this view, the user can navigate through the elements of the Product Tree, which is always displayed in the left side and provides an intuitive view of the system's breakdown structure, to access, with a few mouse clicks, to a Product Tree element, its requirements, interfaces, associated documents, etc.
  - A full data view that displays all record fields for the selected item in a form. Specifically, the relationships across requirements, documents, users, etc are shown in the full view.

  In both views, GECO provides all the filtering and sorting capabilities that are needed to search the desired data.

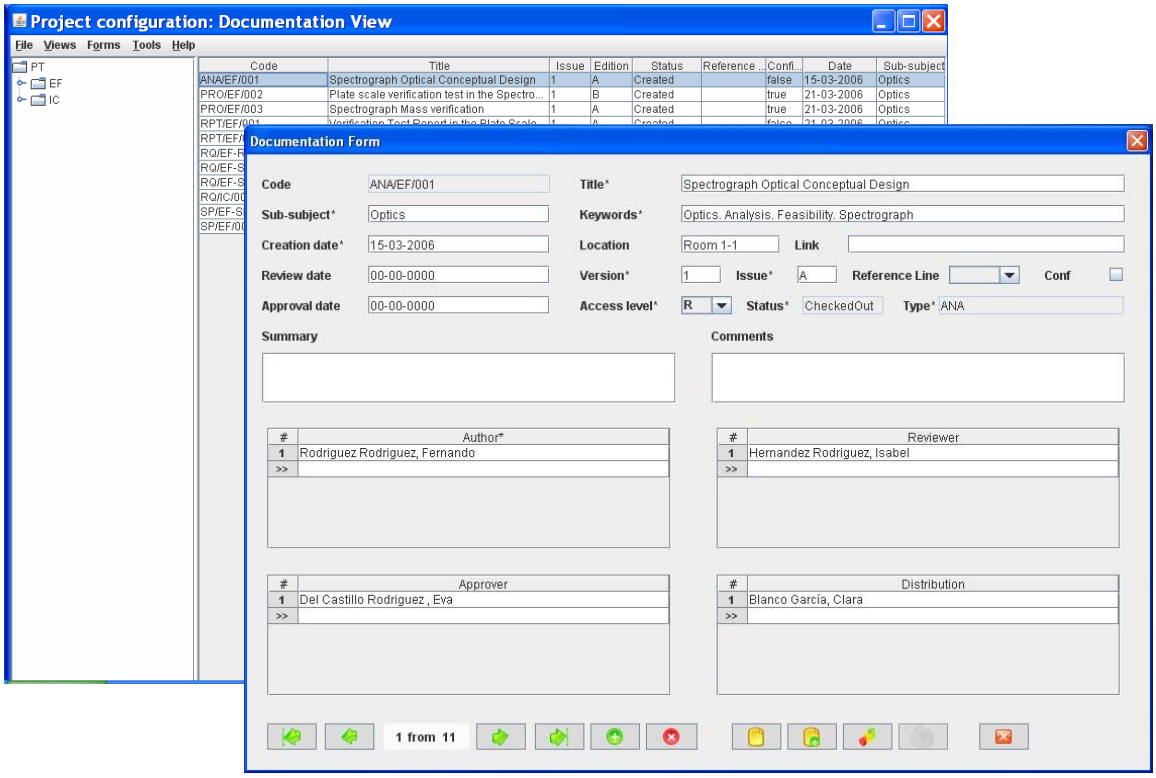

Fig. 1. GECO summary data views and forms.

- **Data input forms**: The tool provides forms for adding or modifying the data associated to each configuration item.
  - PT Elements form: where the definition of the Elements of the PT can be introduced and be modified, the related documents, etc.
  - Parts form: where the definition of the Parts can be introduced and modified, its related documents, etc.

- Requirements form: this form allows the user to enter new requirements, or to modify the already existing requirements, associated to a certain PT element. A major functionality of this form is the ability to track down each requirement to other requirements, what is crucial for identifying what a configuration change or non-conformity actually involves. This utility is also important for assessing the impact of the proposed change in terms of scope, schedule and price.
- Interfaces form: this form allows the user to define and to modify the interfaces between the PT elements, thus showing which elements are related to one another.
- Verification matrix form: this form allows the user to enter and to track the results of the verification tests.
- Configuration Changes form: this form (see Fig. 2) allows the user to start new configuration changes and to add information to the changes that already are in course (conducted actions, changes of status, affected documents, etc). The process associated to the changes of configuration may range from very simple (when the change considered does not affect other elements of the system) to rather complex (when other elements or different work groups are involved). In any case, the tool helps to coordinate this process by sending the necessary notifications (by e-mail) to the users affected by the change, as the configuration change goes through its lifecycle.

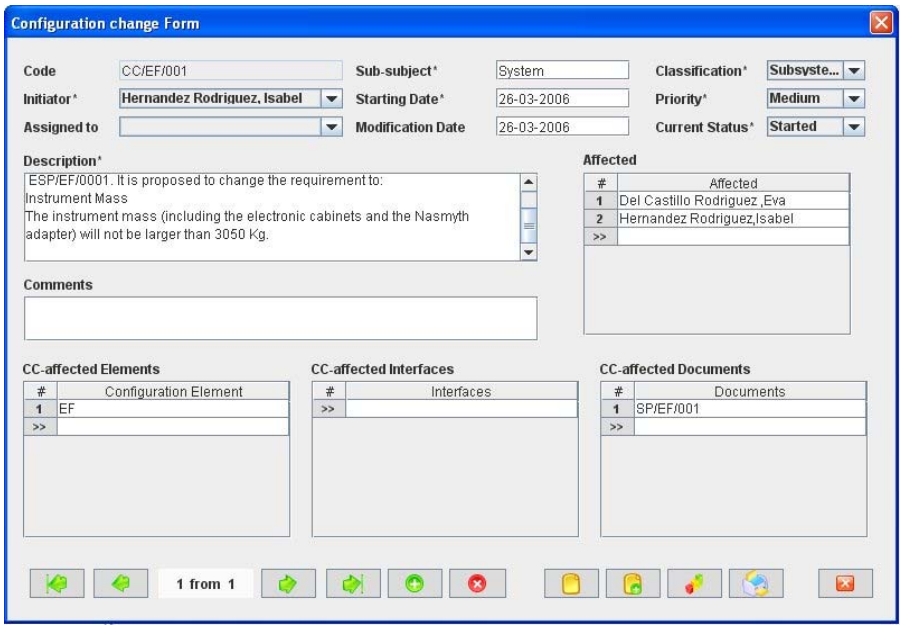

Fig. 2. GECO Configuration Change form.

- Non-conformities form: this form allows the user to start a new non-conformity, as well as to add new bits of information to the already existing non-conformities. Similarly to the configuration changes, the tool sends notifications to the users affected by the non-conformity as it goes through its lifecycle.
- Anomalies form: this form allows the user to start new anomalies or to add information to the already existing ones. Similarly to the configuration changes and non-conformities, the tool sends notifications to the users in charge of the anomaly during the anomaly state transitions and, also, whenever a new individual for investigating the anomaly is identified. The tool helps the user to track the evolution of the anomaly.
- Documents form: this form allows the user to enter or to modify the documents associated to the project and to connect them with PT elements. All documents introduced by GECO or DOCMA are visible from both applications. GECO allows modifying the document properties and gives access to the documents but does not contain the capabilities to manage the document life-cycle provided by DOCMA.

- Users form: this form allows the administrator to define new users or to modify the data associated to the already existing users. All users must have a permission level. The permissions implementation includes two types of permissions: the “admin” level, with all the privileges and able to change everything and the “read” level, for people who only need to consult the contents.

- **Additional utilities:** The tool provides a set of built-in utilities that are described below:
  - Search of child requirements: the tool displays the requirements that have been defined as child requirements of the one requested by the user (i.e., taking into the parent-child, traceability, relationships that were introduced in the Requirements form). This functionality is intended to be use whenever the user is searching for which requirements could be affected by a change in high level requirements.
  - Search of parent requirements (see example in Fig. 3): the tool displays the requirements that have been defined as parent requirements of the one requested by the user (i.e., taking into the parent-child, traceability, relationships that were introduced in the Requirements form). This functionality is intended to be use whenever the user is searching for which requirements could be affected by a change in low level requirements.

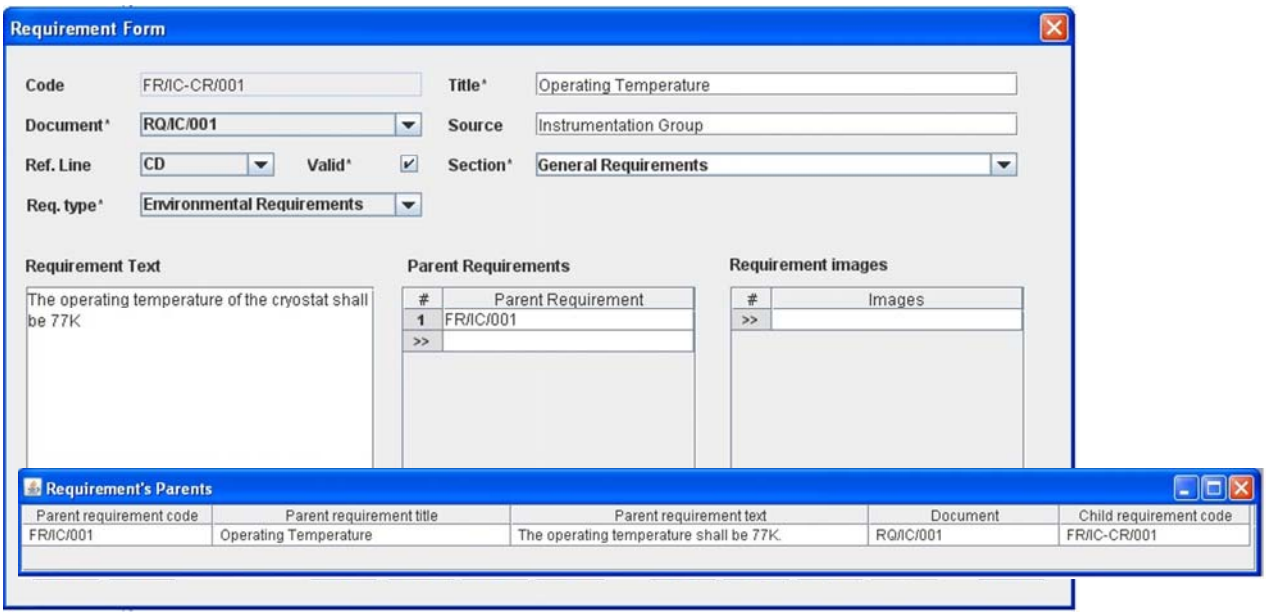

Fig. 3. GECO Searching parent requirement utility.

- Insert automatically the requirements in the verification matrix: all the technical requirement of one subsystem can be automatically introduced in the verification matrix when the user decides that such subsystem is ready for being verified.
- Automatic generation of the requirements document (see example in Fig. 4): the tool gathers the requirements associated to the selected document. These requirements are organized in sections (as defined by the user) and subsections (attending to the requirement types), which can be sorted as the user decides, in order to eventually generate the document section where the requirements are specified. Such section is generated using the OOXML (Office Open XML) standard and can be easily imported into the definitive formal requirements document.

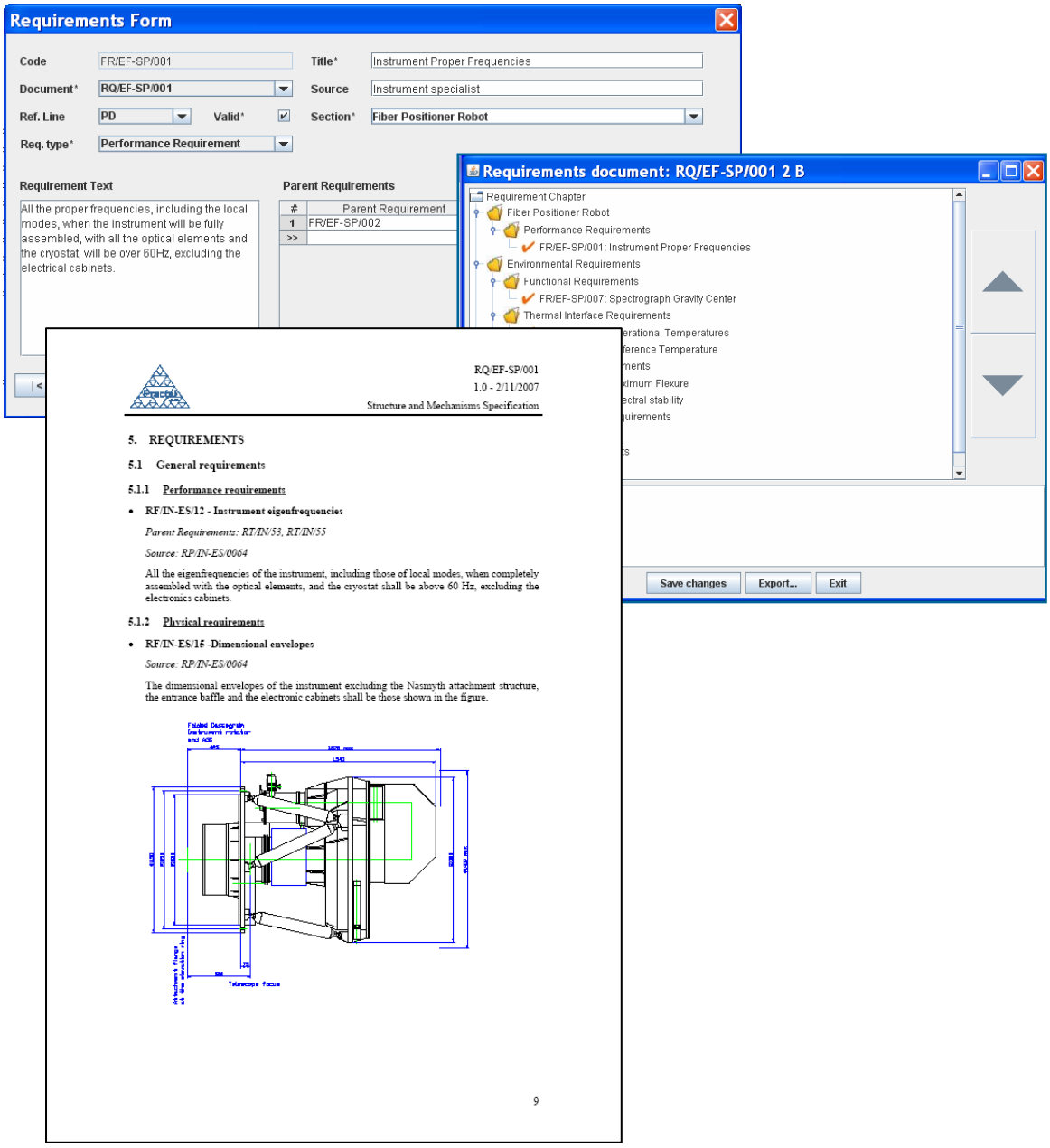

Fig. 4. GECO automatic generation of requirements document utility.

3.2 DOCMA: Documentation Management Tool

DOCMA is a Documentation Management Tool which provides the means to manage the documents generated in a company or organization during the whole document life-cycle.

The initial requirements of the tool can be summarized as the following goals:

1. Providing a documentation archive (electronic library) with the following capabilities:
    - Classifying and arranging the documents into projects.
    - Document searching based on several criteria: by project, authors, title, code, keywords, etc.

- Providing controlled access to projects and documents, according to the permissions defined for projects, documents and users including the possibility to download a read-only copy of a document.
- External access to the authorized user (i.e., provide access from an external network).

2. Manage documents during their lifecycle, which means to be able to:

- Create a new document; assign properties to the document, such as the authors, authorised readers, document type, etc.
- Controlled access to the document for modification (checking out and checking in).
- Controlled review and approval of documents.
- Keep all the involved users informed (via e-mail and internal notifications) about the document status, during its whole lifecycle: document creation, approval, new issues of the document, authorised users alterations, etc.

3. Provide the following system administration capabilities:

- Project administration: define new projects; set or modify their properties; end a project, removal or archive the project.
- User administration: define new users; modify their attributes and permissions in connection with each project; unsubscribe users.
- Database administration: data backups and restores.

DOCMA uses a public relational database and provides a graphical user interface based on WEB forms. This means that anyone having access to the Internet may gain access to a project's documentation, provided that this person has an account in the system and the suitable permissions have been granted to him/her. The main advantage is that there is no need to install any specific software on the client computers; the WEB browser is enough. Similarly, the user does not need to access the e-mail, as the notifications are managed by the Tool and made available to the user via the WEB-based interface.

In order to meet the objectives, DOCMA has been developed as described in the following lines:

- **Project management:** The tool provides a form for defining new projects, as well as for modifying the properties of the already existing ones.

  Similarly to the documents, the projects can also have a life-cycle (see Fig. 5), which includes finalising a project (i.e., the documents associated to the project can not be modified any more), reactivating a finalised project (i.e., the project will be active again), removing a project in the system and restoring a removed project.

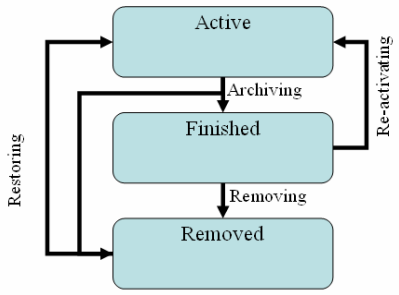

Fig. 5. DOCMA Project life-cycle.

  Users must have the right permission to be able to manage the projects as described in Table 1.

- **Document management:** The tool provides forms for creating new documents, modifying the properties of already existing documents, and managing each document during its lifecycle (see Fig. 6).

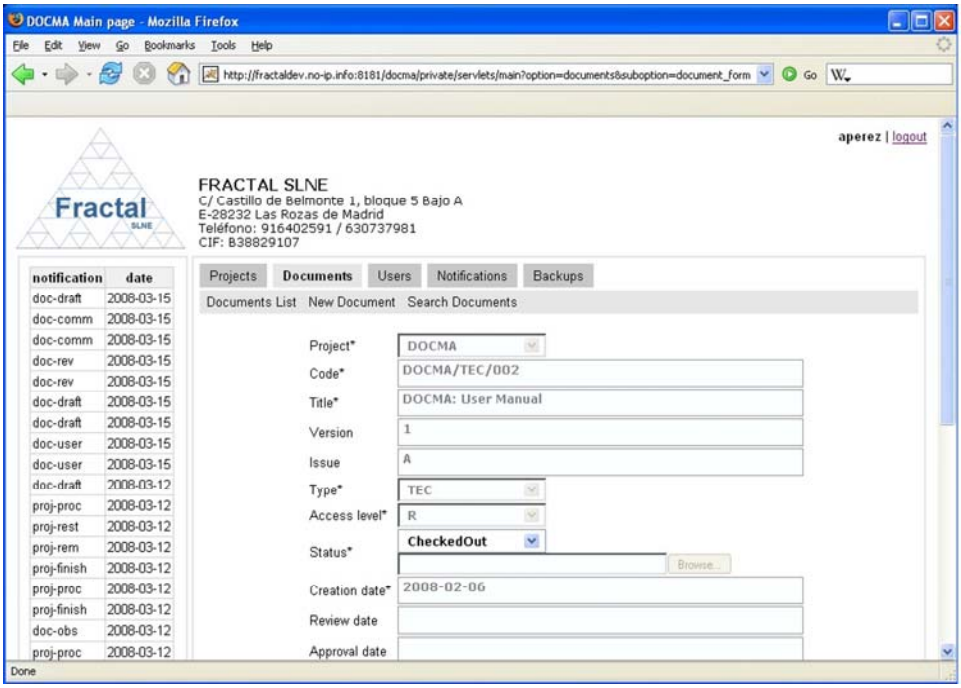

Fig. 6. DOCMA document status change.

The documents lifecycle (see Fig. 7) includes checking the document out (for editing it), checking the document in (to store a new draft or official version in the system), follow up the document review (i.e., formal documents have to undergo a review process, during which the tool sends notifications to the involved people allowing the users to enter comments), and accept in the system the approved version of the document. This lifecycle is repeated each time that a new version of an existing document is created. Finally, the documents can be declared obsolete or be removed from the system.

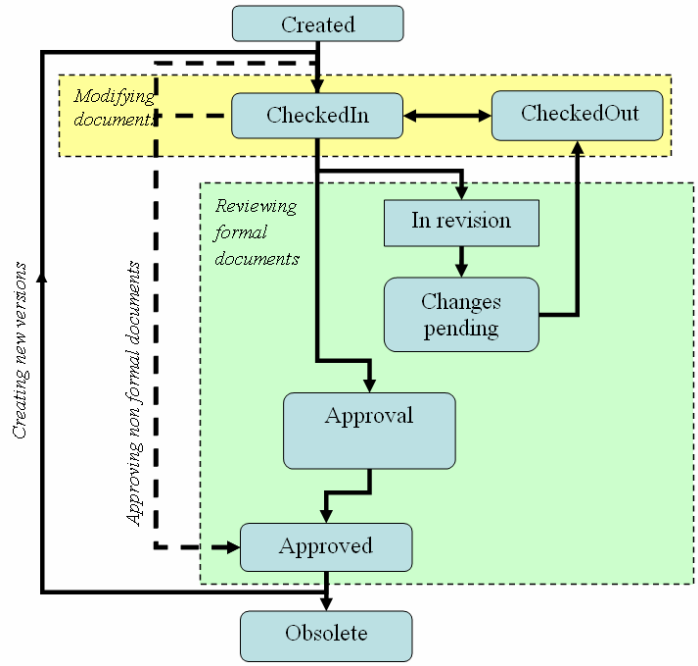

Fig. 7. DOCMA Document life-cycle.

The actions that users can carry out in a document are different according to the permission level granted to each particular user (see Table 1) and, also, according to the document status. It means, as shown in the Document life-cycle figure, not all transitions are allowed (i.e., the initial status of the document will be taken into account by the application to allow the user to reach the requested status).

All documents created by GECO are also accessible by DOCMA. In general, the configuration documents (i.e., requirement documents, specifications documents, interfaces documents and drawings) are created using GECO, as this tool allows to link this documents to the corresponding low level Product Tree elements and, afterwards, they can be managed (i.e., during the transitions of the documents lifecycle) using DOCMA.

- **User management:** DOCMA provides controlled access to all projects and documents in the system, according to the permissions assigned to each user. There are several levels of permissions as described in Table 1:

	Permissions		
Roles	General permissions	Project based permissions	Document based permissions
	Administrator	Project Manager Authorized user (all documents) Authorized user (only approved versions)	Authors Reviewers Approvers

Table 1. User permissions

A properly authorised user can visualise all the information relative to a project and also search projects according to several criteria (i.e., by people associated to a project, by project properties, etc.).

Similarly, a user authorised to access the documents of a certain project can browse the list of documents associated to such project (see Fig. 8), or make queries to the database for searching documents according to several criteria (i.e., by title, by authors, by keywords, etc). The authorized user may also get an electronic copy of the document or have it sent automatically to an e-mail address.

The administrator has the access rights for creating new users and for specifying the roles this user can adopt in the different projects registered in the system. Users that discontinue their relationship with the company or organisation are kept in the system to avoid inconsistencies on the data but their associated permissions are removed and the user can no longer enter the system.

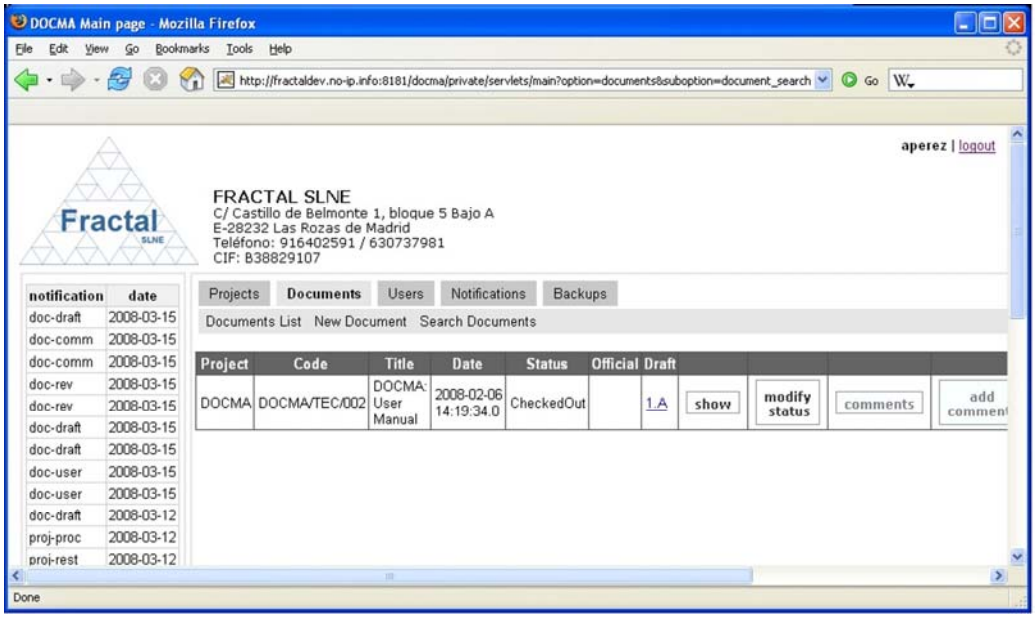

Fig. 8. DOCMA filtered documents list.

- **Notification management:** Each user can visualise and handle the notifications that this particular user has received from the system.

  The Tool includes a set of predefined notifications, both internal and external (via e-mail through SMTP protocol), that are generated automatically whenever a relevant event takes place. For instance: when a document changes its state, when a new project is created, when a new user is registered, when a user is associated to a project, etc.

- **Backup and restore functions**: The tool provides the administrator the means to perform backups and restores of the DOCMA databases, documents and Web applications.

3.3 SUMO: Operation and Maintenance Management Tool

SUMO is an Operation and Maintenance Management Tool which provides the means to manage the operation and maintenance activities and resources during a system operations phase.

The initial requirements of the tool can be summarized in the following objectives:

1. Provide an operation and maintenance repository, where all the information needed to operate and maintain the system is kept and can be accessed according to the access rights of each user.

   The data to be maintained will be the following:

   - Operation and maintenance activities. The tool will allow introducing all the information needed to characterize in detail the operation and maintenance activities to be carried out, which can:
     - Link an operation or maintenance procedure, where the activity to be done is described step by step.
     - Define the activity periodicity (if it applies).
     - Define links between activities (if any). It means, define time constrains between activities that must be executed one after the other.
     - Define the activity duration.
     - Define the activity priority.
     - Assign operation and maintenance resources (as defined in the list below) to the activities.
   - Operation and maintenance resources. The tool will allow introducing all the information needed to characterize in detail the operation and maintenance resources that are needed to perform the activities. The operation and maintenance resources includes the following ones:
     - Manpower
     - Equipments
     - Facilities
     - Spares and supplies

   This repository will include all the filtering and sorting capabilities that are needed to search the desired data (activities and resources) using different criteria: by type, by code, etc.

2. Generate an operation and maintenance schedule for a given period of time. The schedule generation will include several utilities as described in the following lines:

   - The tool must be able to generate automatically the schedule including the periodical activities and, also, the activities linked to other periodical activities.
   - The tool shall allow to introduce manually non-periodical activities (i.e., non-foresee activities such as correctives activities must be input manually).

- The schedule can be manually edited (i.e., the foreseen date of the activities can be manually changed)
- The tool shall provide the means to analyze that the generated schedule can be performed with the available resources.

3. Provide an archive of the executed activities, which can be used to analyse the operation and maintenance plan efficiency.

This archive will include all the filtering and sorting capabilities that are needed to search the desired activities using different criteria: by date, by code, etc.

4. Manage spares and supplies. The tool will save the information related with the use of spares and supplies and the acquisition of new ones. This information will be very useful to analyse the spare and supplies consumption.

5. Send notification whenever needed (i.e., pending activities have not been executed at the defined date; spares have reached the stock limit, etc.)

6. Provide the configuration control and quality capabilities that are needed during the operation phase of the system.

In order to have a self standing tool, SUMO should duplicate part of the capabilities provided by GECO. In particular, the following three items must be also managed by SUMO:

- Anomalies
- Configuration changes.
- Inventory management (i.e., Parts).

This application is still in the inception phase.

4. CONCLUSIONS

The “FRACTAL Systems & Projects” suite includes three applications intended to manage most of the information that is generated during the projects lifetime, from the start of the system design to the operation phase.

The three tools can be used in an independent manner. In this case, each application will provide in the related area (i.e., System configuration, Documentation or Operation and Maintenance management) the capabilities described in the previous sections. Or, they can be used integrated, as the information managed by all applications is stored in one single database, which is accessed by the three tools.

Future development of this suite could include not only improving the capabilities of the tools already described but also adding new applications as a Project control tool.

REFERENCES

[1] Benjamin S. Blanchard, Wolter J. Fabrycky, “System Engineering and analysis” (1998).
[2] Standard ECSS-E-10 (ESA System Engineering Standards)