Abstract—Currently managing an enterprise or institution would be impossible without the use of Information Technology. Providing quality services that allow the user an easy and transparent handling of information is a daunting task for anyone who provides it. ITIL is a framework that can help to provide solutions that improve practice management, integration and manipulation of information. ITIL describes "WHAT TO DO" and not "HOW TO DO", explaining in detail all the processes but does not indicate how they should be implemented.

In this document we proposed to implement Change Management and Configuration Management processes considered to be a priority to support the daily management operations of an organization.

Using Action Research Method, this thesis was applied to a scientific and technical association of public utility whose research, development and implementation is also focused in the field of managing a distributed computing infrastructure.

A prototype system was built, and now is possible to make faster changes on the hardware infrastructure. There is also the guarantee that, by the end of the change, this information is recorded in the CMDB correctly and updated.

A. Motivation

Nowadays the Internet is used on a global scale and is the main vehicle to process information. Modern science problems require large computation time to be solve and generate a high volume of data. To face this challenge the scientific community and enterprises use solutions of distributed computing such as the Grid [1] [2] and Cloud [3] Computing.

With the adoption of distributed computing, Data Centers become harder to manage: they are responsible for managing their local infrastructure but they are also part of broader networks that can be private, national or even global. In this context, Data Centers are responsible for managing the equipment (physical and virtual) and services that can be used by a large communities of users, and should be constantly monitored to ensure the quality of services required by clients of joint infrastructure.

In sum, to administer a modern Data Center is necessary to handle all its components: the Physical and network infrastructure, the services provided and also understand the relations and dependencies among them.

In order to help to solve this complexity there is the ITIL [4] [5] [6] [7] [8] [9] "a set of best-practices for Information Technology (IT) service management that focus on aligning IT services with the need of the business."

ITIL is a vast compilation, with several areas, but the ITIL Service Support area is the practice of disciplines that enable IT services to be provided. Service Support deals with the processes needed to maintain daily operations, and for this reason should be the first to be implemented.

ITIL service Support include two processes inextricably linked: Change Management and Configuration Management. The CMDB (Configuration Management Database) is a repository of data related to all components of the information system and a central part of Configuration Management process.

B. Research Questions

The goal of this thesis is to understand the processes involved when managing the physical and network infrastructures of a Data Center: what are the processes involved,
which are considered do be a priority, who are the persons involved in the processes, and their responsibilities.

Is also part of the scope of this thesis to understand what tools are on the market that can automate and implement the processes found.

In sum the goal is to find an answer to the following questions:

1) How to create the Change Management process to allow, in a coherent and common manner, to manage changes in the physical and network infrastructures?
2) How to create the Configuration Management process and the CMDB, so that the information about the infrastructure, the relations between its various items are known, are globally accessible and updated?
3) What tools can be used and how they can be adapted to implement Change and Configuration processes?

C. Research Method

The research method used in developing the thesis is Action Research [10]. This method indicates that the researcher should try a theory with individuals in real environments. The organization where this thesis applies the theory is the Data Center of Laboratório de Instrumentação e Física experimental de Partículas (LIP) in Lisbon. The context of this institution is explained in Section III.

II. RELATED WORK

The related work of the thesis consists of studies comparing tools for managing and operating the network and services infrastructures of a Data Center.

Several open source solutions for network monitoring are analyzed: Nagios, OpenNMS, and Zenoss [12] and Hyperic, Lithium e o Zabbix [13].


This Section ends with a study [26] that compares software products according to the capacity they have to implement Change Management and Configuration Management processes.

III. MY WORK

In this section the LIP institution is presented, and the main problems in the procedures undertaken to manage the Data Center of the institution are identified. Given the assessment made to LIP Data Center a set of Functional, Non Functional and Business requirements were proposed. In order to accomplish these requirements two systems were designed (according to ITIL): Change Management System and Configuration Management System. The section ends with the description of the technologies involved, (Request Tracker and Zenoss) and how they are adapted, to implement the proposed systems.

A. LIP Context

LIP is a scientific and technical association of public utility that has for goal the research in the fields of Experimental High Energy Physics and Associated Instrumentation. The main research activities of the lab are developed in the framework of large collaborations at CERN and at other international organizations and large facilities in Europe and elsewhere, such as ESA and NASA.

In Lisbon and Coimbra there are two Data Centers hosting the IT infrastructure of the institution. The computation team is also responsible for coordinating the operation of the Iniciativa Nacional Grid (INGRID) and for managing the computational systems of the Grid Central Node in collaboration with FCCN.

B. Main Problems in Managing LIP Data Centers

A large number of interviews to the computation team were made in order to assess how the daily operations were made. The main conclusions were:

- the documentation of how processes are made is not up to date
- the database that holds the IT infrastructure is obsolete
- each functionary holds in his private area the documents and procedures needed for doing his tasks
- every intervention is performed in a ad hoc basis without properly evaluating the impact results
- The flow of information needed when tasks are performed for several people is not being recorded

C. Functional, Non Functional and Business Requirements

The main business requirement is that the systems to be designed should be implemented using open source technologies, and capable of integrating technology already existing in the institution. The main functional requirement is that the system to be implemented should allow the planning of interventions (adding, replacing, remove) in the IT infrastructure.

D. Change Management System

Planning an intervention in the IT infrastructure is planning a change to be made, so a Process for Change Management composed by several activities was designed. (see Figure 1) The system functionalities were established and also the use cases of the system. (See Figure 2)

E. Configuration Management System

When a change is made the IT infrastructure changes. One or several configuration items can be altered. The CMDB is the repository that holds these items and is a part of Configuration Management process.

To achieve this, the Configuration Management Process for LIP was designed. The activities that composed the process are: Planning, Identification, Specification, Control and Auditing and Verification.

In the Planning phase, the configuration items and the CMDB were identified.

The system functionalities were established and also the use cases of the system. (See Figure 4)
F. Request Tracker to Implement Change Management

Request Tracker (RT) [29] is a tracking system which can be used for bug tracking, help desk ticketing, customer service, workflow processes, change management and network operations. In the context of this thesis is used to implement change management system and managing the workflow.

It has the following qualities: accessibility, ease of use, multiuser, ability to track history, immutable history, flexible views, access control, dependency management, notifications and customizable workflow.

In RT each request or piece of work is called a Ticket. This ticketing system allows to register an event or a ticket, assign an owner, or person responsible to the ticket, assign additional interested parties to the ticket, track changes to the ticket, launch activities based on ticket status and priority among others.

RT Configuration: RT was configured according to the proposed change management process. The main convention is based in the correspondence between a Request for Change (RfC) and a ticket in a queue that aims changes in the hardware infrastructure; so all changes will be activities related to hardware attributes devices. The ticket structure was augmented with extra custom fields in order to support the attributes of a RfC and also to support the device name and type which will be changed. Table I summarizes the custom fields in a ticket RT:

<table>
<thead>
<tr>
<th>Formalismo RfC</th>
<th>Ticket RT</th>
<th>Preenchimento</th>
</tr>
</thead>
<tbody>
<tr>
<td>RfC Number</td>
<td>Ticket Id</td>
<td>Automatic</td>
</tr>
<tr>
<td>Creation Date</td>
<td>Created</td>
<td>Automatic</td>
</tr>
<tr>
<td>Expected Start Date</td>
<td>Starts</td>
<td>Automatic</td>
</tr>
<tr>
<td>Start date</td>
<td>Started</td>
<td>Manual</td>
</tr>
<tr>
<td>Expected End Date</td>
<td>Due</td>
<td>Automatic</td>
</tr>
<tr>
<td>Description</td>
<td>Subject</td>
<td>Manual/Mandatory</td>
</tr>
<tr>
<td>Priority</td>
<td>Priority</td>
<td>Automatic</td>
</tr>
<tr>
<td>Urgency</td>
<td>Urgency</td>
<td>Manual/Mandatory</td>
</tr>
<tr>
<td>Category</td>
<td>Category Level1</td>
<td>Manual/Mandatory</td>
</tr>
<tr>
<td>Category</td>
<td>Category Level2</td>
<td>Manual/Mandatory</td>
</tr>
<tr>
<td>Change Type</td>
<td>Change Type</td>
<td>Manual/Mandatory</td>
</tr>
<tr>
<td>Device Name</td>
<td>Device Name</td>
<td>Manual/Mandatory</td>
</tr>
<tr>
<td>Device Name</td>
<td>Device Name</td>
<td>Manual/Mandatory</td>
</tr>
</tbody>
</table>

Table I: Correspondence Between A Ticket and A RfC.

Workflows in RT are implemented using a mechanism of scripts, allowing the definition of a work hierarchy.

When a change is closed, a configured action take place in RT, and the description of the hardware device and his attributes is written in a XML file.

G. Zenoss to Implement Configuration Management

Zenoss [34] is a open source system for managing servers and networks. Zenoss monitors the IT infrastructure, including the network, the servers, air conditioning, heating systems, and it also monitors performance and the availability. It has a CMDB of the IT hardware and software infrastructure.

Larry Klostboer [33] indicates two approaches that can be used to introduce the devices in the CMDB: the first approach (waterfall) is to gather as much of data as possible from throughout the environment, and then integrate and link data, to assemble a complete configuration management picture. This is a more direct approach, but results in a overwhelming amount of data, which delays the completion of the task. The second approach (trickle) is to create integration points in each of the key areas of operational processes that are likely to deal with configuration data. By executing normal operations with these integration points, data related to configuration items trickles into the CMDB as those items change. This is less risky approach, but delays the benefits of having a complete image of Configuration Management process. The second approach is the one adopted in this thesis: a change in a IC under Configuration Management process, always arises as a result of a change request certified and approved. Subsequently the information related to the IC is introduced in Zenoss CMDB.

1) Implementation of Activities of GC Process in Zenoss:

Identification, Specification and Control: Zenoss provides a suitable domain model for the LIP institution. If in the course of activities, this model needs to be extended. Zenoss provides methods (Zenpacks) that allow the creation of new device classes and new methods to monitor them.

So, the Configuration Manager must take in consideration all the activities proposed for Configuration Management process and carefully assess what steps should be undertaken in each activity and if there is the need to extend the model.

The Configuration manager should only trigger the Control Activity after ensuring (covering other activities proposed) that there are no inconsistencies between the information to be added/modified/removed and the contents of the CMDB.

It is possible to introduce information in Zenoss in three different ways:

1) through the graphical interface
2) through the interactive interpreter running on the command line, called Zendmd
3) through a dedicated API

In order to automate the process of inserting contents in Zenoss CMDB, a python application which uses the features provided by Zenoss API was implemented. This application is able to insert XML content provided by RT via http, allowing the Configuration Manager to automatically by executing the application in the command line, to establish connections to Zenoss CMDB and insert new devices.

Auditing and Verification: The tool itself can trigger the Auditing activity taking into account the capabilities to discover and monitor devices. Zenoss is able to discover various types of devices, scan and understand configuration information and provides modeling capabilities, and is extensible.

The Zenoss provides several ways to find and introduce devices and their components: manual discovery and automatic discovery.
IV. Evaluation

This section discusses the results achieved by assessing whether the research questions presented in Section I-B are answered.

1) Answer to Question 1:
A Change Management process, according to ITIL framework, was proposed: the activities were established, roles were created, defining the persons responsible for executing them, and changes were categorized. The main features of a system, that are aligned with the common practices and key procedures of LIP institution and that implements the proposed Change Management were identified.

2) Answer to Question 2:
A Configuration Management process, according to ITIL framework, was proposed: the activities were established, roles were created, defining the persons responsible for executing them. The assets of the institution that must be maintained in the CMDB, and the flow of implementation of CMDB were defined.

3) Answer to Question 3:
The Change Management system was achieved using Request Tracker tool: its large capacity of adjustment to heterogeneous environments, the flexibility in configure workflows and the simplicity of configuration allowed the implementation of Change Management aligned with ITIL framework.
The Configuration Management system was achieved using Zenoss tool: Zenoss domain model proved to be appropriate to implemented in the LIP institution. The features provided, such as monitoring and auditing proved to be an asset in the implementation process.

V. Conclusions

The main questions that this thesis aims to answer have been solved. There is a prototype system in operation and knowledge in LIP institution that enables to continue future developments of the system.

It is concluded that the Change Management and Configuration Management systems were designed and implemented: it is now possible to make changes on the hardware infrastructure, where each participant in the workflow knows what is the status of the task. There is the guarantee that at the end of the change, this information is recorded in the CMDB and updated correctly.

VI. Figures

Figure 1. Proposed Change Management Process.

Figure 2. Use Cases for Change Management System.
Figure 3. Activities of Configuration Management Process proposed to LIP.

Figure 4. Use Cases for Configuration Management System.

Figure 5. Ticket Template configured to Perform a Change.
REFERENCES


[40] www.zope.org