"Linked Open Apps Ecosystem to open up innovation in smart cities"
Project Number: 297363

Deliverable:	**D4.2 System Operation Adaptation**
Version:	**V03**
Delivery date:	**08/02/2013**
Dissemination level:	**PU**
Author:	**RETEVISION**
Reviewer:	**PMO**

Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise.

Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Summary

This deliverable contains all additions, updates, integrations and developments made in the system operation of the Platform.

Administrator need to have real time information concerning the network in order to control the service level agreement for the iCity Platform.

DOCUMENT HISTORY

Version	Date of issue	Status	Content and changes	Modified by
V01	05/10/2012	Draft	Creation	Retevision
V23	20/12/2012	Draft	Defined prototype	Retevision
V03	06/02/2013	Draft peer review	Final draft version peer review.	Retevision

Document contributors

Partner	Contributor
Retevision	Carmen Vicente, Javier Marcos, Luis Moreno and Elena Villa

TABLE OF CONTENTS

1. Introduction 6

1.1 Purpose 6

1.2 Scope 6

1.3 Overview 7

2. System Management Definition 8

2.1 Positioning vs. Existing solution 8

3. iCity Operation System Prototype 10

3.1 Visualization 13

3.2 Monitoring 16

3.2.1 Data capture and collection 16

3.2.2 Monitoring 18

3.3 Inventory 18

3.3.1 Documentation management 19

3.3.2 Network and service management 20

3.3.3 Catalogue management 20

4. Conclusions 21

TABLE OF FIGURES

Figure 1: Operation System Prototype 10

Figure 2: Configutation process 12

Figure 3: Recollection process 13

Figure 4: Events console 14

Figure 5: General information related an alarm event 15

Figure 6: Monitoring module – Data capture and collection 17

Figure 1: Monitoring module – Monitoring 18

Abbreviations and Acronyms

Acronym	Description
WPX	Work Package X
D4.x	Deliverable 4.x
SSH	Secure Shell
FTP	File Transfer Protocol
NFS	Network File System
XML	Extensible Markup Language
SNMP	Simple Network Management Protocol
SMS	Short Message Service
SOS	Sensor Observation Service
SAS	Sensor Alert Service

1. Introduction

This document reports the work achieve in WP4 regarding with system operation adaptation.

1.1 Purpose

The aim of this deliverable is to set up the first version of system operation prototype, which has the main goal of provide real time information concerning the cities infrastructure integrated in the iCity Platform. Also, the system management has to be able to work with different platforms and has to provide access to them identifying each one of the external authentication processes.

Finally, it is important to mention that this document will be alive and modified during the whole life of the project in order to adapt its contents to the final iCity platform architecture. It will gather in the following deliverable:

- D4.7 (M24).
- D4.12 (M36).

1.2 Scope

In the first section of the deliverable is mainly focused on the different integration systems operation, in order to look for the best operation tool, which covers the iCity requirements. In the second section is defined the different functionalities of the system operation.

As a result, the iCity platform first version prototype adapts a system operation which will be improved with more functionalities and features in future versions of the iCity platform prototype.

1.3 Overview

First part of the deliverable is an analysis of systems operation solutions. The analysis presents existing solutions and also elaborates comparisons, in order to establish the best guidelines for deploying system operation adaptation to iCity platform. Regarding System operation, not only commercial solutions have been analyzed. Thus also open source solutions have been taken into account.

Finally, the deliverable is focused on the description of the system operation adaptation to iCity prototype.

2. System Management Definition

2.1 Positioning vs. Existing solution

Nowadays, there are a huge set of monitoring systems, comercial solutions and open source solutions, and then the analysis covers the needs of iCity architecture defined in WP3 in the most optimal way.

Some of the more remarkable monitoring solution for such systems would be:

- Netcool.
- HP NNM.
- Pandora FMS.
- Nagios.
- Zennos.

NETCOOL is a comercial solution based on Tivoli Netcool Omnibus, which gets alarms and events from different systems, provides a set of features to enrich these events and applies complex rules in order to reduce the amount of events that this type of systems receive. Licensing is established by collector and not for each device. This type of licensing is an important advantage than others, so the price of global solution is lower.

HP NNM is a comercial solution that provides a discovery of infrastructures, monitoring systems, network devices and another type of elements. This system solution provides a SNMP monitoring, which is pasive monitoring using a traps reception or active monitoring using MIBS of devices. This system provides a threshold configuration that could be static or dynamic.

PANDORA FMS allows communication within elements through agents, which it is posible to analyze the status and performance of different parameters provided by sources. All communication is done via SSH, FTP, NFS or XML conector to transfer data that it is stored

in a MySQL data base. Data is stored in a central server that allows showing in a web interface.

NAGIOS allows alerting message in a proactive way, so sends an email or SMS or instant message. Finally, this system has the enough intelligent to an event can create an automatic action to solve the incident active by an alarm before client or user notice about it. The environment of this solution is based on linux and is composed about external plugins that can be programmed in bash or Perl giving a high flexibility.

ZENOSS is a monitoring solution base don Open Source Software for monitoring the availability of the components. This solution enables proactive monitoring of elements, thus allows detecting, reporting and resolving problems that affect the correct operation of services. It provides:

- ✓ Monitoring status and performance of the components involved in providing the service and the reception alarms from them.
- ✓ Events are presented and processed in a single console.
- ✓ Monitoring availability, basic inventory/configuration information and managed elements status is provided via SNMP.

As a result, Zenoss is implemented for iCity platform prototype.

3. iCity Operation System Prototype

The operation System is one of the Platform subsystem, which main objective is facilitate the deployment, operation and maintenance of the iCity platform's network resources, urban data, applications and services, providing a global coverage and a more efficient End-To-End management.

This subsystem is based on a modular design which facilitates the individual evolution of each block minimally affecting to the rest of the components, according to the evolution needs and permitting the Platform's sustained future growth based in the addition of Software and Hardware provisions.

It has a robust, reliable, flexible, open, stable and highly scalable design architecture which allows the incorporation of new services and applications and the rising of information volume, devices, data sources, processes, etc., and manages them in an efficient way.

This subsystem will permit the supervision of all the Platform's elements, hardware and software, operating status and other elements belonging to the service, guaranteeing that its operation accomplish the established Service Level Agreement.

It will provide coverage to the operation system different functional modules supporting, at the same time, the different proceeds of incidents, problems, configuration, changes, versions, capacity and availability.

So, the Management System can be divided into the different submodules:

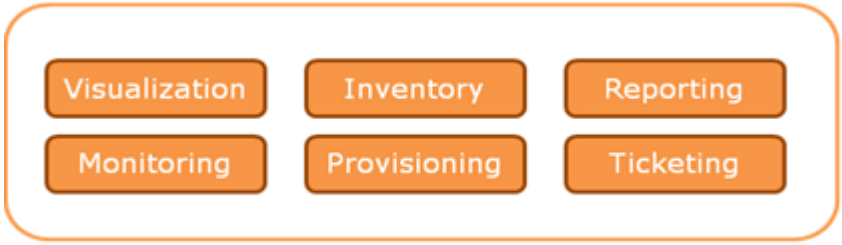


Figure 1: Operation System Prototype

- **Visualization** allows a graphical interface for the system's alerts visualization.
- **Monitoring** allows capture and collection of the data.
- **Inventory** allows a catalogue of elements.
- **Provisioning** defines processes and services associated to each type of element stored in the inventory.
- **Reporting** allows the reports and graphics creation, with the different monitored parameters.
- **Ticketing** provides coverage to the incidents management.

There are two proceeds which allow on the one hand, the sensors' data obtaining as of the SOS service (Configuration Process) and on the other hand, the alerts' obtaining as of the SAS service (Recollection Process).

Configuration Process

The main objective of this process is to save all the information of the Platform elements to enrich the alarms, which are sent to the system. The actualization will be made with two processes:

- GetCapabilities adquires the devices list.
- DescribeSensor obtains the individual information of each device.

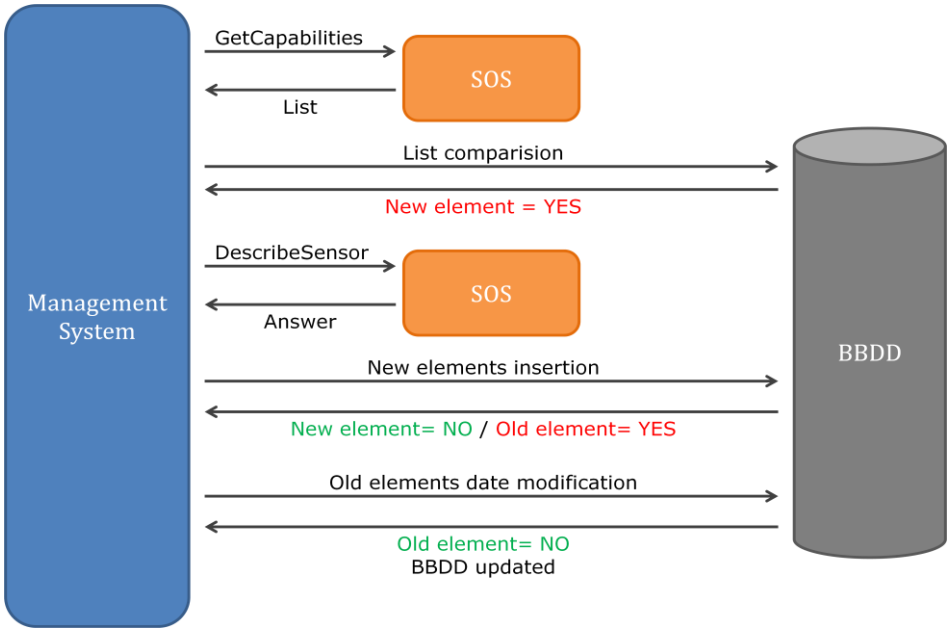


Figure 2: Configutation process

The process will check if the information of each element exists in the Data Base and if they are unsubscribed. Once is checked, it will perform the following operations:

- If the element is in the Data Base and its unregister date is null, it won't perform any operation, because the element will be registered in the system.
- If the element isn't in the Data Base, the Management System will make a *DescribeSensor* request to the SOS Webservice to obtain the necessary information and introduce it in the Data Base.
- If the element is in the Data Base but its unregister date is different from null, the Management System will make a *DescribeSensor* request to the SOS Webservice to obtain the needed information and introduce it in the Data Base. Its unregister date will be updated.

Recollection Process

Once the Data Base information has been updated, the Configuration Process calls the Recollection Process to perform the necessary operation: addAlarmObserver or cancelAlarmOberver.

In the addAlarmObserver process, the collector makes a request to the SAS Webservice with the parameters of the elements that we want to observe. The SAS will return the XMPP channel identifiers where will be the system produced alarms, to enable the subscription to each channel.

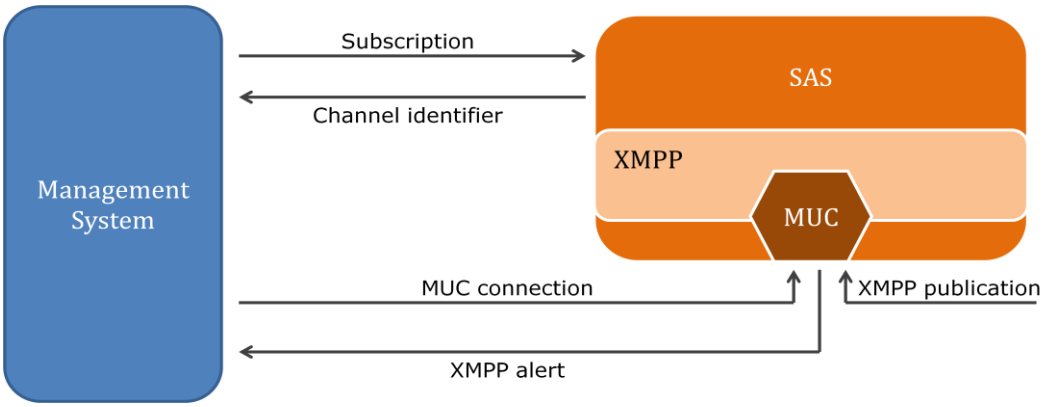


Figure 3: Recollection process

Periodically (configurable time), the process will check the Data Base modifications and will make the necessary management to update the alarm channels of each sensor which its subscribed, doing the following checks:

- If the register date is higher than the last process execution date, the system will make the register of the sensor alarm channel with a *Subscribe* request to the SAS.
- If the unregister date is higher than the last process execution date, the system will make the cancellation of the sensor alarm channel subscription with a *CancelSubscription* request to the SAS.
- In the rest of the cases, the subscription will be renewed with a *RenewSubscription* to the SAS.

When a reception alarm is detected in a subscription channel, this component process, enrich, correlate and send it to the visualization layer.

3.1 Visualization

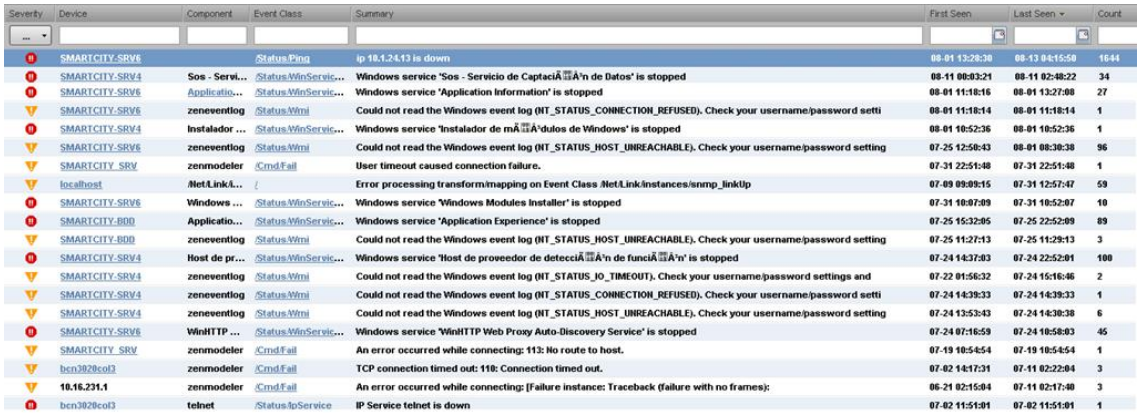
The objective of this module is to visualize the different Platform's elements events: the events detected as a consequence of the monitoring, the events generated by the network devices (SNMP traps) and the events of the Management System.

The visualization/operation module will allow the proactive supervision of the state of all the Platform's devices and components, network interfaces managed, services/processes and Hardware equipment (memory, CPU…). It will be available to show the different devices throughput in a graphical way.

This submodule will have the capacity of generate events considering the established threshold and against an anomalous behavior, which will be a result of the information analysis during a period of time and its variation regarding a pattern or a previous condition.

In that way, the operator will be able to check the state of the elements in real time, permitting a faster incident detection and resolution of problems in case the elements are affected by some type of failure.

In the picture bellow we can see, as an example, the events console of the monitoring tools with a possible information structure for each event:



Severity	Device	Component	Event Class	Summary	First Seen	Last Seen	Count
	SMARTCITY-SRV6		/Status/Ping	ip 10.1.24.13 is down	08-01 13:28:38	08-13 04:15:58	1644
	SMARTCITY-SRV4	Sos - Servi...	/Status/WinServic...	Windows service 'Sos - Servicio de CaptaciÃ³n de Datos' is stopped	08-11 00:03:21	08-11 02:48:22	34
	SMARTCITY-SRV6	Applicatio...	/Status/WinServic...	Windows service 'Application Information' is stopped	08-01 11:18:16	08-01 13:27:08	27
	SMARTCITY-SRV6	zeneventlog	/Status/Wmi	Could not read the Windows event log (NT_STATUS_CONNECTION_REFUSED). Check your username/password setti	08-01 11:18:14	08-01 11:18:14	1
	SMARTCITY-SRV4	Instalador ...	/Status/WinServic...	Windows service 'Instalador de mÃ³dulos de Windows' is stopped	08-01 10:52:36	08-01 10:52:36	1
	SMARTCITY-SRV6	zeneventlog	/Status/Wmi	Could not read the Windows event log (NT_STATUS_HOST_UNREACHABLE). Check your username/password setting	07-25 12:50:43	08-01 08:30:38	96
	SMARTCITY_SRV	zenmodeler	/Cmd/Fail	User timeout caused connection failure.	07-31 22:51:48	07-31 22:51:48	1
	localhost	/Net/Link/i...	/	Error processing transform/mapping on Event Class /Net/Link/instances/snmp_linkUp	07-09 09:09:15	07-31 12:57:47	59
	SMARTCITY-SRV6	Windows ...	/Status/WinServic...	Windows service 'Windows Modules Installer' is stopped	07-31 10:07:09	07-31 10:52:07	10
	SMARTCITY-BDD	Applicatio...	/Status/WinServic...	Windows service 'Application Experience' is stopped	07-25 15:32:05	07-25 22:52:09	89
	SMARTCITY-BDD	zeneventlog	/Status/Wmi	Could not read the Windows event log (NT_STATUS_HOST_UNREACHABLE). Check your username/password setting	07-25 11:27:13	07-25 11:29:13	3
	SMARTCITY-SRV4	Host de pr...	/Status/WinServic...	Windows service 'Host de proveedor de detecciÃ³n de funciÃ³n' is stopped	07-24 14:37:03	07-24 22:52:01	100
	SMARTCITY-SRV4	zeneventlog	/Status/Wmi	Could not read the Windows event log (NT_STATUS_IO_TIMEOUT). Check your username/password settings and	07-22 01:56:32	07-24 15:16:46	2
	SMARTCITY-SRV4	zeneventlog	/Status/Wmi	Could not read the Windows event log (NT_STATUS_CONNECTION_REFUSED). Check your username/password setti	07-24 14:39:33	07-24 14:39:33	1
	SMARTCITY-SRV4	zeneventlog	/Status/Wmi	Could not read the Windows event log (NT_STATUS_HOST_UNREACHABLE). Check your username/password setting	07-24 13:53:43	07-24 14:30:38	6
	SMARTCITY-SRV6	WinHTTP ...	/Status/WinServic...	Windows service 'WinHTTP Web Proxy Auto-Discovery Service' is stopped	07-24 07:16:59	07-24 10:58:03	45
	SMARTCITY_SRV	zenmodeler	/Cmd/Fail	An error occurred while connecting: 113: No route to host.	07-19 10:54:54	07-19 10:54:54	1
	bcn3020col3	zenmodeler	/Cmd/Fail	TCP connection timed out: 110: Connection timed out.	07-02 14:17:31	07-11 02:22:04	3
	10.16.231.1	zenmodeler	/Cmd/Fail	An error occurred while connecting: [Failure instance: Traceback (failure with no frames):	06-21 02:15:04	07-11 02:17:40	3
	bcn3020col3	telnet	/Status/IpService	IP Service telnet is down	07-02 11:51:01	07-02 11:51:01	1

Figure 4: Events console

- Severity: critic level of the event.
- Device: one higher hierarchical level's device above the element which produced the alarm.
- Component: element which produced the alarm.
- Event class: indicator which produced the alarm.

- Summary: alarm description (completely configurable).
- First seen: first time that the alarm was detected.
- Last seen: last time that the alarm was detected.
- Count: number of times that the system has registered the alarm. When an alarm arrives with the same fields as another one that has been previously detected, the counter increases, to avoid the alarms deduplication.

From the tool interface, the operator of iCity platform will be able to see the complete information that the alarm system has got and the different elements of the Platform, for example, their location or IP direction.

Figure 5: General information related an alarm event

The received elements will be stored in a Data Base, which will have got, at least, the following variables:

- Active events: it will contain a list of all the events associated to the active Platform elements, updating in real time. In that way, the operator will be able to identify them in a fast and visual way.
- Historical: it will contain solved past elements. The operator will be able to establish the rules that he consider to pass to the historical active events automatically.

- Details: it will contain the events defined by the operator. The administrator will be able to create new alarms, remove existent alarms and export showed alarms to different formats.

This module will allow to definite automatic actions that will execute when the system receive certain elements and the SNMP, Telnet, SSH or WMI access credentials to the equipments in their different versions.

From this interface, the administrator will be able to modify and remove existent users and create new users, allocating different roles which will permit or limit the access and execution of determined options.

3.2 Monitoring

The monitoring module will facilitate the Platform elements monitoring and supervision (collectors, sensors…), network infrastructure (switches, routers…), collect and store data services, CRM, portal, etc.

This module will contain the following submodules:

- Data capture and collection.
- Monitoring.

3.2.1 Data capture and collection

This submodule will provide the access to the configuration parameters of the capture functionalities. To speed up the capture configuration of homogeneous devices, the system will allow the use of templates, which will group all the device typology relevant parameters. A device can fit with multiple typologies simultaneously, so it can hold to multiple templates.

It will allow copying one of the templates to modify it after, and will give support to the deployment of the new template to all the existent and linked devices. The operators will be able to import and export to XML format all the available templates.

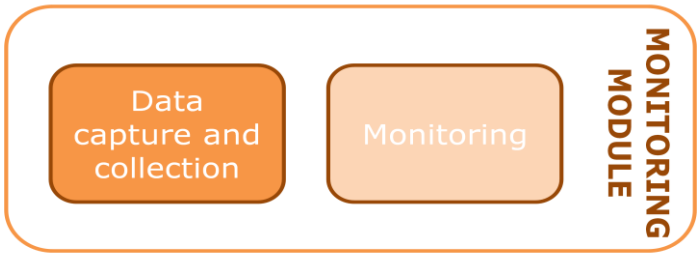


Figure 6: Monitoring module – Data capture and collection

It will have available the following capture functionalities:

- SNMP: capture the devices information with SNMP protocol (SNMP get), SNMP traps and its associated management.
- Poll: specific agents' interrogation in a centralized way.
- IPMI: interrogation of the agents which use the IPMI (Intelligent Platform Management Interface) Protocol.
- Remote services: monitoring of network services published without agent (ping, FTP, HTTP, etc.).
- Web: realization of web pages sequences following the POST and GET events, associated to guarantee the complete availability of themselves, the response times and the expected results.
- Passive: capture of the directories files that the server verifies periodically.
- Log: specific expressions and text sequences (in log files) capture with regular expressions.
- Proxy: system data capture with a group of intermediate servers which gather data from a subgroup of devices and communicate them to the main server.

This submodule will recognize the devices situated in the supervised infrastructure, will link them automatically to their respective device typology and will start the monitoring.

3.2.2 Monitoring

This module will trigger alarms when the conditions based on the value of the equipment parameters get accomplished, triggering alarms to the operators when the alarms reach determined levels of criticality, offering the possibility of envy with multiple messaging mechanisms (e-mails, SMS, etc.).

It will permit the configuration of the trigger conditions to adapt it to the operator needs and will have an expression definition and value calculating language to define complex alarms.

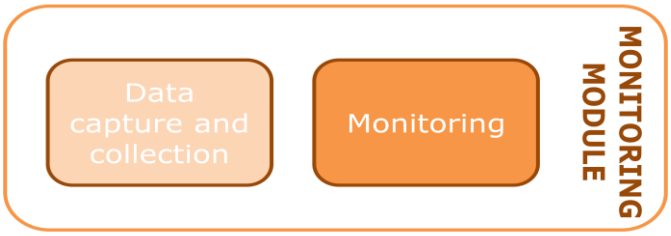


Figure 7: Monitoring module – Monitoring

This submodule will allow visualizing and managing the hierarchy of the alarms associated to different devices and centres and indicators and the gravity or criticality of the alarms depending on multiple parameters (number of similar alarms, alarm duration, number or repetitions of the same alarm in a determined period of time, etc.).

The operator will be able to do the pertinent configurations to make the system take actions automatically against determined alert situations, with local or remote scripts.

This submodule will be the responsible for inform of the disponibility, monitored infrastructure failure and its impact on the client services, to after generate SLA's (Service Level Agreement) statistics.

3.3 Inventory

This module will permit the inventory of the different logical and physical elements (sensors, collectors, transport network infrastructure, etc.) and the data of each Platform equipment.

In addition, this module will support: service provisioning, operating and management and physical elements life cycle.

Conceptually and, with independency of the network, is proposed the following modeling structure (it can be modified): each device and connection is treated as a typology instance (N1), specified for a family (N2) and a component (N3). In exchange, for the services it's needed just one modeling (N2).

The inventory module will include the following submodules:

- Management consultation.
- Documentation management.
- Network and services management.
- Catalogue management.
- Information quality.

3.3.1 Documentation management

The aim of this submodule is to centralize data inventory of equipment and services, avoiding maintenance of inventory information across different systems, therefore there is only one maintained inventory for the iCity platform

This module manages the differente states of the entities in the documentation module (as planned, build, in service, etc.). Also, it provides historical documents in order to search information in an easy way. Historicals save any changes to the configuration and status of equipment, network or services of iCity platform.

Access to external users will be restricted, so that they shall be allowed access to the devices on which the user is responsible for maintenance, so they can document their own networks but they do not have access to other elements of the platform.

3.3.2 Network and service management

This submodule have sufficient management capacity to create, configure and maintain the network and tolos that increase efficiency and efectiveness in order to reduce the time fro provisioning tasks.

Network and service management allows the representation and creation of networks through different types of elements (nodes, connections, networks, etc.) Also, it provide a functional view of all resources, thereby facilitating the automatic design of circuits and networks.

This submodule collects and stores information about changes in the network resource in order to allow visualization fo multiple forms of network planning. In this way, operator can anticipate future capacity needs by extrapolation algorithms relying on the historical.

3.3.3 Catalogue management

This module will provide the submodule ability to evolve and adapt through a master set and customize the types of networks and services in order to assign mandatory attributes, optional attributes, constant values and ranges.

Catalogue management allows management of the structure of services, equipment, connections, routes and cards, that it offers functionalities for managing entieties (cards, equioment, connections, groups, routes, and services) and to establish relationships between these groups and the inventory.

This module is aling with TMN standard (Telecommunications Management Network) and with SID standard (Shared Information Data Model).

4. Conclusions

This first year, a preliminary system has been included in the prototype in order to allow, in a short time, the deployment of pilots and also the development of apps, with a operation system that would monitoring real time information.

Operation system offers the following benefits:

- Openness allows the integration of the whole system is based on universal platform.
- Adaptability provides confront the swelling and much more complex services.
- Expansibility allows the adaptation to future complex situation.
- Flexible provides an easy for correcting, maintenance, update and upgrade.
- Friendly user interface.

Future evolutions of the prototype (D4.7 & D4.12) will follow inputs coming from WP3, WP5 and WP7.