EGSE Health Monitoring as Plug-in for SCOS-2000

Gian Piero Izzo[1], Vincenzo Martucci[2] , Giulio Troso[3] , Enrico Angioli[4]

[1]Vitrociset spa
Via Tiburtina 1020, Rome (Italy)
Email:gianpieroizzo@alia-space.com

[2]Vitrociset spa
Via Tiburtina 1020, Rome (Italy)
Email:v.martucci@vitrociset.it

[3]Vitrociset spa
Via Tiburtina 1020, Rome (Italy)
Email:g.troso@vitrociset.it

[4]Vitrociset spa
Via Tiburtina 1020, Rome (Italy)
Email:e.angioli@vitrociset.it

INTRODUCTION

A typical EGSE is based on a distributed system architecture that can reach a high degree of complexity. In this context, it becomes of crucial importance for the operator to have at any moment a clear view of the health status of the EGSE used resources and equipment connected.

The basic idea is to endow the EGSE with an integrated monitoring system capable to collect key information from the subsystems, process them and make available to the operator as a valuable support in the definition and validation of the test environment setup and in the execution of test campaign with the Unit Under Test (UUT). The monitoring system shall also provide a benchmark to evaluate and optimize the use of EGSE resource, with a consequent performance improvement.

This paper describes a quite innovative approach adopted to develop such monitoring system as a plug-in for SCOS-2000. This solution is hereafter called the Health Monitoring System (HM).

HEALTH MONITORING

The Health Monitoring is the component in charge of handling and reporting the system information during the test execution phase:

- Monitoring Operating System data in EGSE subsystems
- Broadcasting notifications from monitored machines.

The Health Monitoring (HM) subsystem has been conceived as Central Checkout System (CCS) component and logically structured in a Client-Server application, potentially extensible in a topological graph on the entire EGSE system.

Component Role

The role of HM component is to monitor EGSE subsystems and to grab snapshots of the status of the entire system.

The HM module is a “system watcher” in charge of handling and reporting at least the following information:

- Monitoring OS resources in EGSE subsystems, like: disk usage, memory and CPU load

- Status of processes running in the subsystem;
- Monitoring of point-to-point connections between different hosts
- Status of the time synchronization process

The driving factor is to model all the monitored entities as TM parameters defined in the SCOS-2000 MIB, contained in ad-hoc defined TM packets.

This will allow the HM component to show all the monitor information over an SCOS-2000 display.

Core Architecture

The definition of the core architecture of the HM subsystem takes into account the following approach:

- Enables EGSE System to be managed without a heavy resource investment.
- Provides a scalable management architecture.
- Integrates existing management solutions.

Due to the exposed specifications, the HM architecture takes advantage from the JMX Mbean Technology in Java Programming Language.

JMX technology provides the tools for building distributed, modular and dynamic solutions for managing and monitoring devices, applications, and service-driven networks. By design, this standard is suitable for adapting legacy systems, implementing new management and monitoring solutions, and plugging into those of the future.

Compliant to the JMX Architecture the HM module can be divided in three levels (see Fig. 1):

- Instrumentation Level
- Agent Level
- Manager Level

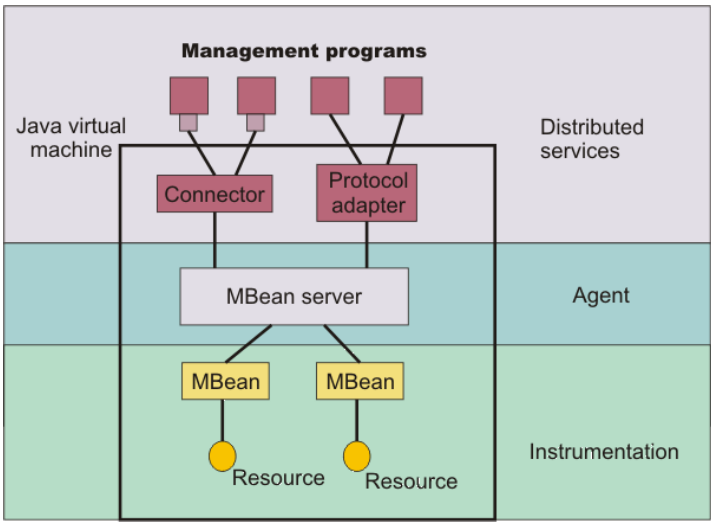


Fig. 1. A JMX Approach to Monitoring

The <u>Instrumentation Level</u> provides the software components (Mbean Plugins) to manage many different kinds of resources. A managed resource can be an application, a service, a device or a low level component in the operating system.

The instrumentation of a resource allows it to be manageable through the agent level described as follow.
With a plug-in development approach, instrumentation software component are designed to be flexible, simple, and easy to implement.

In addition, the instrumentation level also specifies a notification mechanism. This allows resource component to generate and propagate notification events to components of the other levels.

In each point it is possible to build dedicated notification handler to improve performance and reduce data flow over the network.

The Agent Level provides the software component (Mbean Server) for implementing agents. Management agents directly control the resources and make them available to remote monitor. Agents are located on the same machine as the resources they control.

This level builds upon and makes use of the instrumentation level to define a standardized agent to manage manageable resources. The agent consists of a server and a set of services for handling resources. In addition, an agent will need at least one communications adaptor or connector. The server implementation and the agent services are mandatory in the overall architecture.

The agent can be embedded in the machine that hosts the manageable resources when a Java Virtual Machine (JVM) is available in that machine. Likewise, the agent can be instantiated into a mediation/concentrator element when the managed resource only offers a proprietary (non-Java) environment
Agents do not require knowledge of the remote management applications that use them.

The Manager Level provides the interfaces for implementing managers. This level defines management interfaces and components that can operate on agents or hierarchies of agents. These components can:

- Provide an interface for management applications to interact transparently with an agent and its manageable resources through a connector
- Expose a management view of the agents
- Collect and distribute notification broadcasting from agent and instrumentation level.

The following Fig. 2 gives an overview of the whole HM main architecture.

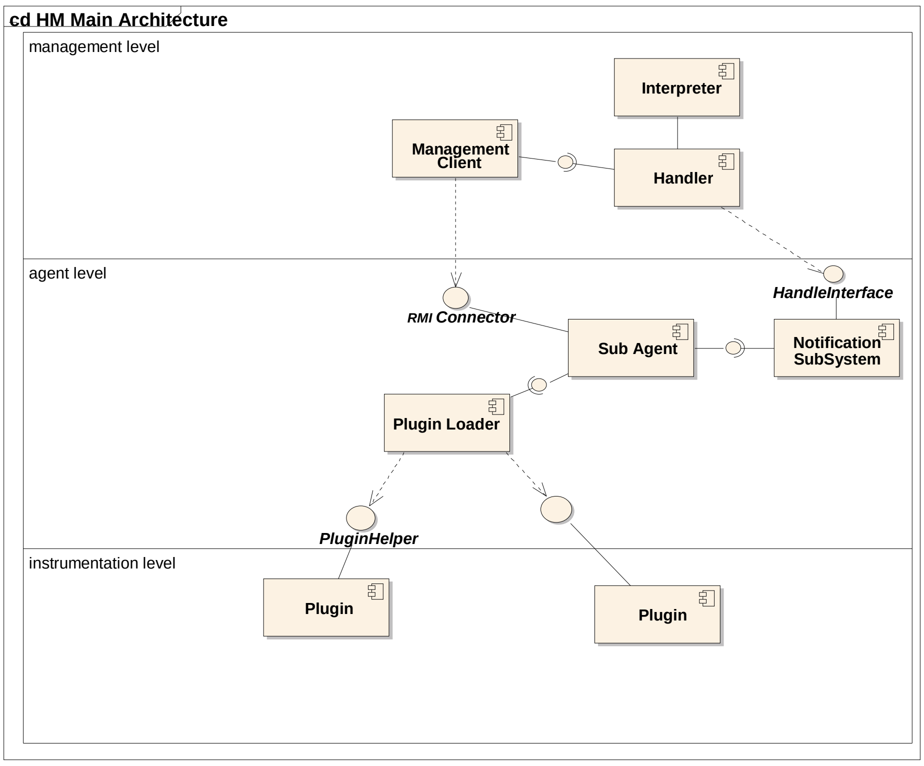
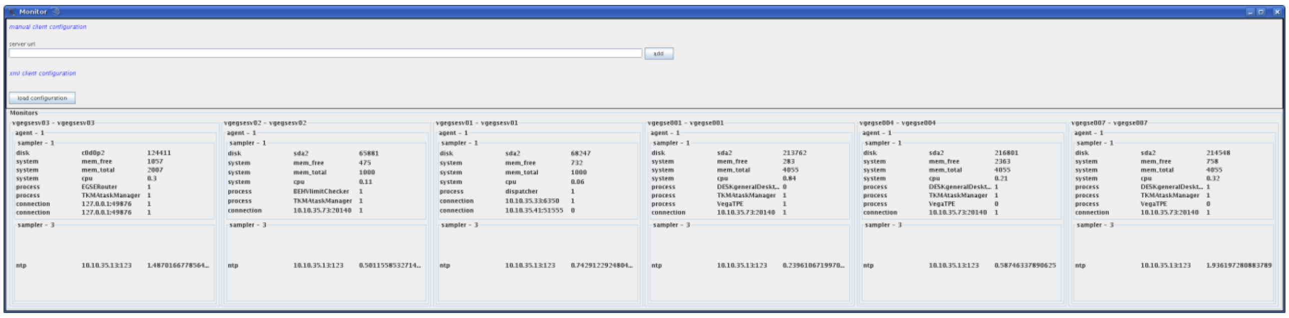


Fig. 2. HM Main Architecture

Client application called HM Agent (including JMX agent and instrumentation levels) can be deployed as services on each EGSE's subsystem and is aimed to collect monitoring data to be sent to the to the Server application called HM Manager.

HM Manager application has to be installed on one EGSE machine acting as a central collector of all the information coming from the distributed clients.
The HM Manager normally is deployed as a server application launched within the standard SCOS-2000 startup process.
It can be launched optionally showing a GUI which allows the operator to see the real flow of information coming from Agents and directed to SCOS-2000 as shown in the picture below:



```
vgegsesv03 - vgegsesv03
agent - 1
sampler - 1
disk         c0d0p2              124411
system       mem_free            1057
system       mem_total           2007
system       cpu                 0.3
process      EGSERouter          1
process      TKMAtaskManager     1
connection   127.0.0.1:49876     1
connection   127.0.0.1:49876     1
sampler - 3
ntp          10.10.35.13:123     1.4870166778564...

vgegsesv02 - vgegsesv02
agent - 1
sampler - 1
disk         sda2                65881
system       mem_free            475
system       mem_total           1000
system       cpu                 0.11
process      BEHVlimitChecker    1
process      TKMAtaskManager     1
connection   10.10.35.73:20140   1
sampler - 3
ntp          10.10.35.13:123     0.5011558532714...

vgegsesv01 - vgegsesv01
agent - 1
sampler - 1
disk         sda2                68247
system       mem_free            732
system       mem_total           1000
system       cpu                 0.06
process      dispatcher          1
connection   10.10.35.33:6350    1
connection   10.10.35.41:51555   0
sampler - 3
ntp          10.10.35.13:123     0.7429122924804...

vgegse001 - vgegse001
agent - 1
sampler - 1
disk         sda2                213762
system       mem_free            283
system       mem_total           4055
system       cpu                 0.84
process      DESKgeneralDeskt... 0
process      TKMAtaskManager     1
process      VegaTPE             1
connection   10.10.35.73:20140   1
sampler - 3
ntp          10.10.35.13:123     0.2396106719970...

vgegse004 - vgegse004
agent - 1
sampler - 1
disk         sda2                216801
system       mem_free            2363
system       mem_total           4055
system       cpu                 0.21
process      DESKgeneralDeskt... 1
process      TKMAtaskManager     1
process      VegaTPE             0
connection   10.10.35.73:20140   1
sampler - 3
ntp          10.10.35.13:123     0.58746337890625

vgegse007 - vgegse007
agent - 1
sampler - 1
disk         sda2                214548
system       mem_free            758
system       mem_total           4055
system       cpu                 0.32
process      DESKgeneralDeskt... 1
process      TKMAtaskManager     1
process      VegaTPE             0
connection   10.10.35.73:20140   1
sampler - 3
ntp          10.10.35.13:123     1.936197280883789
```

Fig. 3. HM Manager GUI

Interfaces with SCOS-2000

The HM module has been designed to be scalable over the EGSE subsystems. The manager level and the instrumentation one have been integrated in the main system with extensible interfaces to provide loosely coupled software components.

One of the objectives of the HM component is to show all the monitor information over an SCOS-2000 display. To provide this functionality the component uses some interfaces based on CORBA protocol to manage the injection of TM parameters inside SCOS-2000.

HM manager takes care to insert all retrieved information into a configurable set of telemetry packets and to inject them regularly inside the SCOS-2000 Telemetry Flow for processing, archiving and displaying.
The HM manager is currently injecting all this parameters in the SCOS-2000 Telemetry Chain using the provided EXIF interface as shown in Fig. 4 (a porting to the new SMF module is foreseen).

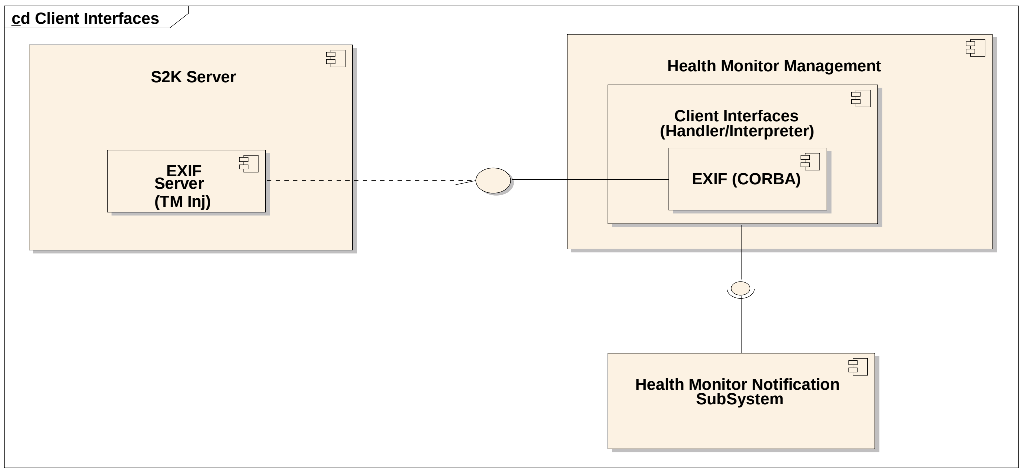
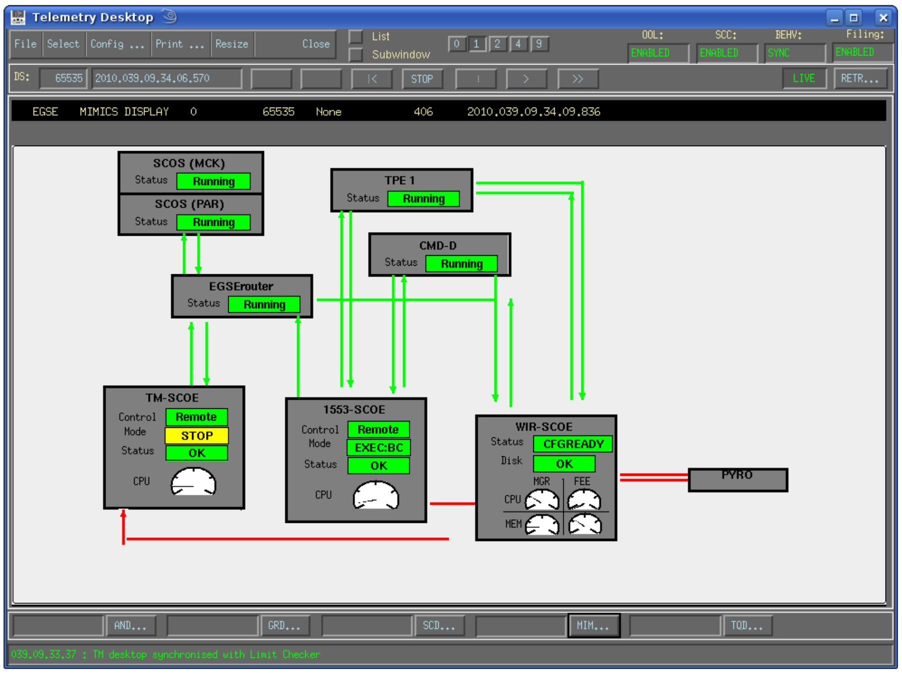

cd Client Interfaces
S2K Server
EXIF Server (TM Inj)
Health Monitor Management
Client Interfaces (Handler/Interpreter)
EXIF (CORBA)
Health Monitor Notification SubSystem

Fig. 4. SCOS-2000 Interface

The mapping of the EGSE status to TM parameters allows their visibility at display and procedure level. Such a feature guarantees the possibility to develop dedicated watchdogs on system variables and react in case of anomalies.

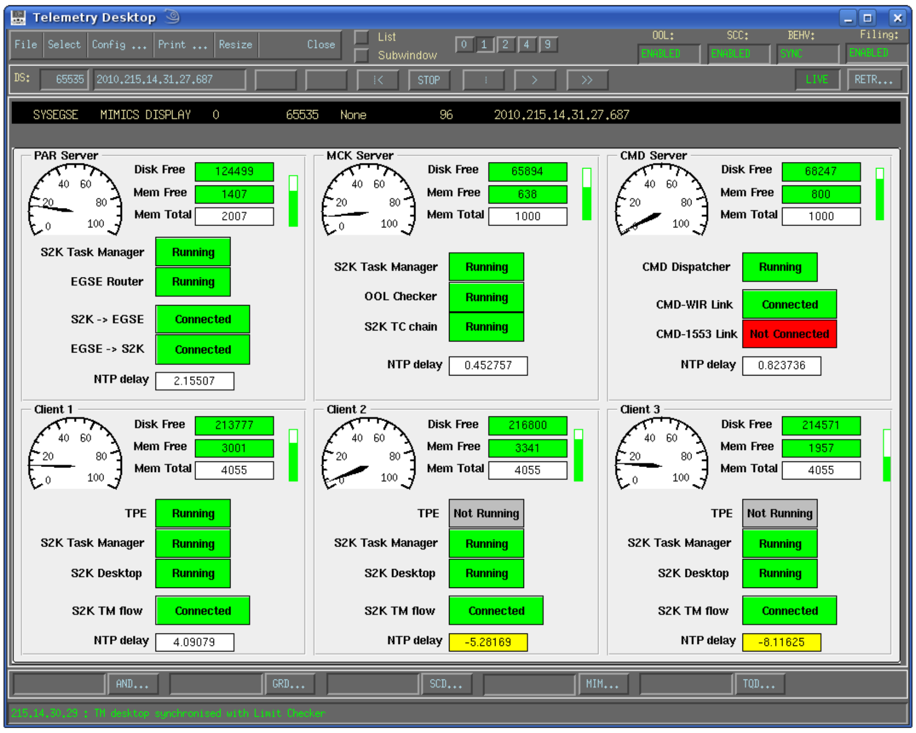
The following picture shows a SCOS-2000 MIMIC Display showing an overview of the VEGA EGSE status using a set of TM Parameters coming from the HM module:



Telemetry Desktop
File
Select
Config ...
Print ...
Resize
Close
List
Subwindow
OOL: ENABLED
SCC: ENABLED
BEHV: SYNC
Filing: ENABLED
DS: 65535
2010.039.09.34.06.570
STOP
LIVE
RETR...
EGSE MIMICS DISPLAY 0 65535 None 406 2010.039.09.34.09.836
SCOS (MCK) Status Running
SCOS (PAR) Status Running
TPE 1 Status Running
CMD-D Status Running
EGSErouter Status Running
TM-SCOE Control Remote Mode STOP Status OK CPU
1553-SCOE Control Remote Mode EXEC:BC Status OK CPU
WIR-SCOE Status CFGREADY Disk OK MGR FEE CPU MEM
PYRO
AND...
GRD...
SCD...
MIM...
TOD...
039.09.33.37 : TM desktop synchronised with Limit Checker

Fig. 5. Example of SCOS-2000 Mimic Display showing HM parameters

This picture below shoes another example of MIMIC Display offering more details on specific parts of the VEGA EGSE and their associated HM TM parameters:



SYSEGSE MIMICS DISPLAY 0 65535 None 96 2010.215.14.31.27.687

PAR Server		MCK Server		CMD Server	
Disk Free	124439	Disk Free	65894	Disk Free	68247
Mem Free	1407	Mem Free	638	Mem Free	800
Mem Total	2007	Mem Total	1000	Mem Total	1000
S2K Task Manager	Running	S2K Task Manager	Running	CMD Dispatcher	Running
EGSE Router	Running	OOL Checker	Running	CMD-WIR Link	Connected
S2K -> EGSE	Connected	S2K TC chain	Running	CMD-1553 Link	Not Connected
EGSE -> S2K	Connected				
NTP delay	2.15507	NTP delay	0.452757	NTP delay	0.823736

Client 1		Client 2		Client 3	
Disk Free	213777	Disk Free	216800	Disk Free	214571
Mem Free	3001	Mem Free	3341	Mem Free	1957
Mem Total	4055	Mem Total	4055	Mem Total	4055
TPE	Running	TPE	Not Running	TPE	Not Running
S2K Task Manager	Running	S2K Task Manager	Running	S2K Task Manager	Running
S2K Desktop	Running	S2K Desktop	Running	S2K Desktop	Running
S2K TM flow	Connected	S2K TM flow	Connected	S2K TM flow	Connected
NTP delay	4.09079	NTP delay	-5.28169	NTP delay	-8.11625

215.14.30.29 : TM desktop synchronised with Limit Checker

Fig. 6. Example of SCOS-2000 Mimic Display showing HM parameters

Component Configuration

An XML based Configuration Management module for HM has been designed to provide easy configuration for all subsystems needed to be monitored. Each agent mounts the related plugins at start-up from this xml file. Such operation will be automatically performed without user action.

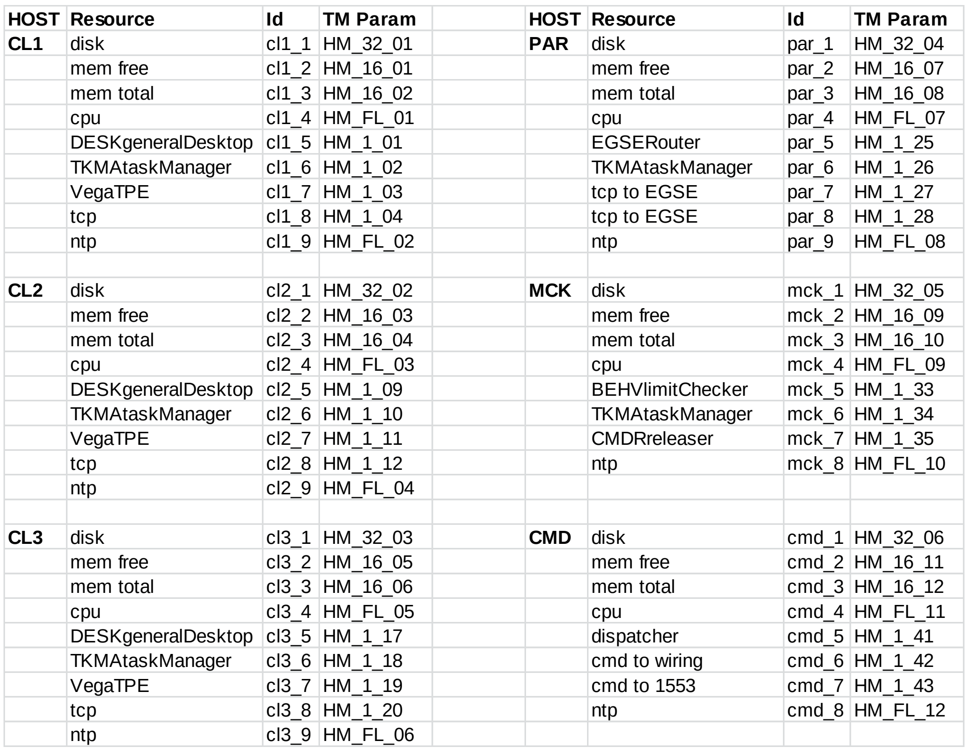
The following Fig. 7 shows an example of XML configuration file for the HM Agent installed on a VEGA EGSE client. The structure is organized in containers, each one containing a set of managed objects defined by their ClassType and Name attributes.

```
<system Id="1" Name="linux" Host_ip="10.10.11.94" Host_port="9999">
        <agent  Id="1" Name="agent" Host_domain="linux">
                <SE Type="mocontainer" ClassType="Sampler" Id="1" Name="resourcesampler" Attr=”4000”>
                        <SE Type="managedobject" ClassType="disk" Id="1553_1" Name="sdb6" >
                        </SE>
                        <SE Type="managedobject" ClassType="system" Id="1553_2" Name="mem_free" >
                        </SE>
                        <SE Type="managedobject" ClassType="system" Id="1553_4" Name="cpu" >
                        </SE>
                        <SE Type="managedobject" ClassType="process" Id="1553_5" Name="Spawn.jar" >
                        </SE>
                </SE>
                <SE Type="mocontainer" ClassType="TCPMonitor" Id="2" Name="tcpmonitor" Attr="4000">
                        <SE Type="managedobject" ClassType="connection" Id="1553_6" Name="10.10.11.2:7777" >
                        </SE>
                </SE>
                <SE Type="mocontainer" ClassType="Sampler" Id="3" Name="ntpsampler">
                        <SE Type="managedobject" ClassType="ntp" Id="1553_7" Name="10.10.35.231:123" >
                        </SE>
                </SE>
        </agent>
</system>
```

Fig. 7. HM Agent XML Configuration File

On top of the agent configuration, an XML file shall be provided to the HM manager which will contain the association between monitored objects (through their Id attribute) and SCOS-2000 TM Parameter names (as defined in MIB).

The following Fig. 8 shows an example of configuration coming from the HM Manager installed on the VEGA EGSE server.



HOST	Resource	Id	TM Param		HOST	Resource	Id	TM Param
CL1	disk	cl1_1	HM_32_01		**PAR**	disk	par_1	HM_32_04
	mem free	cl1_2	HM_16_01			mem free	par_2	HM_16_07
	mem total	cl1_3	HM_16_02			mem total	par_3	HM_16_08
	cpu	cl1_4	HM_FL_01			cpu	par_4	HM_FL_07
	DESKgeneralDesktop	cl1_5	HM_1_01			EGSERouter	par_5	HM_1_25
	TKMAtaskManager	cl1_6	HM_1_02			TKMAtaskManager	par_6	HM_1_26
	VegaTPE	cl1_7	HM_1_03			tcp to EGSE	par_7	HM_1_27
	tcp	cl1_8	HM_1_04			tcp to EGSE	par_8	HM_1_28
	ntp	cl1_9	HM_FL_02			ntp	par_9	HM_FL_08
CL2	disk	cl2_1	HM_32_02		**MCK**	disk	mck_1	HM_32_05
	mem free	cl2_2	HM_16_03			mem free	mck_2	HM_16_09
	mem total	cl2_3	HM_16_04			mem total	mck_3	HM_16_10
	cpu	cl2_4	HM_FL_03			cpu	mck_4	HM_FL_09
	DESKgeneralDesktop	cl2_5	HM_1_09			BEHVlimitChecker	mck_5	HM_1_33
	TKMAtaskManager	cl2_6	HM_1_10			TKMAtaskManager	mck_6	HM_1_34
	VegaTPE	cl2_7	HM_1_11			CMDRreleaser	mck_7	HM_1_35
	tcp	cl2_8	HM_1_12			ntp	mck_8	HM_FL_10
	ntp	cl2_9	HM_FL_04					
CL3	disk	cl3_1	HM_32_03		**CMD**	disk	cmd_1	HM_32_06
	mem free	cl3_2	HM_16_05			mem free	cmd_2	HM_16_11
	mem total	cl3_3	HM_16_06			mem total	cmd_3	HM_16_12
	cpu	cl3_4	HM_FL_05			cpu	cmd_4	HM_FL_11
	DESKgeneralDesktop	cl3_5	HM_1_17			dispatcher	cmd_5	HM_1_41
	TKMAtaskManager	cl3_6	HM_1_18			cmd to wiring	cmd_6	HM_1_42
	VegaTPE	cl3_7	HM_1_19			cmd to 1553	cmd_7	HM_1_43
	tcp	cl3_8	HM_1_20			ntp	cmd_8	HM_FL_12
	ntp	cl3_9	HM_FL_06					

Fig. 8. HM Manager Configuration for VEGA EGSE

CONCLUSIONS

At the time of writing, the HM is successfully used and implemented on the EGSE system for VEGA Launcher, providing the health status (cpu usage, disk usage, major connections) of three servers and three clients, plus the connection status and the content of status words coming from various SCOEs.

Thanks to its scalable and configurable architecture HM module can be easily adopted on every SCOS-2000 based Mission Control System (MCS) and EGSE installation without any specific software customization and with a minimal deployment activity covering the XML file configuration and the correspondent SCOS-2000 MIB population of HM parameters.

Moreover the plug-in oriented architecture allows a simple way to define new types of managed resources (e.g. specific scripts defined by the user) which can be immediately added and used by the HM Agents.