Experiences with the SNMP-based integrated management of a CORBA-based electronic commerce application

J.I. Asensio[†]*, V.A. Villagrá, J.E. López de Vergara, J. Berrocal*
Dep. Telematic Systems Engineering
Technical University of Madrid (DIT-UPM)
ETSI Telecomunicación
Ciudad Universitaria s/n, 28040 Madrid
SPAIN
{jasensio,villagra,jlopez,berrocal}@dit.upm.es

[†]*Visiting researcher from University of Valladolid, Spain*

Abstract

This paper describes the design and implementation of an SNMP-based management system for the CORBA-based electronic brokerage application that has been implemented within the scope of the ABS (Architecture for Information Brokerage Service) ACTS[*] Project. The management system is based on the use of an SNMP-CORBA gateway that uses the JIDM (Joint Inter-Domain Management) Specification Translation algorithms and incorporates ideas from the services and facilities contained in the SNMP part of the JIDM Interaction Translation documents. The paper also focuses on the way the management information has been instrumented within the CORBA application trying to make it transparent with respect to the developers of the functional aspects of the managed application.

Keywords

Management of Distributed Applications, Management of Electronic Commerce, Case Studies and Experiences

[*] This work is partially sponsored by the Commission of the European Union under the project ABS, project number AC206 of the ACTS programme. The ABS consortium is integrated by CNET (France), DeTeBerkom (Germany), Sonera (Finland), CET (Portugal), SEMA (France), VTT (Finland), NTUA (Greece), DIT-UPM (Spain), Degriftour (France) and PoP (Germany). The views presented here do not necessarily represent those of the ABS consortium.

1. Introduction

Distributed applications have created important challenges in the management world during the recent years. Integrated, standard, scaleable, and effective management solutions have been requested in order to cope with the great proliferation and increasing importance of this kind of applications. The irruption of object-oriented distributed processing platforms based on standard and, to some extent, mature specifications such us CORBA [1] has given international consortia the opportunity to work on concrete management frameworks for distributed objects applications. Thus, for instance, ISO and ITU-T are currently working on the application of the ODP-RM [2] principles to distributed management in ODMA (Open Distributed Management Architecture) [3], TINA-C (Telecommunications Information Networking Architecture Consortium) has set up a special working group on Service Management taking its Management Architecture [4] as a basis, the NMF (Network Management Forum) has proposed its Management Systems Framework [5], both the NMF and the OpenGroup have been developing their JIDM (Joint Inter-Domain Management) specification [6] (JIDM-ST) and interaction [7,8,9] (JIDM-IT) translations which have also been proposed as solution for the TMN-CORBA interworking efforts of the OMG (Object Management Groups) Telecom Task Force [10], etc.

These efforts have focused on different aspects of the management problem: identification and definition of new management functionality, adaptation of existing management functionality from the network management field, management instrumentation of distributed applications, integration of these new management frameworks with traditional ones, etc. As a result of those efforts, some components of the ideal framework for the management of distributed objects applications have been specified and are currently been refined but there is not a complete solution yet.

This paper describes the particular approach adopted for the design and implementation of an SNMP-based management system for a CORBA-based distributed application: the electronic brokerage application that is being developed within the scope of the ABS (Architecture for Information Brokerage Service) ACTS European project.

The management system is based on an SNMP-CORBA gateway that uses the JIDM-ST algorithms [6] and that incorporates some of the ideas of the JIDM-IT for SNMP/CORBA interworking. The paper will describe the architecture of the management gateway, how it is related to the JIDM-IT, how management information has been instrumented within the CORBA application, how the SNMP managing console has been designed, and how the application performance is affected by the management system.

The paper is structured as follows: section 2 introduces the ABS brokerage service and the prototype that has been developed and that had to be managed, describing the requirements that, from a management point of view, it imposes. Section 3 describes the proposed management architecture focusing on the definition of management information, the design of the SNMP-CORBA gateway

and its relation to the JIDM-IT specification, the structure of the managing system, and the way the management information has been instrumented. Finally, section 4 concludes the article by commenting some results obtained from the tests performed with the developed management system, and presents some future work trends that are intended to be followed.

2. The ABS Brokerage Application and its management requirements

The brokerage service plays a key role in the first stages of commercial transactions by putting Supply and Demand together. By means of this service, customers get information about a set of providers that might fulfil their demands thus helping them in the decision of which of those providers are the most suitable. Simultaneously, the brokerage service helps the providers in putting their offers closer to the customers.

The ABS consortium was devoted to the translation of the brokerage concept to the Electronic Commerce World [11]. As a result, this consortium worked in the definition and specification of an architecture of electronic brokerage systems based on ODP-RM and TINA concepts [12], developing prototypes of this architecture in CORBA and JAVA, and validating both architecture and prototypes in a set of trials in real electronic commerce environments.

Figure 1 summarises the functionality of the ABS Broker Application and its interactions with the users, the information and content providers and other brokers:

- the ABS broker receives requests from users regarding some kind of information or good they want to obtain;
- the ABS broker contains an internal representation of the information or goods a set of registered content providers offer, as well as information about other existing brokers that can help to satisfy users' demands;
- the ABS broker generates a set of searches and retrieves precise information from the content providers and other brokers, information that is combined in order to satisfy the users' demands;
- the ABS broker presents the obtained results to the users. Those results usually consist of a list of suitable content providers (registered in the locally or in other external brokers) that offer information or goods with the characteristics required by the users.

In March 1997, the first ABS brokerage architecture was released and the corresponding prototype was finished in December 1997 using CORBA as the middleware technology (more concretely, OrbixWeb 2.0.1 was employed) and JAVA as the programming language. A second and more complete version of the architecture was finished in March 1998 and the final implementation of the new prototype was completed in September 1998 (using OrbixWeb 3.0). Details about the different components of the ABS broker system architecture can be found in [13].

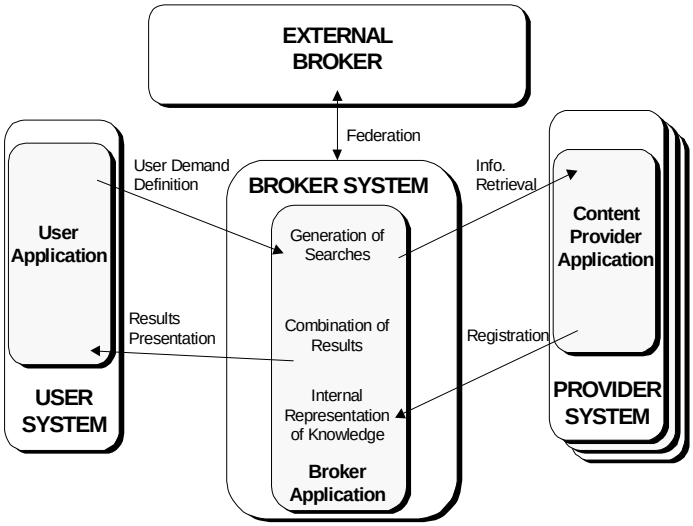

Figure 1: ABS broker functionality and its main interactions with other actors in the brokerage scenario.

During the trials of both prototype versions several aspects of the CORBA-based broker application had to be managed:

- The configuration of the application: individual application components had to be launched and stopped.
- The performance and accounting of the application: different performance and accounting parameters regarding individual application components and the application as a whole had to be retrieved.
- The faults of the application: application components had to be able to generate notifications about unexpected conditions.

Therefore, a management system for a CORBA-based application had to be designed taking into account the above requirements and the following constrains:

- The management system didn't have to be implemented from scratch. Existing components from other management systems had to be reused whenever possible.
- The management of the brokerage application had to be integrated with the management of network and system aspects.
- The development of the different application components didn't have to be affected by the introduction of the support for the management functionality.

The previous items led to several design decisions that were taken during the first stages of the development of the ABS management system:

- As application, network and system management had to be integrated, and, simultaneously, a fast development of prototypes had to be achieved, the SNMP standard [14] was chosen as the basis for the managing system. The

simplicity of the SNMP management framework, the great number of available SNMP management platforms (even in the public domain), and the availability of SNMP network and system management agents were key facts for this choice. The use of the OSI management framework was also considered but its power, and therefore its complexity, was not needed for the characteristics of the ABS management system.

- An SNMP-CORBA gateway approach was adopted for achieving interoperability between the SNMP (the managing part) and CORBA (the managed part) frameworks. The Multiarchitectural Manager [15] was another suitable approximation but it was preferred to make the ABS management system as independent as possible from the services of a particular SNMP management platform. The design of the SNMP-CORBA gateway for the ABS broker management was greatly influenced by the JIDM concepts [6,8,9] mainly because of the generic SNMP SMI to CORBA IDL translations algorithms it provides and because of the way it handles SNMP tables in the CORBA domain.
- The instrumentation of the management information in the broker application was based on the "interceptor" concept included in the last versions of the CORBA standard. This design choice was taken due to two facts. First of all, most of the management information variables that wanted to be obtained from the application components were related to the number of invocations of a particular method of the CORBA interfaces of an application component or related to the time consumed during the completion of those methods. That information could be obtained within the appropriate "interceptors". Secondly, the instrumentation of the management information didn't have to affect the development of the management applications components. The introduction of an "interceptor" is something completely transparent to the developer of a CORBA component.

In the following sections, the ABS management architecture will be described in detail, trying to explain how different aspects from other initiatives regarding the management of CORBA-based applications have influenced its design.

3. The ABS Management System Architecture

Figure 2 shows the ABS Management System architecture developed for the second version of the ABS brokerage service architecture. The following subsections are devoted to the description of the main issues of the developed management system.

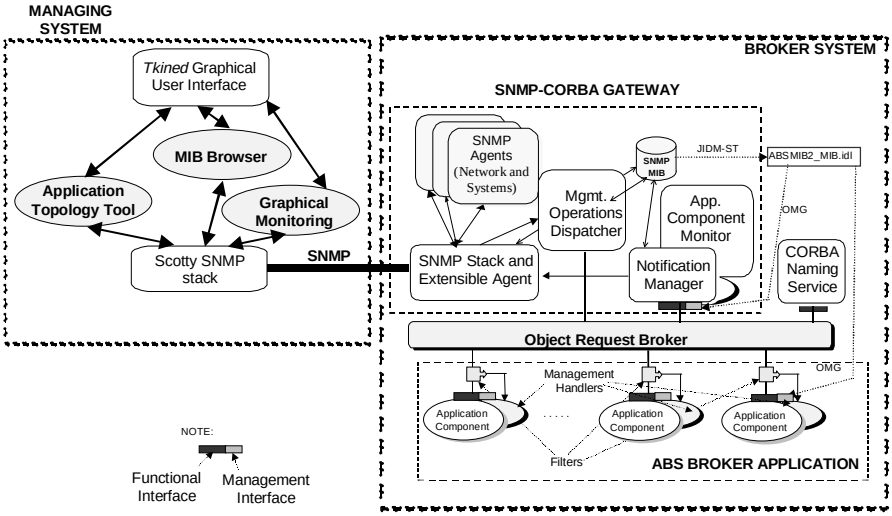

Figure 2: The ABS Management System Architecture

3.1 The Management Information Model

Specific ABS service and application management information has been defined according to the rules of Internet SMI (Structure of Managed Information) [16]. The use of Internet SMI for the definition of the management information in this kind of scenarios (SNMP management of a CORBA application) is also suggested in [9].

One different SMI group has been defined to be maintained by each broker application CORBA component. In those SMI groups there are different types of management information variables:

- Management information related to the behaviour of the particular application component itself.
- Management information related to the behaviour of the application as a whole. This management information is maintained by those application components that control the interactions of the broker application with its environment. For instance, the so-called USA (User Service Agent) can maintain information about the average duration of the requests delivered by the users.
- Management information about the configuration of the management part of that particular CORBA application component (management configuration management). By means of that management information the application component knows, for instance, how to send notifications, whether it has to send monitoring notifications or not, etc.

Most of the defined management information for the ABS broker application components consists of simple SNMP variables. Nevertheless, some SNMP tables have also been defined. These tables are used to store management information regarding the set of "User Agents" and "Content Provider Agents" created when different Users and Content Providers gain access to the broker application. These agents are created by two broker architectural components: the USA (User Service Agent) and the CPSA (Content Providers Service Agent). The way the information contained in these SNMP tables is accessed and maintained has been a key issue in designing the SNMP-CORBA gateway and the instrumentation of the management information in the ABS broker application (as it will be described in sections 3.2 and 3.3).

For network and system management already existing definitions of managed information has been used.

3.2 The SNMP-CORBA Gateway

This is the central component of the whole ABS management system architecture.

The SNMP-CORBA Gateway has to give support for the following functionality:

- It has to receive and decode SNMP PDU's.
- It has to decide which application component maintains the information related to the SNMP received request.
- It has to generate the appropriate CORBA request to perform the received request and collect the corresponding CORBA responses and generate the subsequent SNMP PDU's to be sent to the SNMP managing system.
- It has to collect events generated by the application components and send them back to the managing system.

In order to support the above functionality, the SNMP-CORBA gateway was structured in the following blocks (depicted in Figure 2).

The SNMP Stack and Extensible Agent

This SNMP-CORBA gateway block receives SNMP requests (from the managing system) and, by using some configuration information, redirects them to some other specific software component capable of processing them. The goal is the separation of requests regarding network, system application, and service management.

From an implementation point of view, this block consists of a JAVA SNMP extensible agent so as to facilitate its integration with OrbixWeb (the JAVA implementation of the CORBA platform that was used for the implementation of the ABS broker application). The SNMP requests can be redirected to other SNMP agents running in different ports (this approach is used for network and system management) or to the *Management Operation Dispatcher* that is described in the following paragraphs.

Management Operations Dispatcher

This SNMP-CORBA gateway block receives management operations requests from the *Extensible Agent*, locates the Broker System component that can satisfy the request, redirects the requests as CORBA operations, collects the results and returns them to the *Extensible Agent* in a way suitable for the generation of SNMP responses. Two different versions of this component were developed.

In the first version of the ABS management system [17] (developed for the prototype of the first version of the ABS broker system architecture) a very simplified *Management Operations Dispatcher* was used. It was based on a very fixed syntactic relationship established between the SMI MIB file containing the definition of the management information and the configuration of the ABS broker CORBA application. For instance, each application component needed to be maintained by only one CORBA server with the same name than the corresponding MIB group. This fixed relationship also implied that the application components had to maintain very simple management interfaces based on generic "get" and "set" methods. Therefore, the resulting management system based on this gateway had two important drawbacks:

- the compulsory use of the generic management methods made the handling of SNMP tables very difficult as those generic interfaces didn't allow the retrieval of a particular element of a table (it wasn't possible to indicate the coordinates of a particular requested element), and
- the management system would only be useful for CORBA applications structured in a way similar to the ABS broker application (one CORBA component in only one server with a fixed name, etc.)

The use of new management interfaces customised for the handling of tables was considered but, instead of that, for the second version of the *Management Operations Dispatcher* (developed for the more complex second version of the ABS broker application), several concepts, principles and procedures of the JIDM [6,8,9] solution were adopted for the development of the SNMP-CORBA gateway.

The JIDM proposal for achieving interoperability among the SNMP, OSI Management and CORBA domains from the management point of view is composed of two parts:

- The JIDM Specification Translation [6] which defines algorithms for the different combinations of translations among management information specifications written in OSI/GDMO, SNMP/SMI and CORBA/IDL.
- The JIDM Interaction Translation that defines a set of CORBA facilities intended to help in the achievement of interoperability among the CORBA, OSI and SNMP management domains. The JIDM IT defines both generic management CORBA facilities and management domain specific CORBA facilities (devoted to the OSI-CORBA [8] and SNMP-CORBA interoperability [9]).

For the functionality required for the management of the ABS broker application, the following JIDM aspects were found to be useful and therefore were adopted:

- The JIDM Specification Translation algorithms for the translation of SNMP/SMI specifications into CORBA/IDL specifications.
- The following items of the SNMP part of the JIDM Interaction Translation [9]:
  - The *SMI Repository Service* which maintains information about the OID hierarchy of the SNMP MIBs and the textual names associated with each OID in it.
  - The mechanism proposed for the mapping of names of SNMP variables to a hierarchical name tree based on the *CORBA Naming Service* specification.
  - The *SNMP Naming Service* Specification which is an extension of the *CORBA Naming Service* (adding the capability of lexicographically ordering of entries).

Nevertheless, the *SMI Repository Service* and the *SNMP Naming Service* were not exactly implemented as they are defined by JIDM. In order to simplify the development of the *Management Operations Dispatcher* and to reduce the number of CORBA interactions needed to support management operations (and thus, to improve the performance of the management systems) the following simplifications were made:

- Instead of implementing the *SMI Repository Service* its functionality was provided by the direct access to the information stored in the SMI MIB file.
- Instead of implementing a new *SNMP Naming Service*, the *CORBA Naming Service* was directly used and the extensions intended to be provided by a generic *SNMP Naming Service* were directly implemented in *the Management Operations Dispatcher* itself.

The drawback associated to this simplification is the lack of flexibility in the obtained solution: there are not general *SNMP Naming and SMI Repository Services* that might eventually be accessed by other CORBA objects. But, as this is not going to happen in the ABS environment, the simplified solution is much better due to its simplicity and its better performance.

The Notification Manager

This SNMP-CORBA gateway block is responsible for the reception of notifications from the applications components and their translation to SNMP Traps by using the services of the *SNMP Stack and Extensible Agent*. It is not based on the *SNMP Notification Service* defined in [9] because the implementation of this service was too complex compared with the simple needed functionality, regarding notifications, in the ABS environment: all the application components send their notifications to the *Notification Manager* (there are not other possible destinations for the notifications generated in the broker system) and all the received notifications are directly translated to SNMP Traps and sent to a single

SNMP managing system. Furthermore, only a limited event filtering capability was needed in order to specify whether notifications about activation or deactivation of components had to be forwarded to the managing system or not.

The *Notification Manager* uses typed notifications defined in a similar way to what is proposed in current OMG works on a generic *CORBA Notification Service* [19] and implements a simple method with a single operation to accept the generated notifications.

The Application Components Monitor

This block periodically checks what ABS application components are up and running and maintains a table with that information. This table can be accessed through the management interface of the *Notification Manager*. Furthermore, the *Application Components Monitor* produces notifications in order to inform the managing system about the changes in the status of the application components. From and implementation point of view, due to its close interrelation both the *Application Components Monitor* and the *Notification Manager* are located within the same CORBA server.

3.3 The CORBA management interfaces and the instrumentation of management information

In the ABS Management System, the approach of the TINA Management Architecture [4], which is also suggested in [9] is adopted for the instrumentation of the defined SNMP MIB in the CORBA application. In this approach, computational objects provide two types of interfaces: one functional interface and one management interface. The management interfaces for the different application components are obtained by applying the JIDM Specification Translation algorithms to the defined ABS SNMP MIB. For this step the SMI to IDL translator developed by Lucent Technologies [20] was used.

When developing the ABS SNMP MIB, it became apparent that three types of management variables (all of them accessible trough the IDL interfaces obtained from the application of the JIDM ST algorithms) could be distinguished according to the way their value might be obtained:

- Management variables associated to the number of invocations of particular application component methods (methods of the IDL interfaces). For instance, the number of accesses to a database.
- Management variables associated to the average duration of a particular operation (associated to an application component method). For instance, the average time needed to resolve a user request.
- Management variables related to some internal parameter of an application component. For instance, the number of internal failures of an application component.

As it was explained in section 2, the development of the different application components should not be affected because of the introduction of the support for the management functionality. In other words, from an implementation point of

view, the management instrumentation should be as transparent as possible to the developers of the managed application components.

Taking into account the above requirement, the following two complementary mechanisms were chosen to support the instrumentation of the three types of management variables previously described:

- The instrumentation support for the first two types (those associated to the number of invocations or the average time consumed by an IDL method) was achieved by using (as in [21]) a proprietary mechanism provided by OrbixWeb 3.0: the so-called "Orbix Filters". These filters can be considered as an implementation of the concept of "Interceptor" included in version 2.2 of the CORBA specification. The "Orbix Filters" allow the processing of CORBA requests and responses before and after they are passed to the application components. These filters are completely transparent to the developer of the managed application component.
- The instrumentation support for the third type (that associated to some internal parameter of the application component) is based on the value of some internal variables provided by the application component developer's code (the instrumentation of this type of management variables cannot be completely transparent to the developer).

From an implementation point of view it is important to point out the following items:

- As OMG IDL does not support the definition of multiple interfaces for a single CORBA object, an inheritance relationship was established between the classes implementing the functional and management interfaces thus obtaining a class with the methods of both interfaces. The implementation of the logic of the management interfaces is supported by the so-called "Management Handlers" classes (schematically shown in Figure 2). Therefore the developer of an application component only has to derive its implementation class from the corresponding "Management Handler". The "Management Handlers" contain member attributes devoted to the storage of the values of the variables of the corresponding SMI group. These attributes, whose values are accessed through the operations of the management interfaces implemented by the "Management Handlers", can also be modified by the code of the implementation class of a particular application component. In this way, the application developers take the control of the management information without being concerned about how it is accessed.
- The "Management Handlers" also instantiate the corresponding "OrbixFilters" which are used to store management information about the use and time consumption of application components functional methods. The management information obtained by the filters is also accessible through the management interfaces implemented by the "Management Handlers".

Both the “Management Handlers” and the “OrbixFilters” are also used to obtain information about the configuration and the dynamic behaviour of the managed application:

- When the *“Management Handlers”* classes are instantiated, they send a notification to the *Notification Manager* which forwards it to the *Application Component Monitor* (described in section 3.2) to inform it about the activation of a CORBA application component (this configuration information can subsequently be accessed by the managing system)
- The filters can “capture” operations received and sent by a particular CORBA server. When a filter detects an operation, it can send a notification to the managing system by means of the *Notification Manager*. The managing System can use this information to track CORBA requests and to graphically display the interaction among application components.

3.4 The Managing System

The SNMP managing system has been developed using the *tkined/scotty* [23] management platform. On top of this platform, a set of small management application was developed in order to customise its functionality for the management of the ABS Broker System application. From a management operator point of view, the ABS managing system can perform the following tasks:

- Browse the ABS MIB. Each SNMP request will be forwarded by the SNMP-CORBA Gateway towards the proper ABS Broker System component.
- Graphically monitor network, system and application SNMP variables.
- Monitor the status of the application components.
- Graphically and dynamically show the invocation of CORBA operations by using the SNMP traps associated to the notifications generated by the filters.
- Obtain Brokerage Service management information from the combination of several application management parameters.

4. Conclusions and Future Work

In this paper the architecture of the management system for the ABS brokerage application has been presented as an example of how a CORBA-based application can be managed by means of existing SNMP management platforms and tools. JIDM principles have played a key role in the design of this management system mainly in the phase of the management information models translation. The JIDM-IT ideas had to be applied in a simplified way in order to reduce the complexity and potential performance problems in the development of the SNMP-CORBA gateway.

During the integration of the ABS management system with the ABS broker system prototype, two main conclusions were obtained:

- The way the management information has been instrumented has proven to be a very good one as the developers of the ABS application components didn’t have to be aware of the management part of the overall system.

- The introduction of the management system, even with the modification explained above, produced important application performance degradation. In addition to the performance problems, the development process also suffered from the already known problem of the great number of source files generated by the JIDM-ST algorithms [24]

We are currently working in different issues in order to improve the functionality of the developed management system:

- Automatic generation of "Management Handlers" taking the MIB definition as the starting point.
- Improvement of the *Notification Manager* in order to implement the JIDM *SNMP Notification Service*.
- Separation of the JIDM-IT functionality from the gateway components in order to develop generic CORBA services (as described in [9])

Acknowledgments

The authors wish to thank the whole ABS team at DIT-UPM and the ABS project partners for their invaluable help and excellent work.

References

[1]. Object Management Group. "The common object request broker Architecture and Specification". Revision 2.2. OMG document formal/98-07-01. February 1998.
[2]. ITU-T Draft Recommendation X.901. "Basic reference model of open distributed processing – Part 1: Overview and guide to use".
[3]. ISO/IEC. "Open distributed management architecture". Committee Draft. ISO/IEC JTC1/SC21 X.708, June 1996.
[4]. TINA Consortium. "Management architecture". Version 2.0. December 1994.
[5]. Network Management Forum. "OMNIPoint Integration Architecture", Forum TR114, 1995.
[6]. The Open Group. "Inter-domain management: specification translation". Open Group Preliminary Specification P509. March 1997.
[7]. Subrata Mazumdar, and Ken Swanson. "WEB Based Management - CORBA/SNMP Gateway Approach". Seventh IFIP/IEEE International Workshop on Distributed Systems: Operations & Management, L'Aquila, Italy, October 28-30, 1996.
[8]. Object Management Group. "CORBA/TMN interworking final submission – JIDM interaction translation". OMG document *telecom/98-05-02*. May 1998.
[9]. Object Management Group. "CORBA/TMN interworking final submission – JIDM-SNMP". OMG document *telecom/98-05-03*. May 1998.
[10]. Object Management Group. "CORBA/TMN interworking RFP". OMG document *telecom/97-09-04*. September 1996.
[11]. Juan I. Asensio, José I. Moreno, Víctor A. Villagrá, Jesús Redondo,

Alexandre Nolle, Ionel Tothezan, George T. Karetsos. “Application of TINA-C computing and service architecture concepts to the development of an advanced information brokerage service in the context of EC”. *TINA’97 Conference*. Santiago, Chile. November 1997.

[12]. Juan I. Asensio, Víctor A. Villagrá, José I. Moreno, Julio Berrocal. “An approach to electronic brokerage in TINA environments”. 5th International Conference on Intelligence in Services and Networks (*IS&N 98*). Antwerp, Belgium. May 1998.

[13]. ABS Consortium. “Broker system version 2”. *Deliverable D32*. April 1997. More information about the ABS project can be accessed at *http://b5www.deteberkom.de/ABS/*

[14]. M. Schoffstall, M. Fedor, J. Davin, J. Case. “A simple network management protocol (SNMP)”. Internet Request for Comments 1157. October 1990.

[15]. Alexander Keller, "Towards interoperable management architectures: making SNMP management platforms suitable for CORBA". Proceedings of the 5th HP OpenView University Association Plenary Workshop (HPOVUA'98), ENST Bretagne, Rennes, France. April 19-21, 1998.

[16]. J. Case, K. McCloghrie, M. Rose and S. Waldbusser. “Structure of management information for version 2 of the simple network Management Protocol (SNMPv2)”. Internet Request for Comments 1902. January 1996.

[17]. Juan I. Asensio, Víctor A. Villagrá. "A simplified approach to the management of a CORBA-based electronic brokerage application". Proceedings of the 5th HP OpenView University Association Plenary Workshop (HPOVUA'98), ENST Bretagne, Rennes, France. April 19-21, 1998.

[18]. A.Schade, P. Trommler, M. Kaisersweth. “Object instrumentation for distributed applications management”. Proceedings of the IFIP/IEEE International Conference on Distributed Platforms (ICDP ’96), Dresden, Germany. 1996.

[19]. Object Management Group. “Updated revised joint notification service submission”. OMG document *telecom/98-03-06*. March 1998.

[20]. Subrata Mazumdar, Ken Swanson, and Helena Sarin. “WEB based management: CORBA/SNMP gateway approach”. *http://nsm.research.bell-labs.com/~mazum/CorbaSnmp*

[21]. Fraunhofer Institut Informations-und Datenverarbeitung. “The CORBA-Assistant: monitoring of CORBA-based applications”. White Paper. June, 1997.

[22]. M. Debusmann, R. Kroger, C. Weyer. “Towards an automated management of distributed applications”. Proceedings of the IFIP International Working Conference on Distributed Applications and Interoperable Systems (DAIS’97). BTU Cottbus. October, 1997.

[23]. J. Schönwälder, H. Langendörfer. “Tcl extensions for network management applications”. 3rd Tcl/Tk Workshop. Toronto, Canada. July 1995.

[24]. N. Lynch, K. Hyland. “Web Enabled TMN Manager for an international trouble ticketing service”. 5th International Conference on Intelligence in Services and Networks (*IS&N 98*). Antwerp, Belgium. May 1998.