

# AN INTEROPERABLE FEDERATED NAMING SERVICE - SUPPORTING A PAN- EUROPEAN SERVICE PLATFORM

Aart T. van Halteren<sup>1</sup>, Patricia Tangney<sup>2</sup> and Vincent Walsh<sup>3</sup>

## Abstract

This paper describes the implementation of a Federated Naming Service required for a widely distributed multi-ORB heterogeneous platform. This platform is being built to demonstrate and test the applicability of CORBA technology and TINA principles to support Telecommunication Public Network Operators (PNOs) in their attempt to open their interfaces to third party service providers.

The introduction of a Naming Service requires the definition of a naming hierarchy and configuration of ORBs to interwork with the Naming Service. A distribution strategy for the Naming Service in the ESP was devised to minimise the cost of name look ups and to increase reliability of the platform. The support for federation of Naming Services i.e. interoperability of existing Naming Services is assessed.

A federated Naming Service is the primary bootstrapping mechanism that enables the application objects throughout Europe to find each other. Unfortunately, the interworking between the available Naming Service implementations can not be achieved due to difference in Repository Ids. without further extensions to the Naming Services.

## 1. INTRODUCTION

The EURESCOM<sup>4</sup> P715 Project<sup>5</sup> is building an experimental environment, the EURESCOM Service Platform (ESP), which exploits the Common Object Request Broker Architecture (CORBA) middleware technology and is based on the architectural principles from Telecommunication Information Network Architecture (TINA).

---

<sup>1</sup> KPN Research, PO Box 15000, 9700 CD Groningen, The Netherlands, Email: A.T.vanHalteren@research.kpn.com

<sup>2</sup> Broadcom Éireann Research Ltd., Kestrel House, Clanwilliam Pl., Dublin 2. Tel : +353-1-6046000

<sup>3</sup> Broadcom Éireann Research Ltd., Kestrel House, Clanwilliam Pl., Dublin 2. Tel : +353-1-6046000

<sup>4</sup> European Institute for Research and Strategic Studies in Telecommunications GMBH -owned by 23 Public Network Operators from 22 European Countries

<sup>5</sup> P715 commenced in March 1997 and is due for completion in February 1998. P715 participants include the Telecom Public Network Operator in Britain, Finland, France, Holland, Ireland and Germany

This paper contains an interim result from the building of this multi-ORB heterogeneous platform. It describes a federation of heterogeneous CORBA Naming Services implementations which is required by the ESP. The paper describes the limitations to the interoperability of the Naming Services currently available within the ESP. It describes how these limitations will be overcome for the ESP, in order to provide a reliable and efficient Naming Service for the EURESCOM Service Platform.

A reliable and responsive Naming Service is a key issue for a large distributed system such as the EURESCOM Service Platform. The heterogeneity of the ORB and Service implementations which must interwork within the ESP pragmatically test the interoperability claims of ORB vendors. Experiences from such interworking tests highlight the need for greater attention to be paid to the support for the federation services in CORBA Specifications. Interoperability tests are of interest to all potential investors in CORBA middleware technology. The interoperability experiences described in this paper are limited to the that of interoperating Naming Service implementations.

Section 2 describes the EURESCOM Service Platform infrastructure and the rationale for building this platform. Section 3 describes the necessity for enhancing the basic ESP platform with a Naming Service. Section 4 gives a brief overview of the Naming Service Specification from the OMG. Section 5 describes the approach taken to building a federated Naming Service for the ESP and discusses the interoperability issues which arose from implementing this Federation.

## **2. P715 EURESCOM SERVICE PLATFORM (ESP)**

The EURESCOM P715 project is building an experimental environment, EURESCOM Service Platform (ESP), which exploits Object Management Group (OMG) Common Object Request Broker Architecture (CORBA) middleware technology and is based on the architectural principles from Telecommunication Information Network Architecture (TINA) from TINA-Consortium (TINA-C). TINA based services promise the ability to allow network and service providers to interoperate.

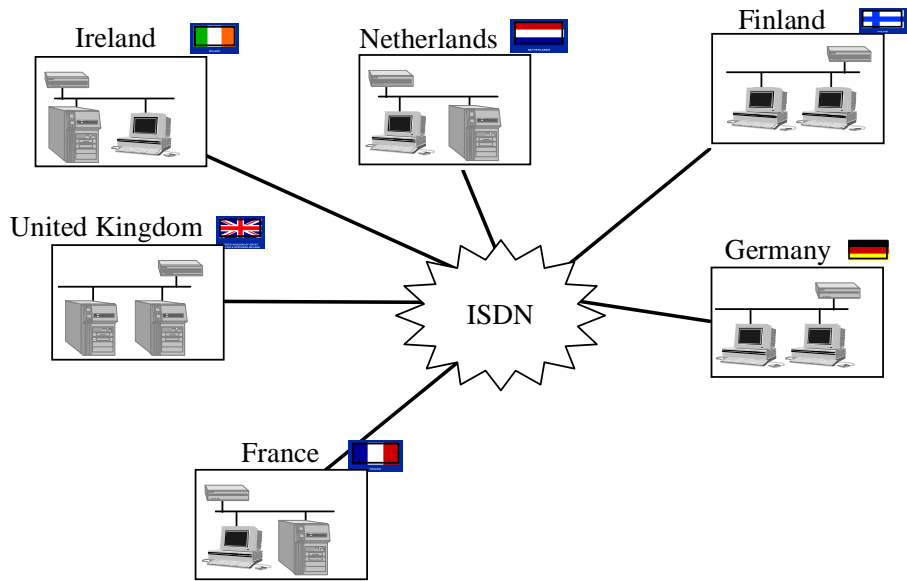
TINA-C, a consortium of PNOs, telecommunication vendors and IT vendors, have developed an open architecture for telecommunication services including voice, interactive multimedia, mobile and information services. TINA includes a Distributed Process environment (DPE), and an object-oriented service and management architecture. P715 is building an implementation of the TINA Service Architecture, to support the management of services to be deployed on the ESP.

The ESP facilitates experiments investigating and demonstrating the value of distributed object technologies and platforms for multimedia services and multimedia services management.

The ESP is distributed, with one or more nodes on each of the project participants sites i.e. Telecom Éireann/Broadcom<sup>6</sup>, BT (British Telecom) , NL (KPN Royal PTT Nederland NV), AF (FINNET Group, Finland) , FT (France Telecom) and DT (Deutsche Telecom). The ESP nodes are currently interconnected through a central router at EURESCOM Headquarters in Heidelberg, using Internet Protocol (IP) over the public ISDN network (Figure 1).

---

<sup>6</sup>Telecom Eireann is being represented by Broadcom Eireann Research Ltd. within the P715 project



**Figure 1** - Network connectivity in the ESP

The Object Request Broker (ORB) implementations on the ESP nodes interwork using the Internet Inter ORB Protocol (IIOP). Table 1 illustrates the heterogeneity of the ORB implementations, hardware and software platforms of which the ESP is composed. Most of the participant ESP sites are utilising more than one ORB implementation.

The OMG has standardised a range of services, CORBA services [1],[2], which extend the functionality of the core CORBA standard, and will be leveraged by the ESP. The ESP is being developed and tested incrementally. This paper describes the enhancement of the ESP with a Naming Service. The ESP could be extended with other commercially available products providing CORBA Services, e.g. persistence, trading, security, transactions.

Testing the interoperability of ORB and CORBA Service implementations will help demonstrate that PNOs can benefit from distributed object technologies and middleware platforms through such characteristics as interoperability, portability, reliability and extensibility.

**Table 1** ESP ORB Implementations

ORB Implementation	Operating System	Hardware
COOL ORB	Solaris 2.5.1	SPARC 20, SPARC ULTRA 2
NEO 2.0	Solaris 2.5.1	Sun Ultra 1
Orbix 2.2 MT	Solaris 2.5.1	SPARC ULTRA
Orbix 2.2 MT	Windows NT 4.0	PC Pentium 2x200,
OrbixWeb 3.0	Solaris 2.5.1	SPARC ULTRA
OrbixWeb 3.0	Windows NT 4.0	Pentium 90

Visibroker 2.5 (C++)	Solaris 2.5.1	Sun Ultra 1
Visibroker 2.5 Java	Solaris 2.5.1	Sun Ultra 1

### 3. ENHANCING THE ESP WITH A NAMING SERVICE

Obtaining an object reference is a prerequisite before any object interaction can take place, the ORB handles location of objects based on object references. Early in the development of the ESP, the requirement for a Naming Service emerged. During the first phase of ORB interoperability tests for the ESP nodes, CORBA Interoperable Object References (IORs) were distributed by project participants using WWW or e-mail. This is most unsatisfactory, because :

- not all ORBs implement persistent IORs for servers, and therefore a new IOR is generated and had to be re-distributed every time a server is rebooted;
- copying a IOR from an e-mail message or from the Internet is tedious and prone to errors;
- as the number of services grow, the number of objects grow, causing numerous IORs to be published.

The CORBA Naming Service provides the principal mechanism through which clients of an ORB-based system can locate objects that they intend to use. Section 4 gives a brief description of the Naming Service. Section 5 describes the ESP Naming Service which requires a federation of the naming services on the ESP nodes. Section 5.4 describes the difficulties encountered when try to develop an interoperable federated naming service. Section 5.5 describes the future direction of the ESP federated Naming Service.

### 4. CORBA NAMING SERVICE

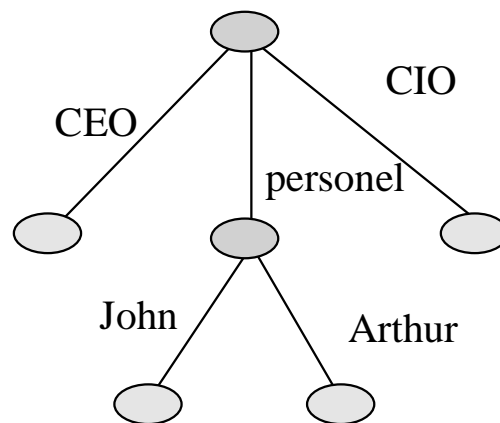
The CORBAservices Naming Service allows a human readable name to be associated or *bound* to an object. The reference to that object can subsequently be found by *resolving* that name within the Naming Service. This section contains a very brief description of the Naming Service, further details can be found in [1].

Using the Naming Service a name is bound to an object relative to a naming context. Different names can be bound to an object in the same or different contexts at the same time, this is called a name binding. A naming context is an object that contains a set of name bindings in which each name is unique. In file management terms, a naming context is basically a directory structure for objects. A name is always resolved relative to a context, there are no absolute names. To resolve a name is to determine the object associated with the name in a given context. To bind to a name is to create a name binding in a given context.

Because a context is like any other object, it can also be bound to a name in a naming context, thus creating a naming graph. A naming graph allows more complex names to reference an object. For an example of a naming graph, see Figure 2. Given a context in a naming graph, a sequence of names can reference an object. This sequence of names (called a compound name) defines a path in the naming graph to navigate the resolution process. The naming service provides the principal mechanism through which most clients of an ORB-based system locate

objects that they intend to use. Given an initial naming context, clients navigate naming contexts retrieving lists of names bound to that context.

A server registers an object reference with the Naming Service by binding the object reference to a naming context. This name can then be used by other components in the system to find the registered object.



**Figure 2 - A Naming Graph**

In addition to resolving object references, the Naming Service offers operations for managing naming contexts. A naming service offers one naming context, which is referred to as the root context. An object reference to the root context can be obtained by calling `CORBA::ORB::resolve_initial_references("...")`. This object represents the root of the tree of the naming Service. Operations on the Naming Service are usually invoked on the root naming context, but naming contexts which are created relative to the root can also be used.

#### **4.1. CORBA NAMING SERVICE IMPLEMENTATIONS**

The following CORBA Naming Service implementation are available on the ESP :

- OrbixNames 1.03(C++)
- SUN NEO (C++)
- Visigenics Naming Service for C++ v2.5
- Visigenics Naming Service for Java v3.0
- CoolORB (C++)
- OmniBroker (C++)

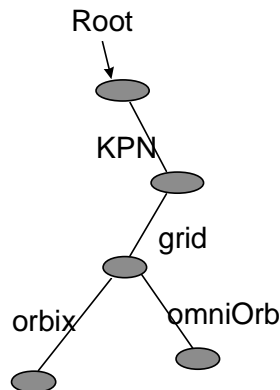
### **5. ESP FEDERATED NAMING SERVICE**

Each CORBA 2.0 compliant ORB should support the `resolve_initial_references()` call, allowing each partner to access the naming services available at their ESP node. In addition to using a standard mechanism for exchanging IORs, the introduction of a Naming Service in the ESP provides an interesting assessment of the Naming Service specification. To enhance the ESP with the Naming Service it was necessary to do the following :

- to define a naming hierarchy (Section 5.1)
- to configure the heterogeneous ORBs to interwork with the Naming Service (Section 5.2)
- to define a distribution strategy for the Naming service in a wide-area network i.e. the ESP (Section 5.3)
- to assess the interoperability of the existing implementations of the Naming Service (Section 5.4)

## 5.1. A NAMING HIERARCHY

Before publishing IORs in the naming service, it is important to have an agreement which names to use. It was decided to have a naming context for each partner, so references to objects running on a particular partners' site can be published under this context. Figure 3 represents a part of the naming tree. The circles represent naming context objects, the squares represent IORs and the lines represent a binding. The IOR to the grid object running at the KPN site, which is implemented using Orbix can be retrieved by calling *resolve("KPN.grid.orbix")* on the root naming context.



**Figure 3** - A Federated Naming Service

The way the naming tree is organised does not say anything about the location of naming context objects. The sub-tree of BT can easily be maintained in a Naming Service implementation that differs from the Naming Service implementation which maintains the sub-tree of TI. To distinguish between the Naming Service as described in the CORBA standard and the way this is implemented as operating system processes, we use the term Naming Server to describe the entity that implements the specification

## 5.2. CONFIGURING THE ORBS

Since the root context can be resolved by calling *resolve\_initial\_references("NameService")*, each ORB must be configured to return the correct root context. The table below shows how each ORB can be configured.

ORB	Configuration for Naming Service
-----	----------------------------------

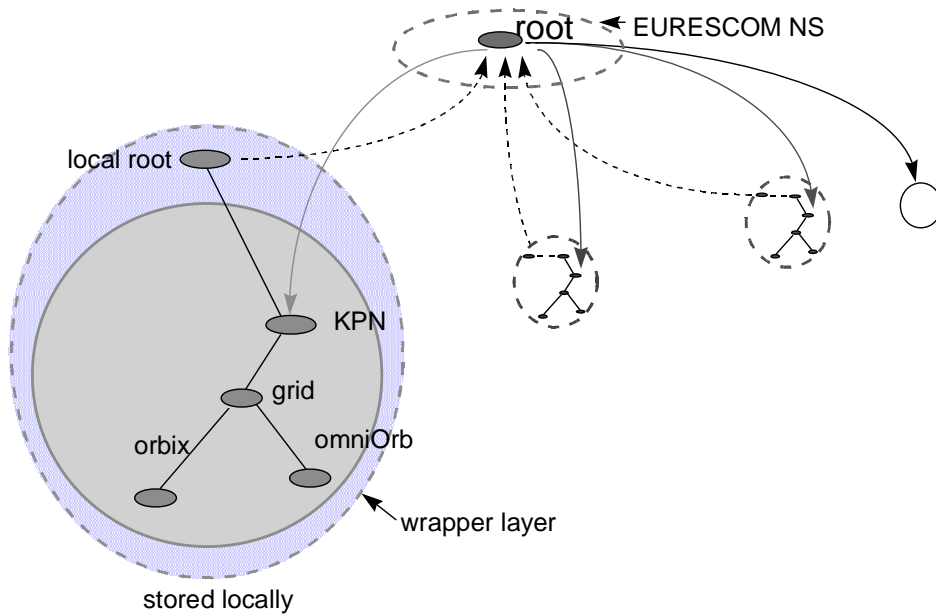
<b>implementation</b>	
<b>Orbix</b>	Orbix can be configured through the locator, which can be told at which node an implementation is expected to run.
<b>NEO</b>	NEO does not support the <i>resolve_initial_references</i> call but has a proprietary way for resolving a reference to the root naming context
<b>Visibroker for Java</b>	The Visigenic Naming Service is started by a Naming Factory. The reference of the Naming Service may be obtained by using the Visigenic bind mechanism. This is applicable for Visigenic based clients only.
<b>Visibroker for C++</b>	Version 3.0 does support <i>resolve_initial_references()</i> for use with the Visibroker Naming Service. The root context (in the Visibroker NS) can be set for a particular client program on initialisation of the ORB (via a command line option, for example).
<b>COOL ORB</b>	COOL ORB does not support <i>resolve_initial_references()</i>
<b>OmniBroker</b>	The root context can be set by an application parameter or by use of a method of class CORBA_ORB::naming ( <appropriate IOR> );

**Table 2** - Configuration of ORB implementations to incorporate the Naming Service

### **5.3. DISTRIBUTION STRATEGY**

In a Wide Area Network (WAN), such as the ESP, it is very important to decide where Naming Servers are operated. A single Naming Server, at a central location, could easily become a performance bottleneck. This would imply that whenever a Name must be resolved, an IIOP call must be made to this central server. This IIOP call must then be transported over the ISDN lines to the host which runs the Naming Server. Potentially, the ISDN connection can be down, if there has not been any other traffic over this line for a while. The initial set-up time for an IP connection over ISDN can be around 10 seconds, which implies that the time taken to resolve an IOR becomes undesirable.

A distribution strategy, which would minimise the number of WAN IIOP calls, has been devised for the ESP. The basic idea is to store local names at a local Naming Server, if a name cannot be resolved within the local Naming Server then a central Naming Server (installed at the EURESCOM Node) will attempt to resolve the name within the other partners Naming Services. Re-use of existing NamingServers can be achieved by building a thin wrapping layer around such servers. The wrapper forwards all methods of the naming context object to the local NamingServer. If the local Naming Server can't resolve a name, the wrapper catches the exception and goes to a global NamingServer, which is running at a central location. This global NamingServer (EURESCOM NS) maintains federation links to all the local NamingServers, therefore enabling a global view on the whole naming tree. Figure 4 shows how the local and central NamingServers should interwork.



**Figure 4 - Wrapping existing NamingServers to increase performance**

A major benefit of the wrapper approach is that local object references can be resolved locally, therefore eliminating the need for a WAN IIOP call. Remote object references can still be retrieved, through the use of a global NamingServer.

An additional benefit of the wrapper approach, is that the overall system becomes more resilient against partial failures. If there is a temporary break-down in the ISDN connections, or the machine running the global NamingServer has a break-down, local objects can still be found through the local NamingServer. The global NamingServer could even be replicated, reducing even further the chance of an overall failure of the Naming Service. However in order for this approach to work successfully a Naming Service implementations must be interoperable, difficulties in implementing this approach have arisen due to the inability to interoperate Naming Services from different vendors. Section 5.4 describes these interoperability problems in more detail.

#### 5.4. INTEROPERABILITY OF NAMING SERVICES

The CORBA services Naming Service specification was written to be flexible in order that it can be used in a variety of systems. Unfortunately this flexibility has been interpreted differently by ORB vendors, and the result is that most naming service implementations we used are not interoperable, thus not allowing a federated naming service. This is mainly caused by differences in the RepositoryId for NamingContext objects. Table 3 gives an overview of the RepositoryId used by the various NamingService implementations.

Naming Service implementation	RepositoryID of Naming Context objects
OrbixNames 1.03	”IDL:CosNaming/NamingContext:1.0”
NEO	“IDL:op.sun.com/CosNaming/ACL_NamingContext:1.0”



<b>Visigenic Naming Service for Java 3.0</b>	”IDL:omg.org/CosNaming/NamingContext:1.0”.
<b>Visibroker for C++ 2.5</b>	“IDL:omg.org/CosNaming/NamingContext:1.0”
<b>COOL ORB</b>	“IDL:CosNaming/NamingContext:1.0”
<b>OmniBroker</b>	“IDL:omg.org/CosNaming/NamingContext:1.0”

**Table 3** - Differences in the RepositoryId

## 5.5. ENHANCING NAMING SERVICES FOR THE ESP

These interoperability problems became apparent when trying to implement the Federated Naming Service as described in Section 5.35.4. These interoperability issues are currently being addressed by the OMG. OMG has realised that the current Naming Service standard is too flexible and has released a Request for Proposal (RFP), titled “*Interoperability Name Service Enhancements*”, on the 4<sup>th</sup> Dec. 1997. The timetable on the Request for Proposals (RFP) states that the specification should be approved by the OMG Architecture Board in July 1998. Implementation will then follow, but unfortunately too late for this project.

The ESP in this project needs a federated naming service and cannot wait until the new RFP is standardised. Therefore it will be necessary to enhance the currently available Naming Service implementations to support their interoperation. The project is currently working on a developing a “wrapper” as depicted in Figure 4.

## 6. CONCLUSIONS

In the European Service Platform a reliable and responsive Naming Service is a key issue for the platform. It is the primary bootstrapping mechanism that enables the application objects throughout Europe to find each other.

Unfortunately, the interworking between the available Naming Service implementations can not be achieved due to difference in Repository IDs (See Table 3). It is also interesting to note that not all ORB implementations have implemented the standard CORBA 2.0 call *resolve\_initial\_references*(“*NameService*”). (See Table 2). This call should return an object reference to the root context of the Naming Service and make the interworking between applications and Naming Service independent of the ORB and Naming Service implementation.

Building wrappers around the current Naming Service implementation seems a good way forward to achieve a reliable and responsive Naming Service for the service platform. In the near future, experiences from this wrapper approach will be available.

## 7. REFERENCES

- [1] Common Object Services, Volume 1, OMG Document Number 94-1-1, 1994
- [2] Common Object Services, Volume 2, OMG Document Number 96-7-1, 1996.

- [3] The Common Object Request Broker: Architecture and Specification, OMG Document Number 97-09-01, 1997
- [4] Interoperability Name Service Enhancements Request For Proposal, OMG Document: orbos/97-12-33, 1997