



**Poseidon House
Castle Park
Cambridge CB3 0RD
United Kingdom**

TELEPHONE:
INTERNATIONAL:
FAX:
E-MAIL:

**Cambridge (01223) 515010
+44 1223 515010
+44 1223 359779
apm@ansa.co.uk**

APM Business Unit

ReTINA briefing for GPT Transaction Workshop

Chris Mayers

Abstract

GPT wish to apply transactional techniques in their new systems. A joint workshop between GPT, Newcastle University, and APM has been arranged for the 20 and 21 May 1996.

This presentation is a briefing about the deployment of the ORB and Object Services in ReTINA, specifically with respect to the JET project. It uses only public material from that project. It does not cover the CORBA Object Transaction service, since this is expected to be covered by Newcastle.

APM.1768.00.01

Draft

17th May 1996

Briefing Note

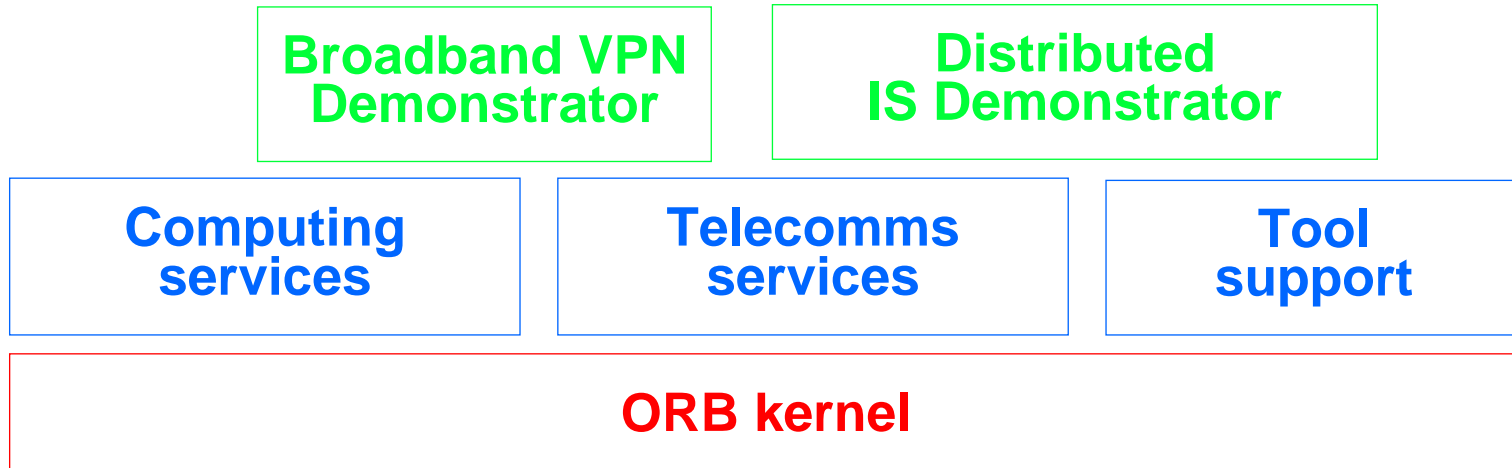
Distribution:

Supersedes:

Superseded by:

ReTINA

An Object Request Broker for Telecommunications



General Requirements

- **Conformance**
 - Upwards compatibility with CORBA 2.0
- **Modularity and Scalability**
 - an ORB everywhere
- **Flexibility**
 - choice of protocols, bindings, and policies
- **Performance**
 - real-time resource control and kernel determinism



Environments

- **ReTINA supports two environments**
 - **real-time: for resource and QoS management**
 - **general-purpose: for long-term persistent data**
- **Both use (interoperable) ORBs**



Services - Technical Approach

- **Data Management**
 - via integration of O₂ OODBMS
- **Query and Persistence services**
 - via O₂ objects
- **Transaction services**
 - via O₂ transactions embedded in a CORBA-compliant Object Transaction Service



The CORBA Object Transaction Service

- **The CORBA OTS is not bundled as a traditional TP monitor, or database**
 - these often handle the **Atomicity, Isolation and Durability of the ACID properties...**
 - ... **Consistency is always application-specific**
- **The CORBA OTS deliberately does not support**
 - **Isolation (handled by the Concurrency service)**
 - **Durability (handled by the Persistence service)**
- **The CORBA OTS is a transaction coordination framework, not simply a service you call**



CORBA Object Transaction Service - Implementations

- Possible implementations
 - on top of an existing TP monitor
 - integrated with ORB binding mechanisms (and the CORBA Event service)
 - integrated with ORB-specific mechanisms
 - integrated with specialized language mechanisms
 - driven declaratively



CORBA Object Transaction Service - Options

- **An implementation may (but need not) support**
 - **Nested transactions**
 - **Checked behaviour**
- **Omitting these simplifies the implementation**



Implications of the CORBA Transaction Service

- It is theoretically possible to implement the CORBA OTS entirely independently of an ORB
 - all the necessary interfaces are specified...
 - ...but performance would be disappointing
- We require engineering API hooks into a modular ORB
 - to support a portable OTS implementation...
 - with acceptable performance



Other implications of the CORBA OTS

- It provides the ACID properties
 - these are unnecessarily strong for some real-time applications
- It is based on the X/Open distributed TP model
 - this is important for interoperability with legacy applications and TP systems
 - ... but may involve too much overhead for some real-time applications



Distributed Transactions Are Expensive

- **The typical two-phase-commit protocol is too slow in many situations**
- **Conclusions**
 - **avoid distributed transactions if at all possible**
 - **exploit locality**
 - **use an application-specific distributed algorithm to control consistency instead**



Conclusions

- **The CORBA OTS is potentially a good fit to a modular ORB**
- **It will require specialized implementations for real-time**
 - **if application programmers are to see a conventional ACID transaction model**
- **Integration with binding mechanisms is a promising approach**
 - **but not the only one**
- **Similar issues will arise with the other CORBA Object Services**

