



---

**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**

---

## **ANSA Phase III**

# **Multi-media and real-time progress (MC 13/7/95)**

**Dave Otway**

### **Abstract**

This presentation gives an overview of the DIMMA multi-media and real-time work, plus a report on the C++ ODP API. The reports on the framework ORB and the stub generator can be found in APM.1522 and APM.1523

---

APM.1521.01.01

**Draft**  
Briefing Note

12th September 1995

---

**Distribution:**  
**Supersedes:**  
**Superseded by:**





# **Multi-Media and Real-Time (Progress Report)**

*Dave Otway*

*Guangxing Li*

*Youcef Laribi*



## Requirements

- **Add real-time capabilities to the ANSA/ODP architecture**
  - without destroying its ability to cope with:
    - federation, heterogeneity, scaling
- **Provide interoperability between real-time and non real-time objects**
  - predictable islands in a an unpredictable sea
- **Provide real-time guarantees in an asynchronous distributed system**
  - make high performance distributed systems
  - make predictable distributed systems
- **Application driver is multi-media**



## Approach

- **interaction models: client/server + streams**
- **invocation models: call/reply (RPC) + signals**
- **control model: asynchronous + synchronous programming**
- **binding model: implicit + explicit**
- **QoS model: addresses non-functional requirements**
- **scheduling model: resource separation, pre-allocation, deadlines**



---

## Proof of concepts

- **ANSAware/RT**
  - add prototype real-time mechanisms to ANSAware 4.1 with Posix Threads
- **added resource separation, pre-allocation, deadlines, priorities, real-time scheduling, Qos & explicit binding**
- **Performance gain (measured against ANSAware 4.1)**
  - null RPC 30% faster, 1000 byte RPC 16% faster
- **Available now over OSF/1, HP/RT, LynxOS**
- **Being ported to NT**



## Background

- ANSAware is not lightweight enough
- ODP Computational model is now an ISO standard
- CORBA is becoming the “standard” DPE product
- We can’t get immediate access to a full function CORBA source
- Phase III requires a prototype DIMMA DPE
- RETINA (and TINA-C) (and some sponsors) have a requirement for a Telecomms ORB with Real-Time capabilities
- DCAN requires a Real-Time DPE for ATM network management
- C++ is taking over from C as the “standard” implementation language
  - implementations of templates and exceptions are spreading



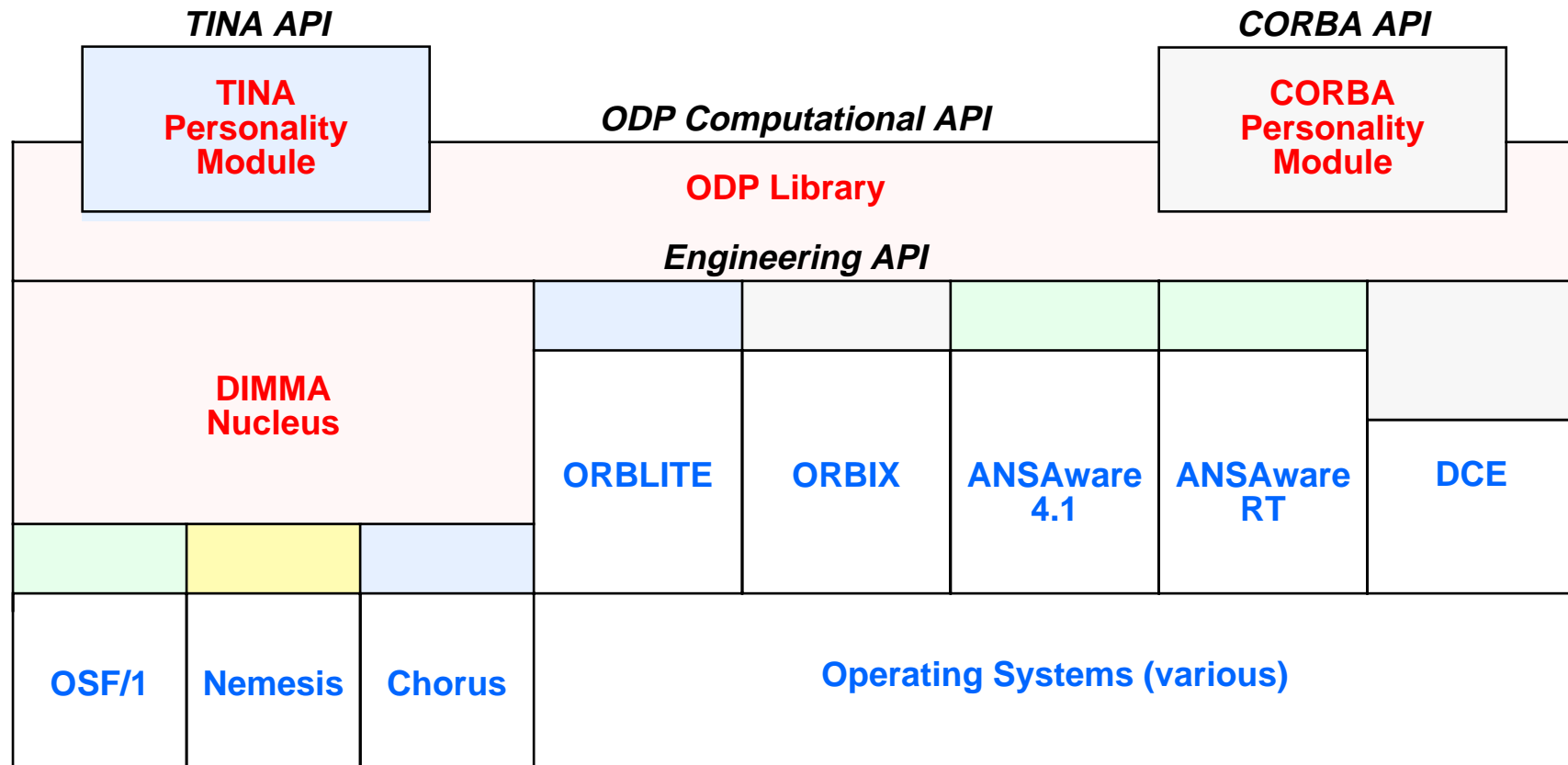
## Current Objectives

- build prototype DPE components that can be combined together to satisfy the requirements of all related projects with minimal overlap and reworking
- understand, explain and map the semantic differences between the different APIs, IDLs and DPEs
- design and implement a full function ODP conforming DIMMA DPE
- provide compatibility with CORBA and TINA
  - interoperability by mapping IDLs and protocols
  - application portability by mapping APIs
  - engineering portability by adapting to different underlying DPEs





# API & DPE Master Plan



Generic
Phase III
RETINA
DCAN
?



## Component reports

- **stub generation and type checking**
  - Youcef Laribi (APM.1523)
- **A lightweight ORB framework [ DIMMA Nucleus ]**
  - Guangxing Li (APM.1522)
- **A C++ ODP API**
  - Dave Otway (follows on)

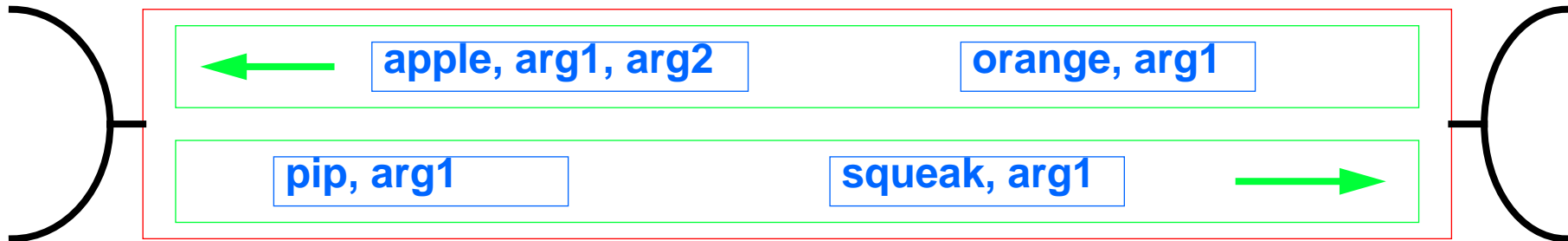


---

## Computational API

- **Stage 1: objects**  
(operational) interfaces  
multiple results  
basic types  
the type “Any”  
(operational) signatures  
invocation references  
named terminations  
local garbage collection  
(hand coded) trader client stub
- [ stage 1+ sequences implemented over DIMMA and ANSAware 4.1]
- stage 2: structured types, threads, service withdrawal
- stage 3: streams, synchronous programming, explicit binding, QoS
- stage 4: preprocessor

## Streams



- A stream has a set of flows
- A flow has a set of frames and a direction
- A frame has a name and a set of typed arguments
- Streams are typed and can be conformance type checked
- Frames are transmitted by non-blocking writes and read by blocking reads

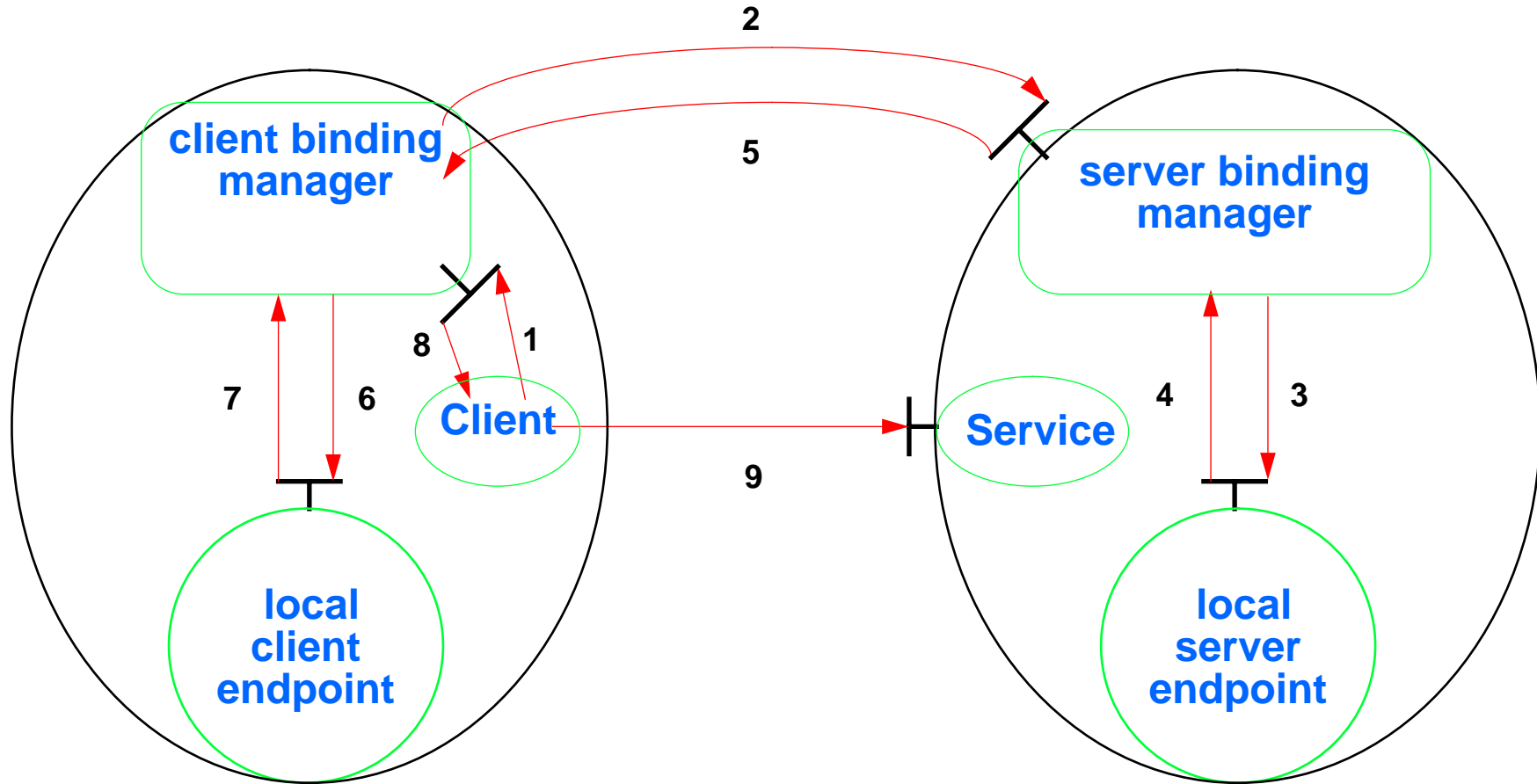


---

## Synchronous Programming

- **a reactive system continuously interacts with its environment**
  - **its execution is divided into a sequence of discrete instants**
  - **each instant reacts to its inputs and produces the corresponding outputs**
  - **reactions are synchronised with real-time by input time signals**
- **deterministic behaviour**
  - **bounded execution paths, calculable in advance**
  - **with guaranteed pre-scheduled resources:**
  - **programs have predictable timing and reproducible behaviour [ even in asynchronous systems ]**

# Explicit binding





## Related Activities

- **RETINA**
  - TINA computational API, engineering API, DIMMA Nucleus, Chorus and ORBLITE Adapters, stub generator
- **DCAN**
  - ODP computational API, engineering API, DIMMA Nucleus, OSF/1 and Nemesis adapters, stub generator
- **HP**
  - DIMMA engineering
- **GPT & GEC Avionics**
  - ODP computational API, engineering API, DIMMA Nucleus, stub generator
- **ICL**
  - DIMMA Nucleus, Stub generator, CORBA computational API



## Reading list

- APM.1555: ODP C++ API Design Overview**
- APM.1554: DIMMA Stub Generator Design and Implementation**
- APM.1553: An Overview of DIMMA nucleus**
- APM.1476: ANSAware/RT 1.0 Manual**
- APM.1393: Streams and Signals**
- APM.1392: The ANSA Binding Model**
- APM.1295 An Overview of the Distributed Interactive Multi-Media Architecture**