Poseidon House
Castle Park
Cambridge CB3 0RD
United Kingdom

TELEPHONE: **Cambridge (01223) 515010**
INTERNATIONAL: **+44 1223 515010**
FAX: **+44 1223 359779**
E-MAIL: **apm@ansa.co.uk**

ANSA Phase III

ANSAworks presentation - JET

Nicola Howarth

Abstract

APM.1752.01 **Approved** 22nd April 1996
External Paper

Distribution:
Supersedes:
Superseded by:

ANSAworks presentation - JET

ANSAworks presentation - JET

Nicola Howarth

APM.1752.01

22nd April 1996

The material in this Report has been developed as part of the ANSA Architecture for Open Distributed Systems. ANSA is a collaborative initiative, managed by Architecture Projects Management Limited on behalf of the companies sponsoring the ANSA Workprogramme.

The ANSA initiative is open to all companies and organisations. Further information on the ANSA Workprogramme, the material in this report, and on other reports can be obtained from the address below.

The authors acknowledge the help and assistance of their colleagues, in sponsoring companies and the ANSA team in Cambridge in the preparation of this report.

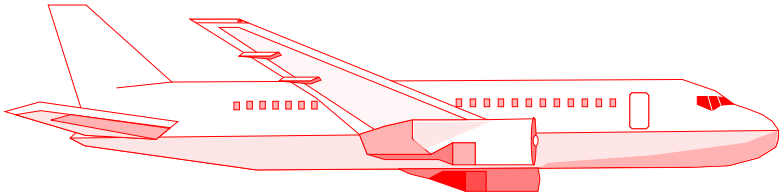
Architecture Projects Management Limited

Poseidon House
Castle Park
CAMBRIDGE
CB3 0RD
United Kingdom

TELEPHONE UK	(01223) 515010
INTERNATIONAL	+44 1223 515010
FAX	+44 1223 359779
E-MAIL	apm@ansa.co.uk

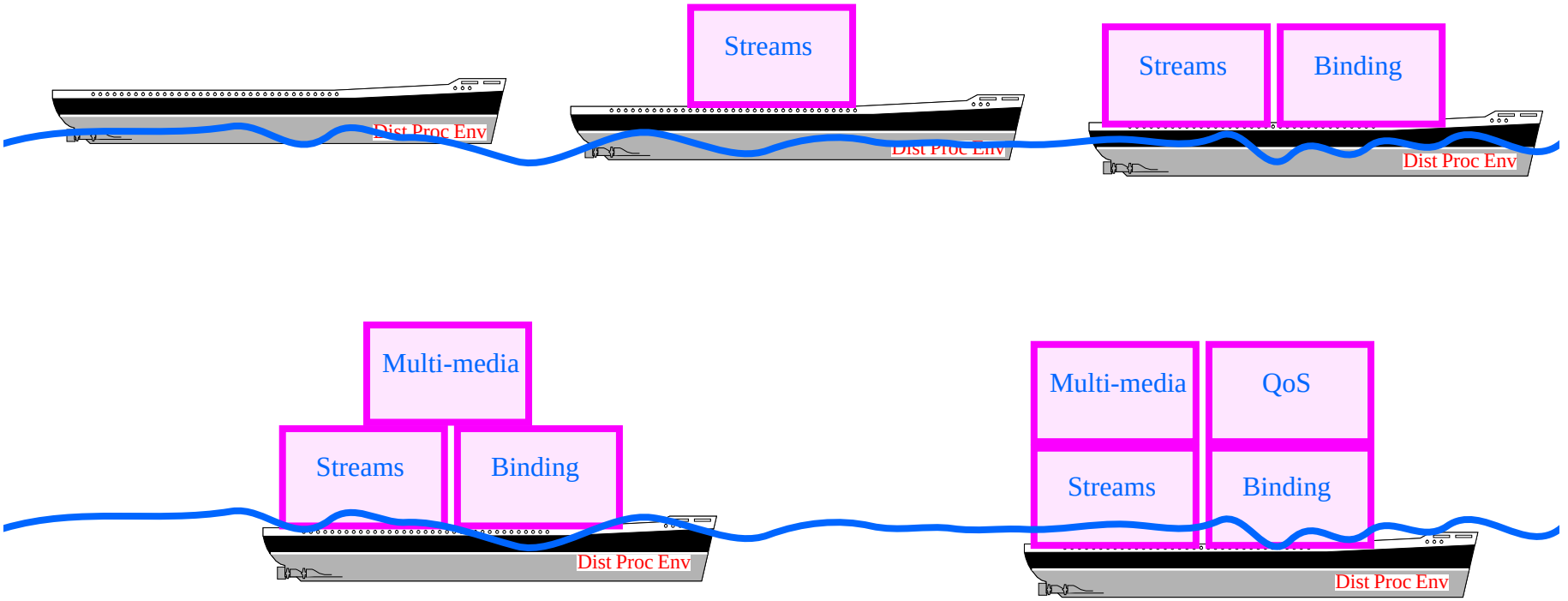
JET

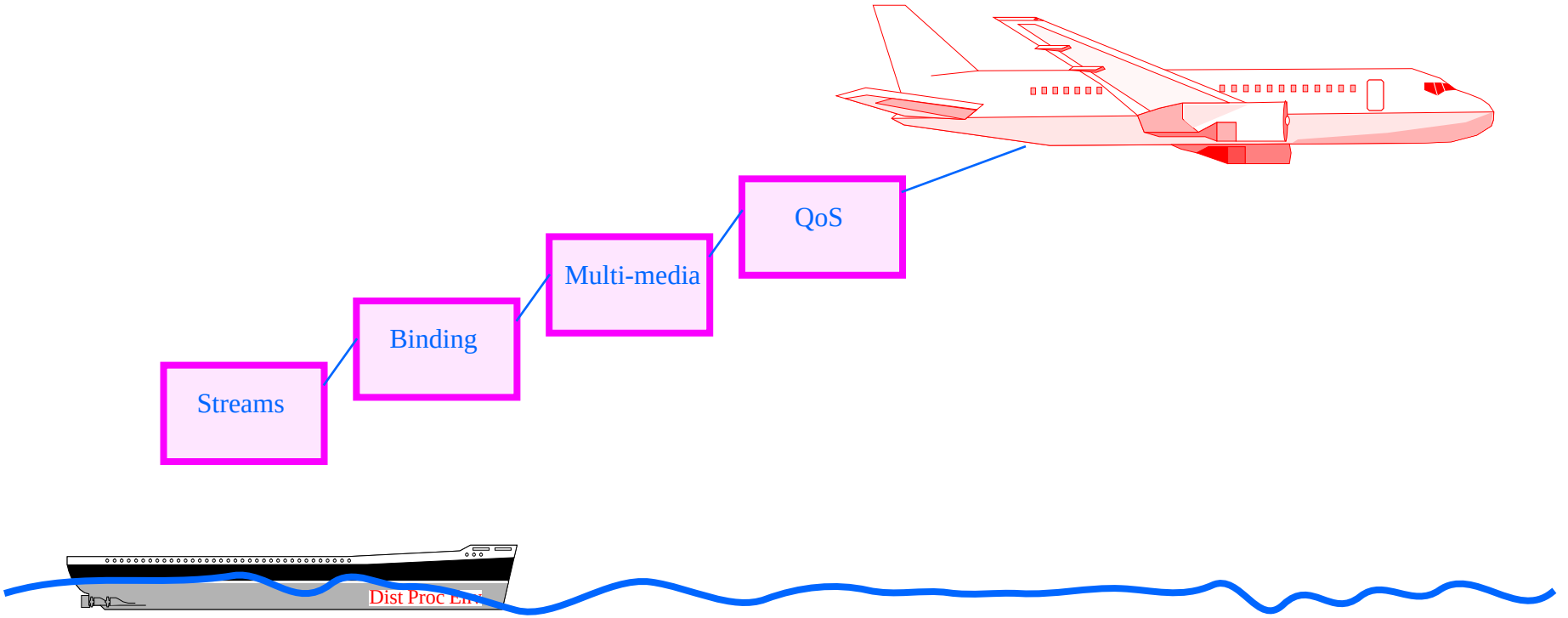
CORBA API on the DIMMA Platform

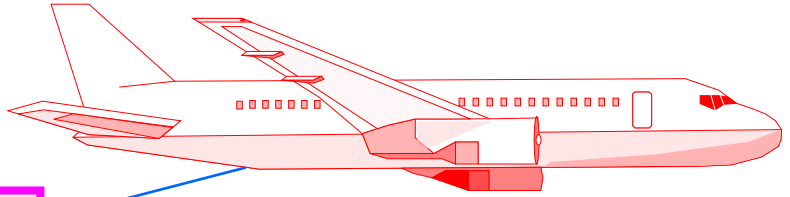


Nicola Howarth

Existing dist. proc. environments are not sufficient

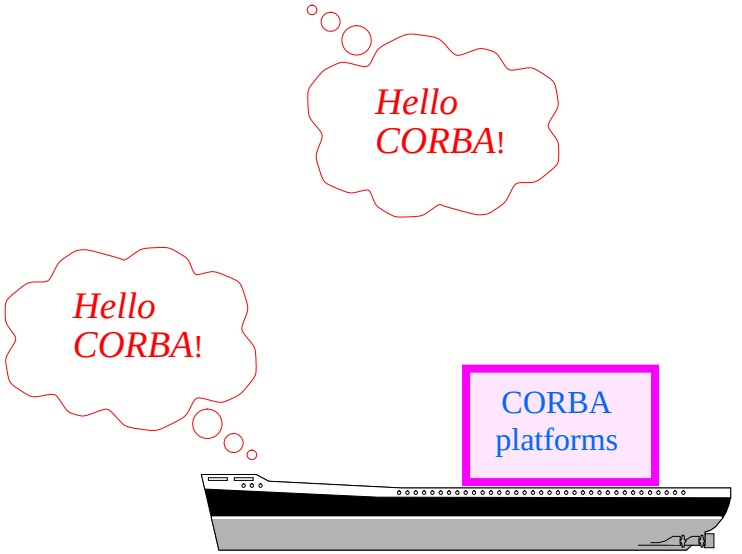



QoS
Multi-media
Binding
Streams



CORBA

Extensions


Hello
CORBA!
Hello
CORBA!
CORBA
platforms

Why JET?

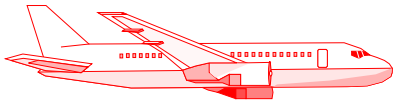
- CORBA is the standard - this is what people want

- CORBA on its own is not sufficient - no support for streams, binding, etc.

- The Distributed Multi-media architecture DIMMA which can provide this support is not CORBA-compliant

No answer

- so JET aims to provide a subset of CORBA API supported on the DIMMA environment

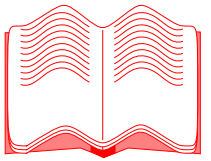


JET will provide:

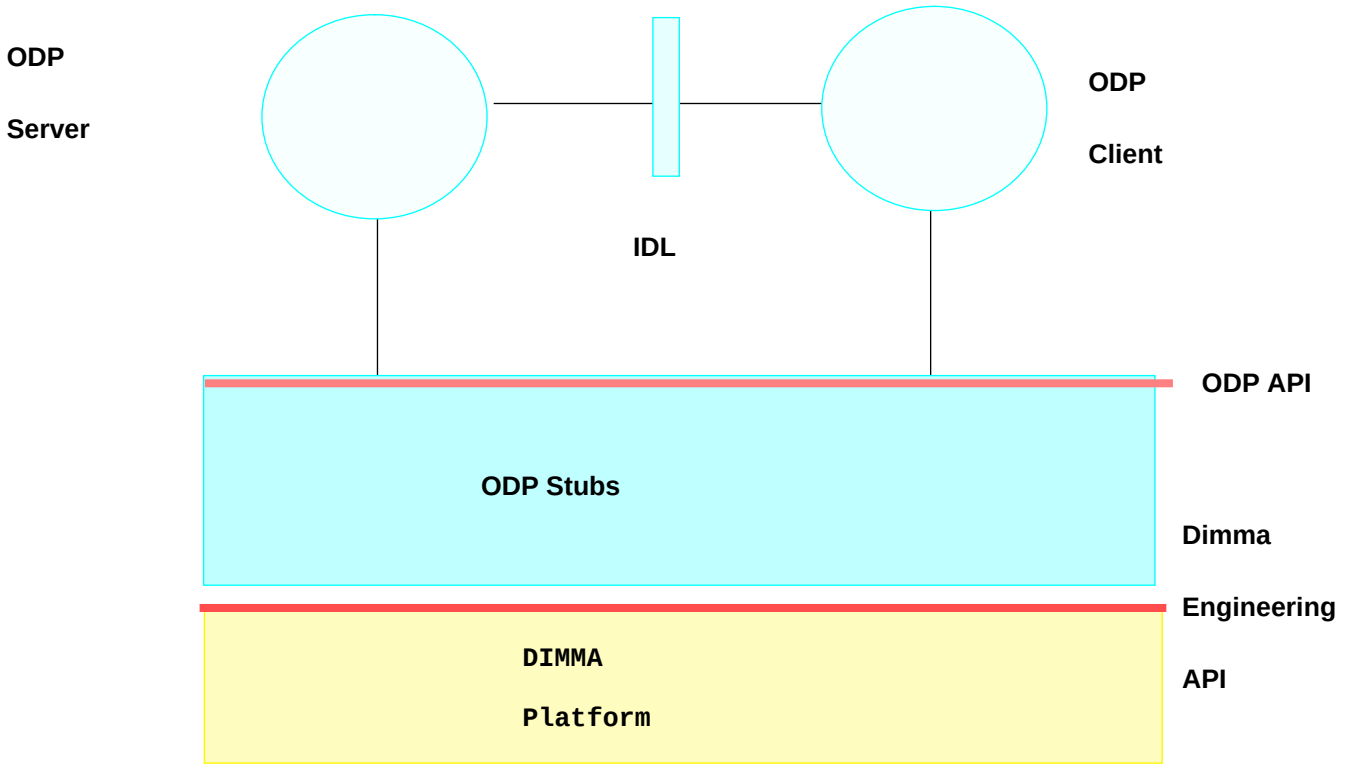
- a subset of a CORBA compliant platform for experimentation with extensions to CORBA for binding, streams, and so on

- inter-operability with and portability to other CORBA platforms

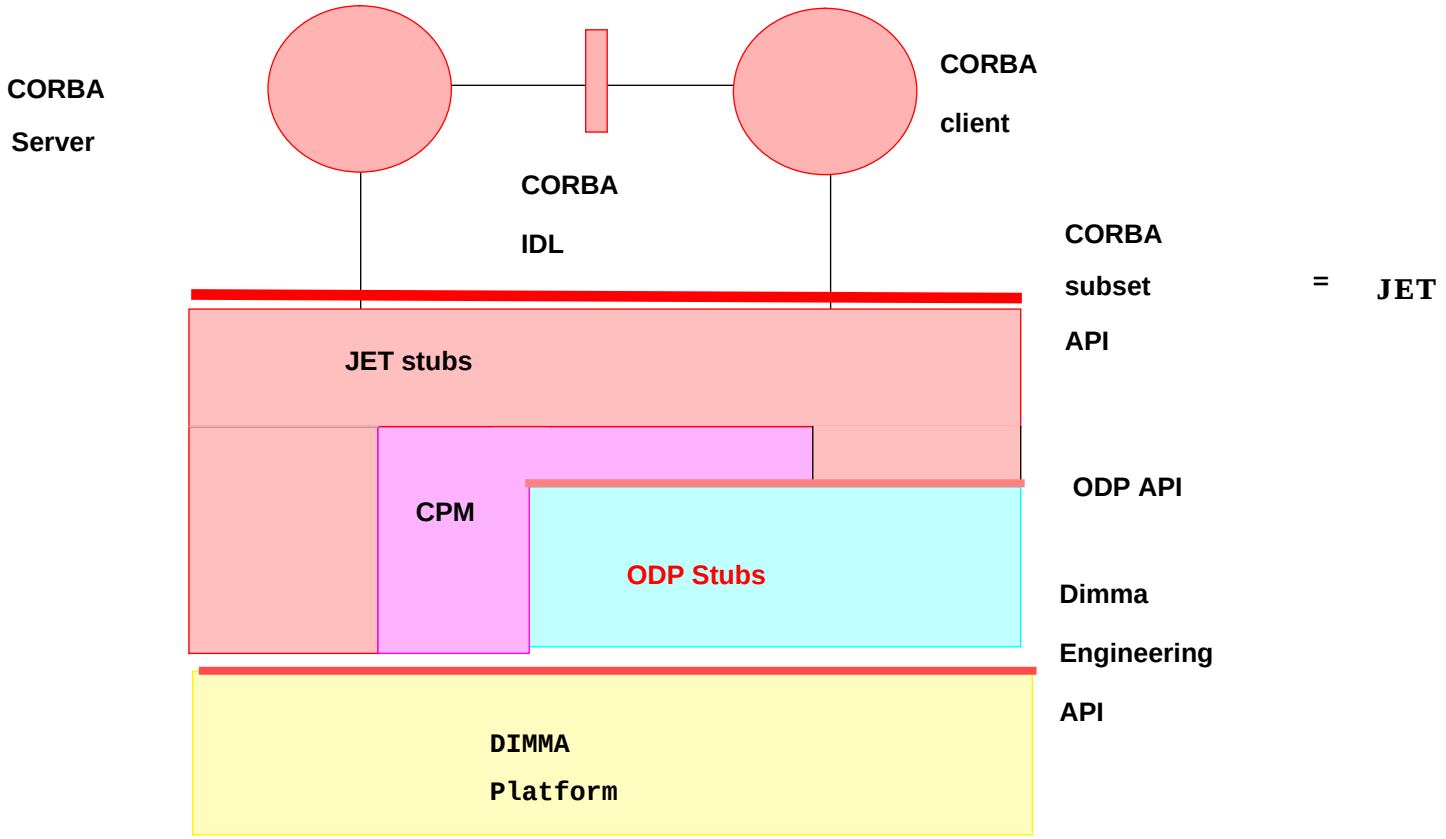
- conclusions from this work can be input to OMG



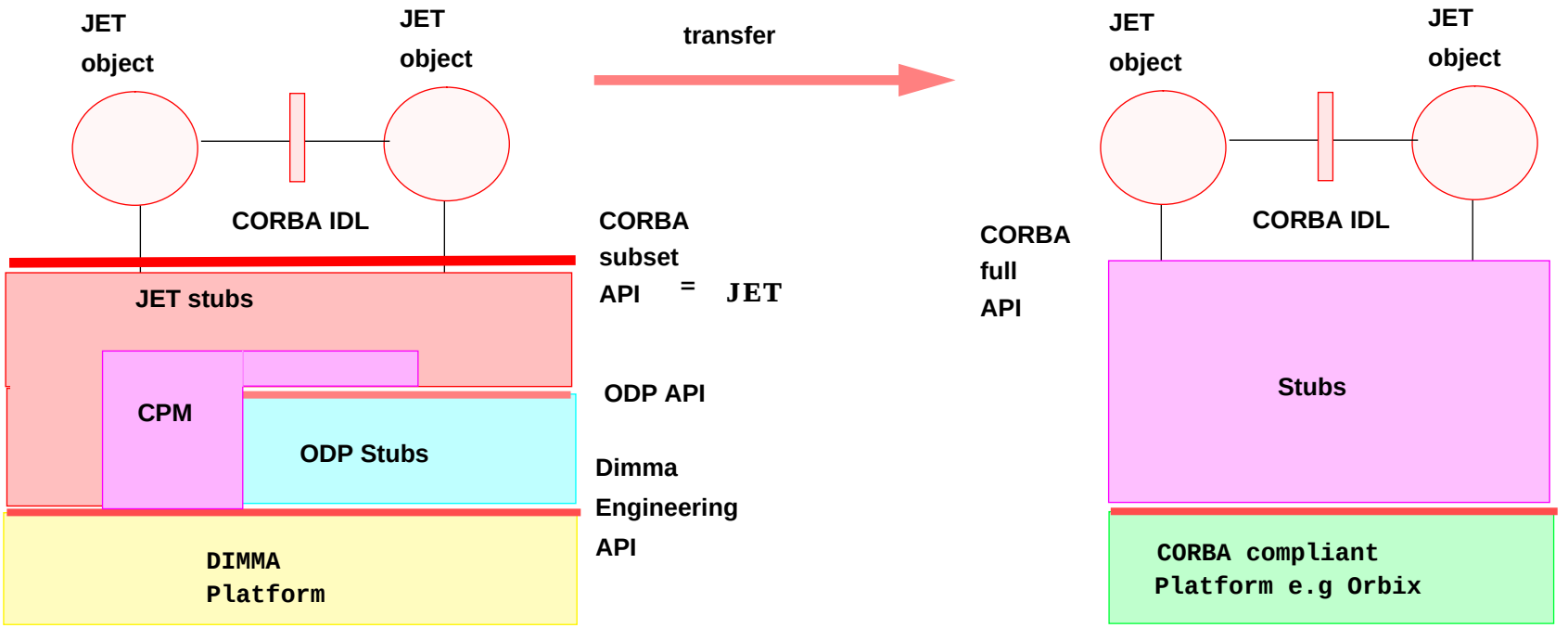
DIMMA and the ODP API



How JET works - CORBA and the ODP API



How JET works - JET and CORBA



Portability

- Other applications using same CORBA subset can port to JET
- JET applications without extensions can port to other platforms
- Minimal changes required for aspects not yet covered by CORBA
  - in the code (e.g. _bind, trading)
  - in the environment (eg. makefiles)

Interoperability

- JET applications will interoperate with
  - applications on the DIMMA/ODP platform
  - other CORBA platforms (using IIOP)

Why a Subset of CORBA

- full CORBA support is not necessary to provide the functionality needed
- the subset will be sufficient so as not to restrict the application programmer in writing distributed applications
- it will be extendable as further requirements become known
- it will avoid some less desirable features of CORBA

not included in the Subset

- context parameter
- attributes
- oneway invocations - will be provided by streams
- BOA object adapter
  - ill-defined
  - unnecessary - can be done implicitly and via explicit binding support
- concrete any
  - no type checking at compile time
  - difficult to implement
  - unnecessary - use union
- dumb pointers T* - impinges on memory management

Memory Management - T_var v. T*

- A CORBA API can be implemented using either T_var or T*
- you can do everything with one or the other
- you don’t need both

- T_var uses smart pointers with a reference count
- easy to implement garbage collection

- T* needs release/duplicate
- garbage collection harder

- support is provided for T_var only

Implementation

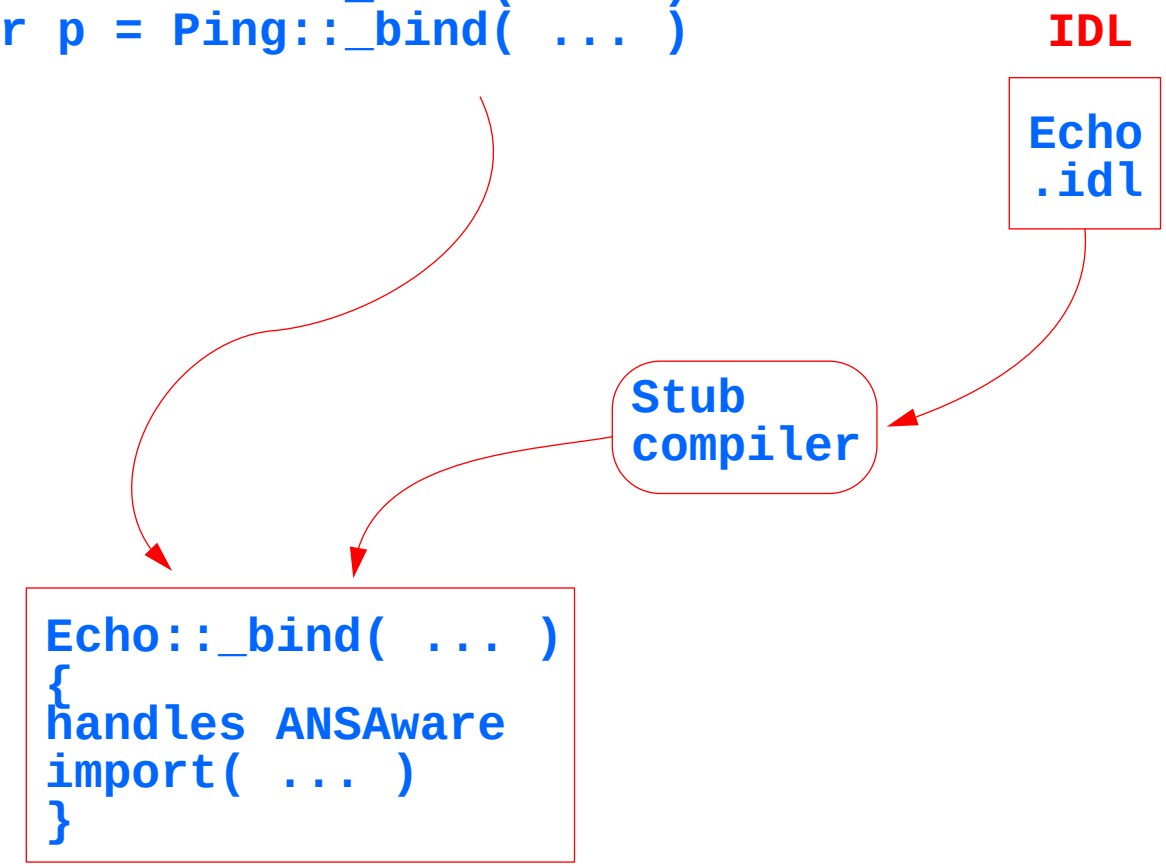
- stubs, generated from CORBA IDL by the stub compiler
- ODP API
- CORBA Personality Module (CPM)

Major Features

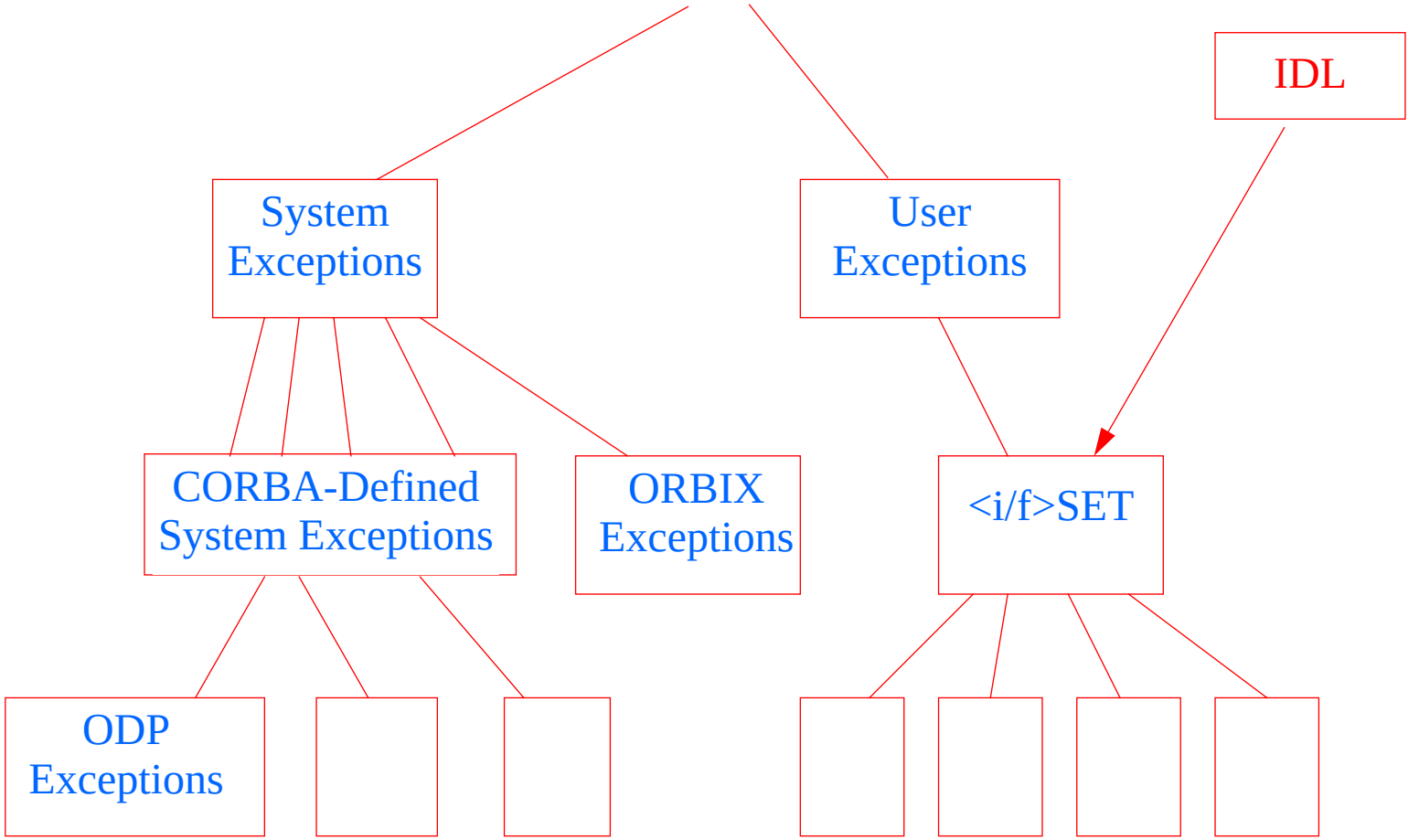
- support for all types other than any (structures, unions, arrays etc.)
- invocations (with in/inout/out)
- top level marshalling (underlying marshalling done in ODP API)
- _bind (trader import for client)
- exceptions

_bind

```
client:
Echo_var e = Echo::_bind( ... )
Ping_var p = Ping::_bind( ... )
```



Exceptions


IDL
System Exceptions
User Exceptions
CORBA-Defined System Exceptions
ORBIX Exceptions
<i/f>SET
ODP Exceptions

Summary

- JET is important in that it provides a basis for further work in all the areas mentioned
- Further details in APM.1657, “JET: CORBA API on the DIMMA platform”
- Contains: CORBA IDL stub compiler, CORBA API, CORBA personality module
- Demonstration will show how a CORBA program can be directly transferred from JET to another ORB