Prototyping a CORBA based MMS
-Industrial Communications with CORBA-

OMG-Manufacturing DTF

San Francisco September 2000

Eric Gressier-Soudan

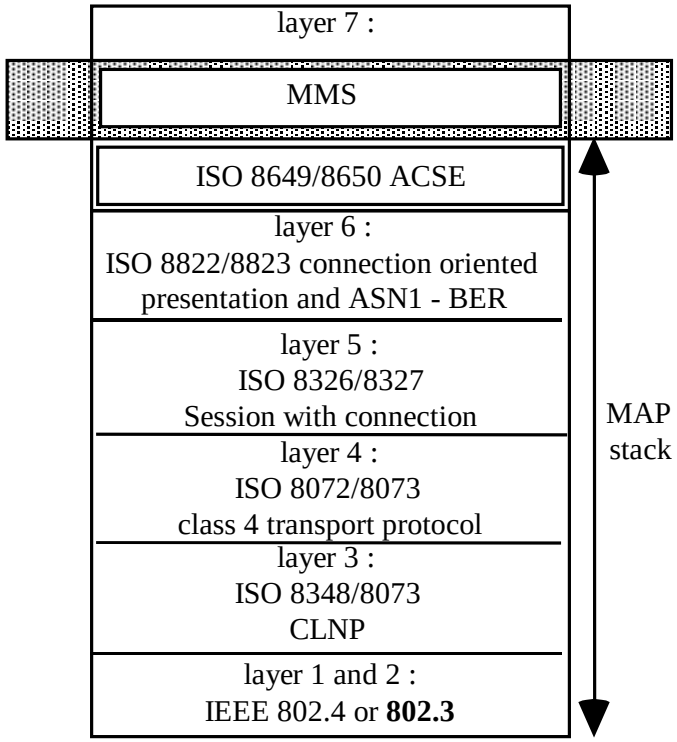
Laboratoire Cedric-CNAM
PARIS-France

Goal : An Open Architecture based on a middleware approach for Manufacturing Applications

We want to apply a Distributed System approach to build an industrial messaging service for manufacturing applications:

MMS like Object Oriented QoS aware ORB based solution
(QoS deals with communication support)

MMS and MAP from ISO

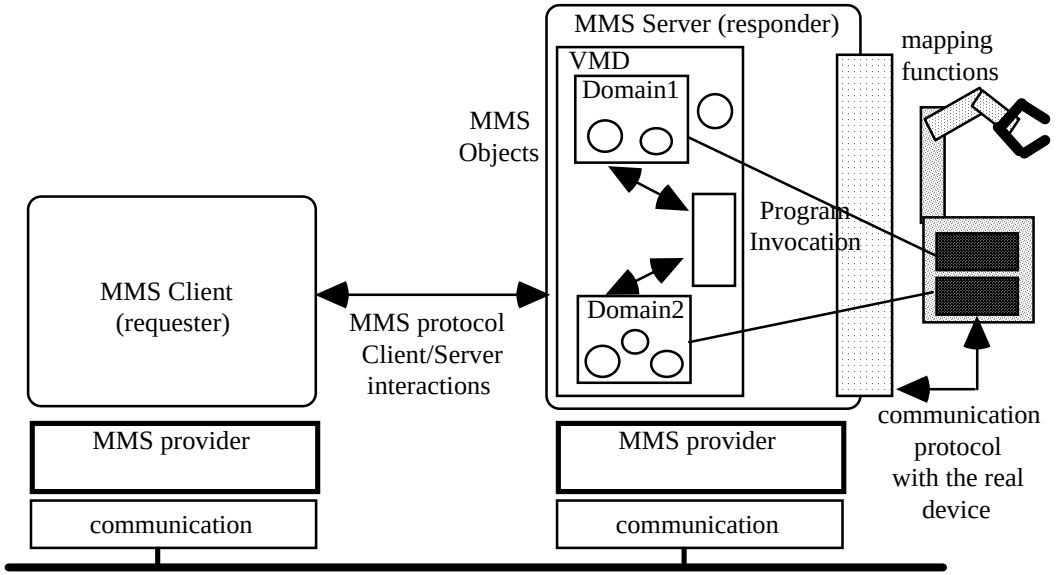


Defines mains functions of an industrial messaging protocol issued in the 90's

can be extracted from MAP

NO REAL TIME, NO OBJECT ORIENTED ISSUES addressed
Encapsulation exists and QoS deals with connection parameters (reliability …) no time related parameters for each services

MMS main features



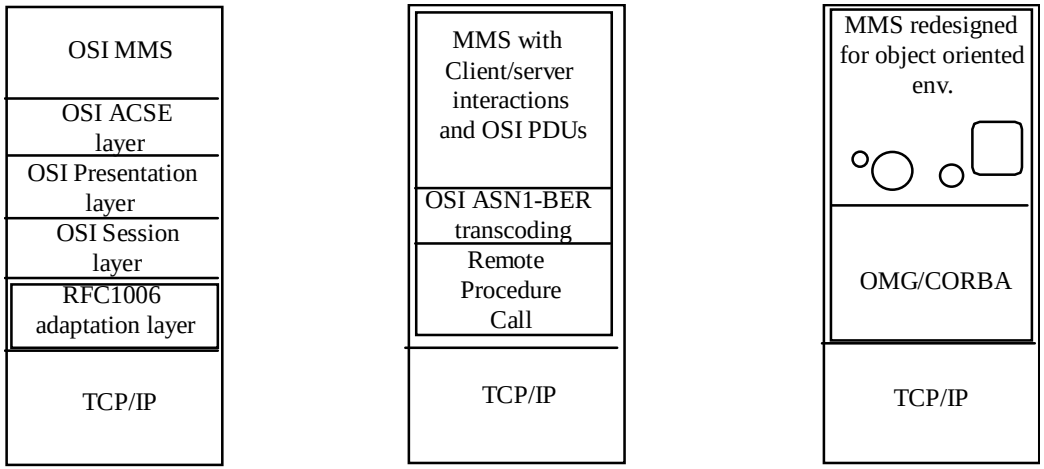
- Basic Abstractions : **Virtual Manufacturing Device** (**VMD**), **Domain**, **Program Invocation**, **Variable**, Semaphore, Event, File...

- OSI services defined for each abstraction

- Client/Server based Interactions

users' requirements issued from CCE-CNMA (93) and ITCIM(92) inquires Esprit projects:
"MMS over TCP/IP would be clever"

Ways to implement MMS over TCP/IP protocols



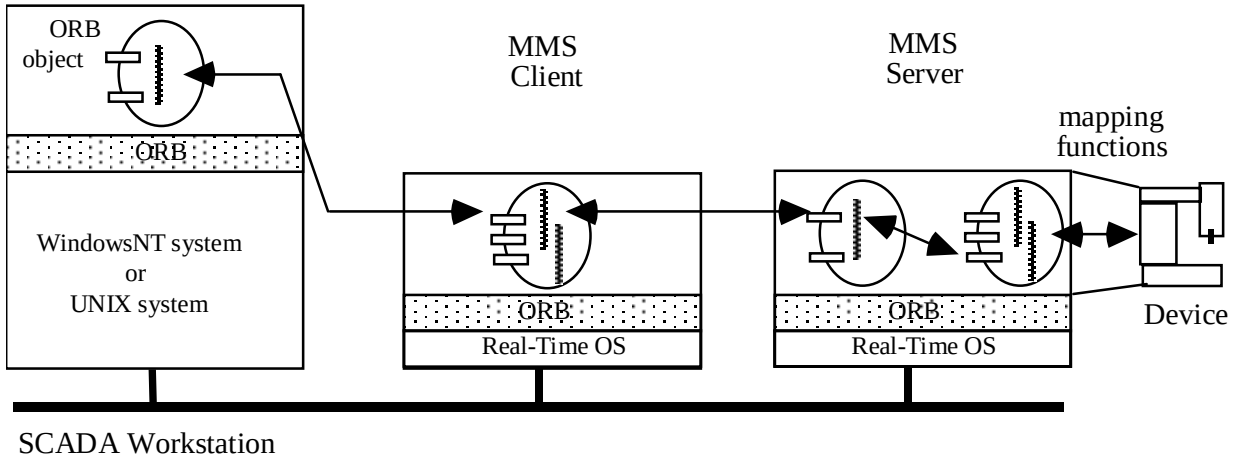
(a) -> ISO upper layers over TCP/IP, MMS conformance (RFC1006 based products today)

(b) -> RPC based middleware approach, partial MMS conformance limited to MMS PDUs, our first experiment used Sun's RPC and ASN1-BER transcoding functions

(c) -> extends (b) to OO environments, any compliance to the MMS standard is lost ! but the result is not so far from the ISO design : 5-7(low) MAP layers can be seen as a pre-ORB architecture ;-)

Solution (c) currently experimented

ORB based approach



- **Support of Hardware and System Heterogeneity :** Computers and Devices on the factory floor come from various suppliers.
- **Client/Server interactions**
- **Uniform object oriented approach for manufacturing applications** : OO design methodologies, OO programming language, and OO environment support
- **CORBA MMS can be integrated to Intranet/Extranet/Internet architectures :** benefit of IPV6, mobile IP, WEB, SNMP … for manufacturing (important in 96)

Advantages of CORBA

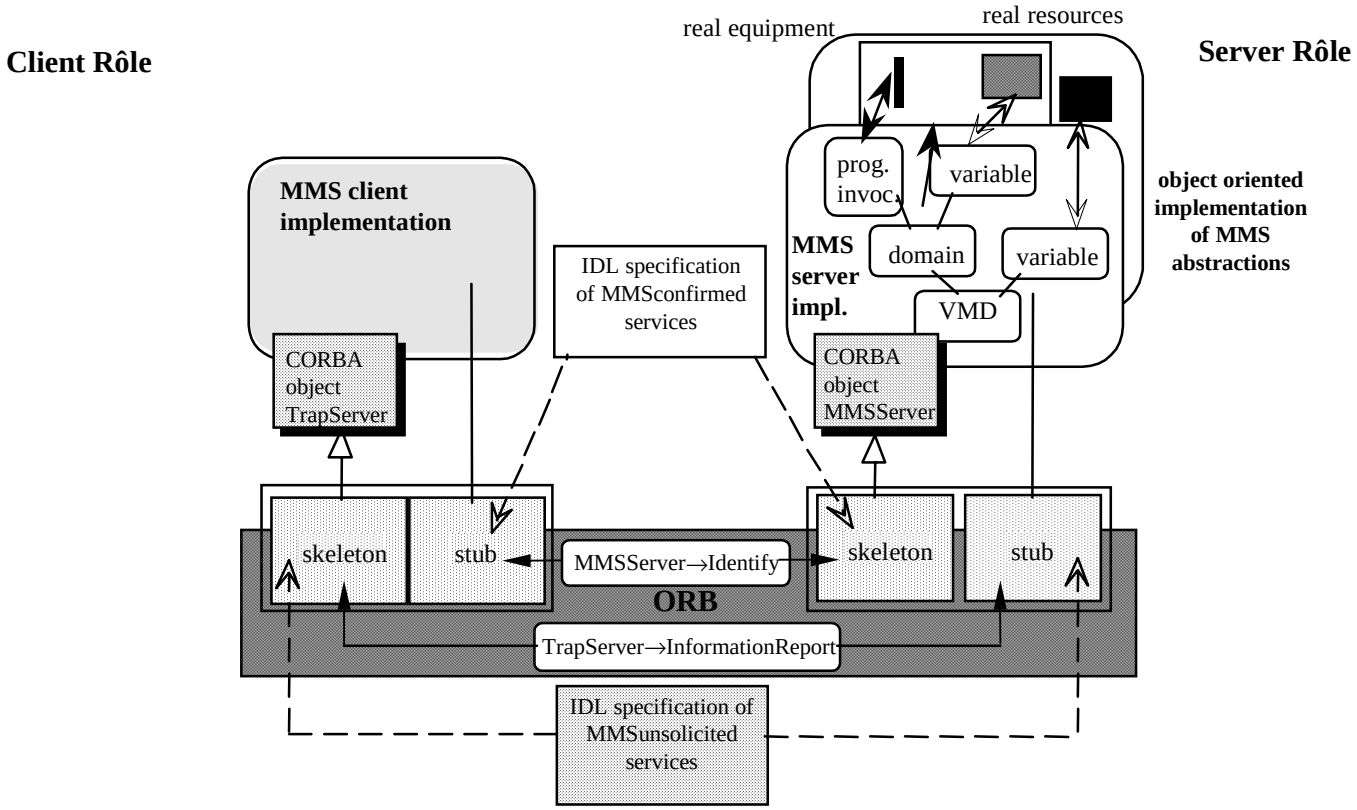
COS Services :

- Replace MMS objects : Events(MMS-Events), Persistence(Domains Download/Upload), Concurrency Control(Semaphores)
- Add Value : Security, Time Synchronization, Transaction

CORBA 3.0 :

- RT-CORBA
- Minimum CORBA
- CCM

Objectified MMS architecture model



VMD = CORBA object, other MMS abstractions implemented as programming objects

IDL specifications use MMS services and PDUs, method invocations are synchronous

(request PDU => parameters , response PDU => results , error PDU => exceptions)

Principles of the IDL specification for implemented services

Example of the Identify service :

```
interface serveurMMS {
    …
    IdentifyResponse Identify() raises (ServiceError);
    …
    };
```

Service request without parameters → No call parameters

Identify-Response ::= SEQUENCE {
VendorName [0] IMPLICIT VisibleString,
modelName [1] IMPLICIT VisibleString,
revision [2] IMPLICIT VisibleString
}

→

```
struct IdentifyResponse {
    string vendorName;
    string modelName;
    string revision;
    };
```

Error PDU →

```
exception ServiceError {
ServiceErrorType serviceErrorVal;
};
```

```
ServiceError ::= SEQUENCE {
 errorClass [0] CHOICE {
        vmd-state       [0] IMPLICIT INTEGER
         {
         other                          (0),
         vmd-state-conflict             (1),
         vmd-operational-problem        (2),
         domain-transfer-problem        (3),
         state-machine-id-invalid       (4)
         },
        application-reference[1] IMPLICIT INTEGER
        {
         other                          (0),
         application-unreachable        (1),
         connection-lost                (2),
         application-reference-invalid  (3),
         context-unsupported            (4)
         },
        …
 }
}
```

```
enum CategorieVmdState{
 otherVmdState,
 vmdStateConflict,
 vmdOperationalProblem,
 domainTransferProblem,
 stateMachineIdInvalid
};

enum CategorieApplicationReference{
…
 otherApplicationReference,
 applicationUnreachable,
 connectionLost,
 applicationReferenceInvalid,
 contextUnsupported
};

enum TypeError {
 vmdState,
 applicationReference,
 …
};

union ServiceErrorType switch(TypeError) {
 case vmdState : CategorieVmdState Valeur_vmdState;
 case applicationReference : CategorieApplicationReference  Valeur_applicationReference;
 …
};
```

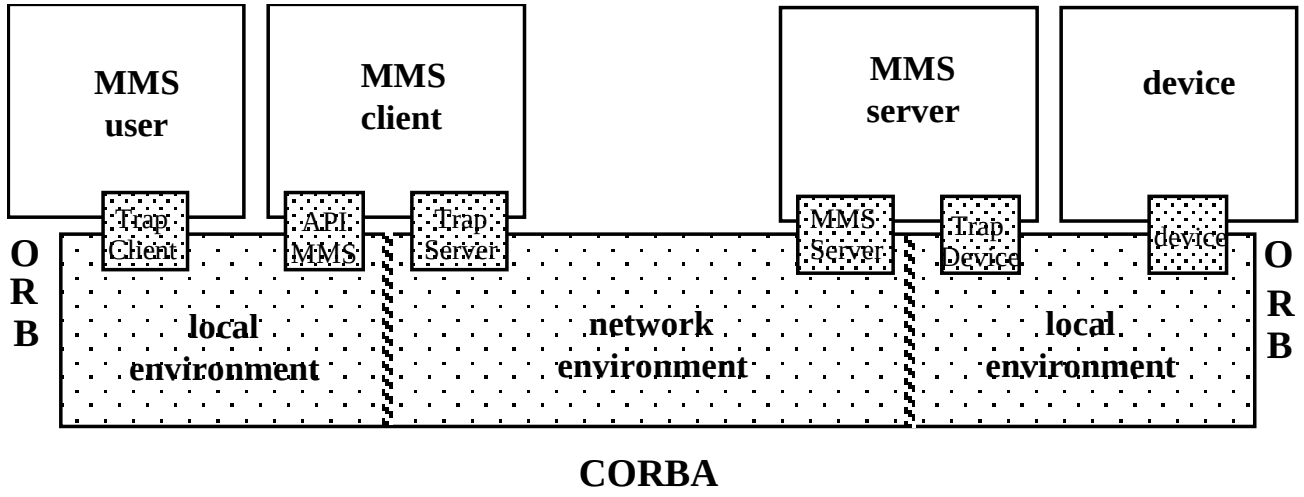
Losses from the ISO standard

- Association vanishes : association scope MMS objects, association parameters negociation, PSAP addressing disappear
  in CORBA : the ORB handles implicitly the relationship between client and server, the VMD IOR becomes its name and gains location transparency

- MMS Requests are concurrent and asynchronous, the ORB solution is synchronous and needs multi-threading support in server and client sides

- Unconfirmed services from the server to the client (*UnsollicitedStatus* and *InformationReport*) are mapped on synchronous methods without result parameters, the MMS server should be a CORBA client and the MMS client should be a CORBA server too

CORBA-MMS implementation

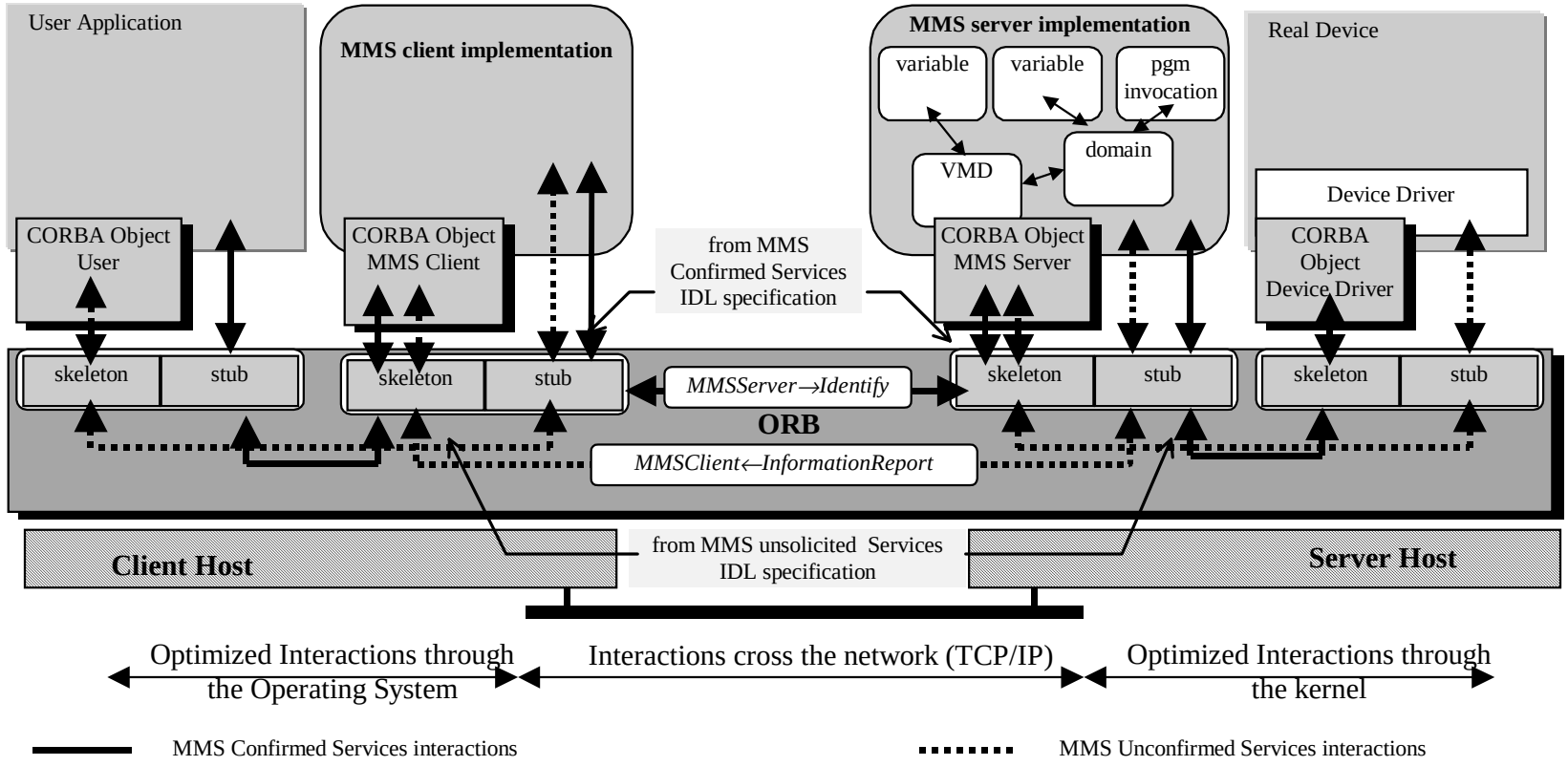


First prototype written in C++ over the ChorusOS micro-kernel and its ORB COOL (97)

Uses :

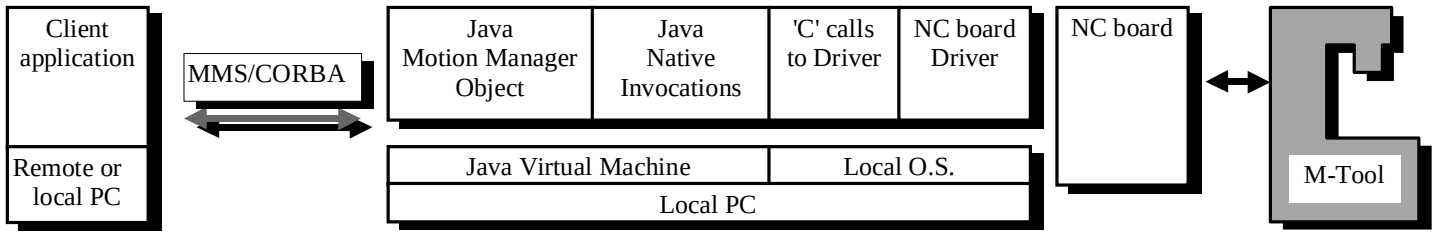
- Invocations over Network : TCP/IP (IIOP) or Distributed IPC directly over Ethernet or VME bus
- Local Invocations : no network protocols

Objectified mobile MMS implementation



- JAVA provides code mobility : the MMS client and the MMS server are implemented in JAVA and then can migrate from architecture to architecture independantly
- This architecture runs under Linux and Windows 98, using ORBacus, and Jonathan ORBs

COCA Project (end 98)



COCA PROJECT (with IUT St Denis - GRPI) :
Architecture of Java objectified MMS demonstrator.

The device is a NC, driven through Java Native Interface (experimented with Linux)

A framework to address Real Time Constraints in Communications

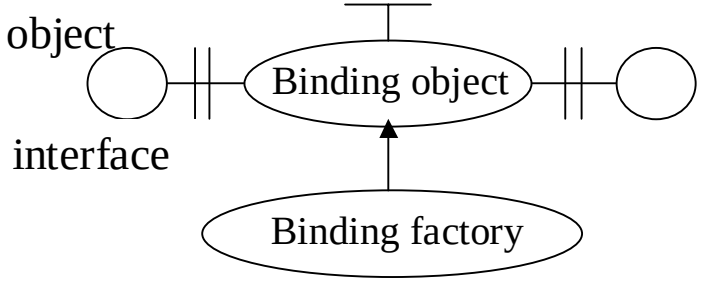
Goal : deterministic behaviour[and deterministic execution time]

- First step : a QoS aware network for an Objectified MMS
- Requirement : reuse of the existing Java/CORBA MMS prototype
- Solution : extension of the Jonathan flexible-ORB framework for ATM networks (allows pluggable transport protocols through binding objects)

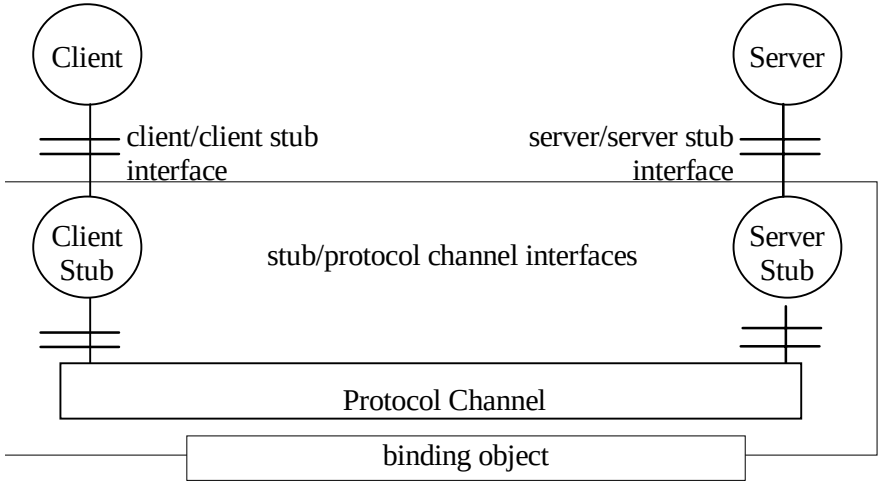
Model of distributed systems with real-time constraints

RM-ODP extented with the ReTINA proposal :

- Computational Point of view



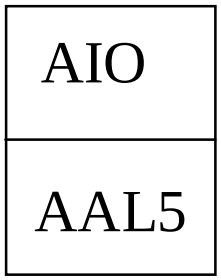
- Engineering Point of view



implemented within Jonathan, a flexible micro-ORB from CNET

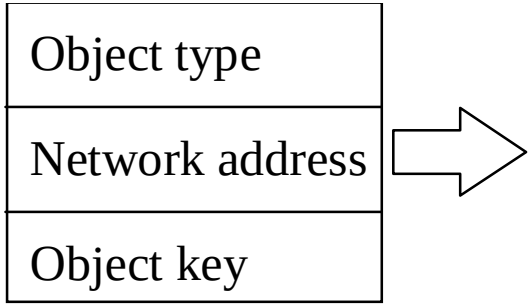
AJAX (Atm for Jonathan And linuX in 99)

AJAX protocol stack :



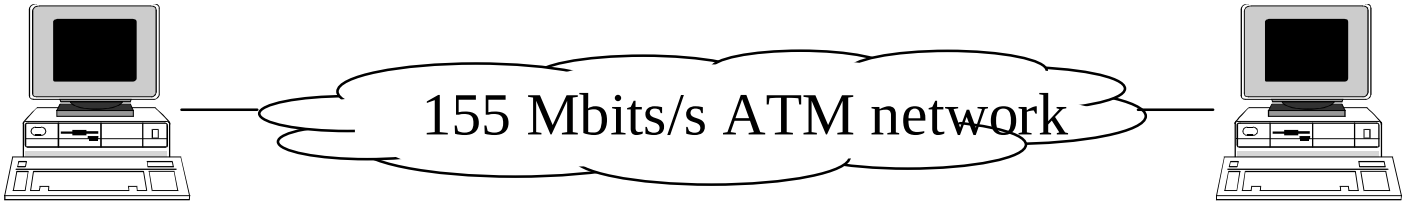
AIOP carries CORBA method invocations
(≡ IIOP in the TCP world)
1. handles method calls and their returns
2. fragmentation/defrag. of large messages
3. detect comm. errors & cell losses

AJAX object references :



20 bytes ATM NSAP address
(≡ IP add. + port numb. in the TCP world)

Experiments



200 Mhz PC
Linux 2.1.x kernel
JDK 1.1.x

200 Mhz PC
Linux 2.1.x kernel
JDK 1.1.x

10 concurrent binding objects with PCR=10 Mbits/s between MMS objects

Current limitations

Cell losses for throughput > 15 Mbits/s

Cause : the JDK JVM is not enough efficient

- overhead in program exec.
- multiple copies of data between several memory buffer areas

Work to be done !

We achieved a first step toward a Real-Time Objectified MMS

To fullfill our requirements we need more :

- A **real-time executive**
- A **real-time Java Virtual Machine** with an efficient buffer management (0-copy) (**Lizzi's results** : Java threads on two different hosts (no JIT) delivers a throughput of 97 Mbps, an end-to-end delay of 15 ms (1 switch crossed), with a loss ratio of 0.3 % (due to buffering losses, no cell losses).)
- **COTS based solutions**

Conclusion

Is MMS dead ?

- Few implementations of MMS (too complex, expensive)
- Concepts are not ! Still alive in fielbuses …
- The standard TASE.2 uses MMS as an underlying protocol
- Industrial Messaging Services needed in home networks

Which future for OO-QoS aware-mobile-MMS ?

- All required buzz words ;-)
- QoS vs CoS approaches ? COTS will influence the choice