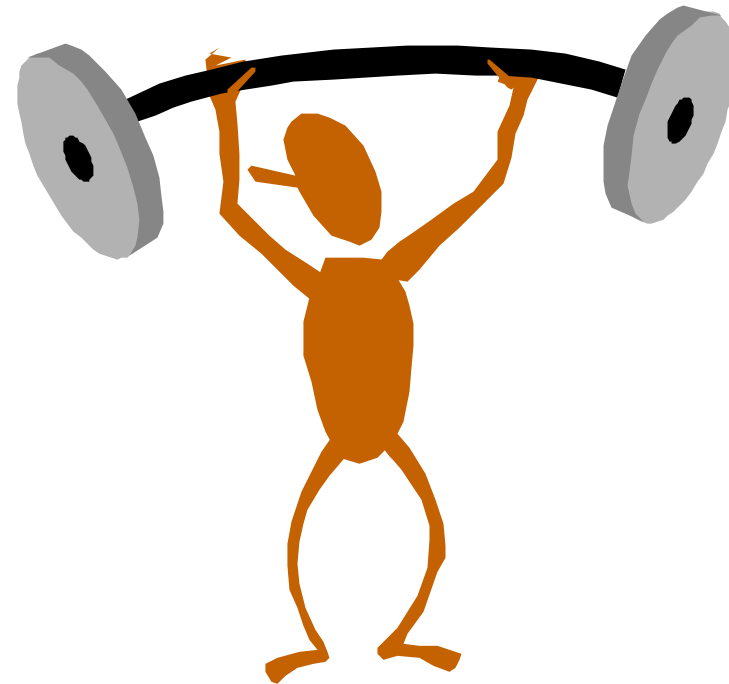# Flexing FlexiNet's Muscles
adding support for IIOP

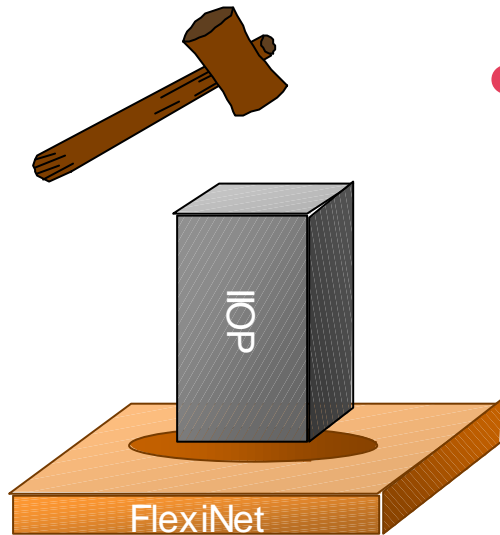Matthew Faupel

8 April 1998

Motivation

- Provide a range of protocols for negotiation

- Allow interworking with other systems

- Leverage existing CORBA services

- Validate the FlexiNet modular structure

How well does IIOP fit in?

IIOP

FlexiNet

- IIOP would seem to have a poor fit with the FlexiNet architecture, as:

  - FlexiNet has no IDL to resolve multiple Java-IDL type mappings

  - IIOP wire format is not well structured for incremental layering
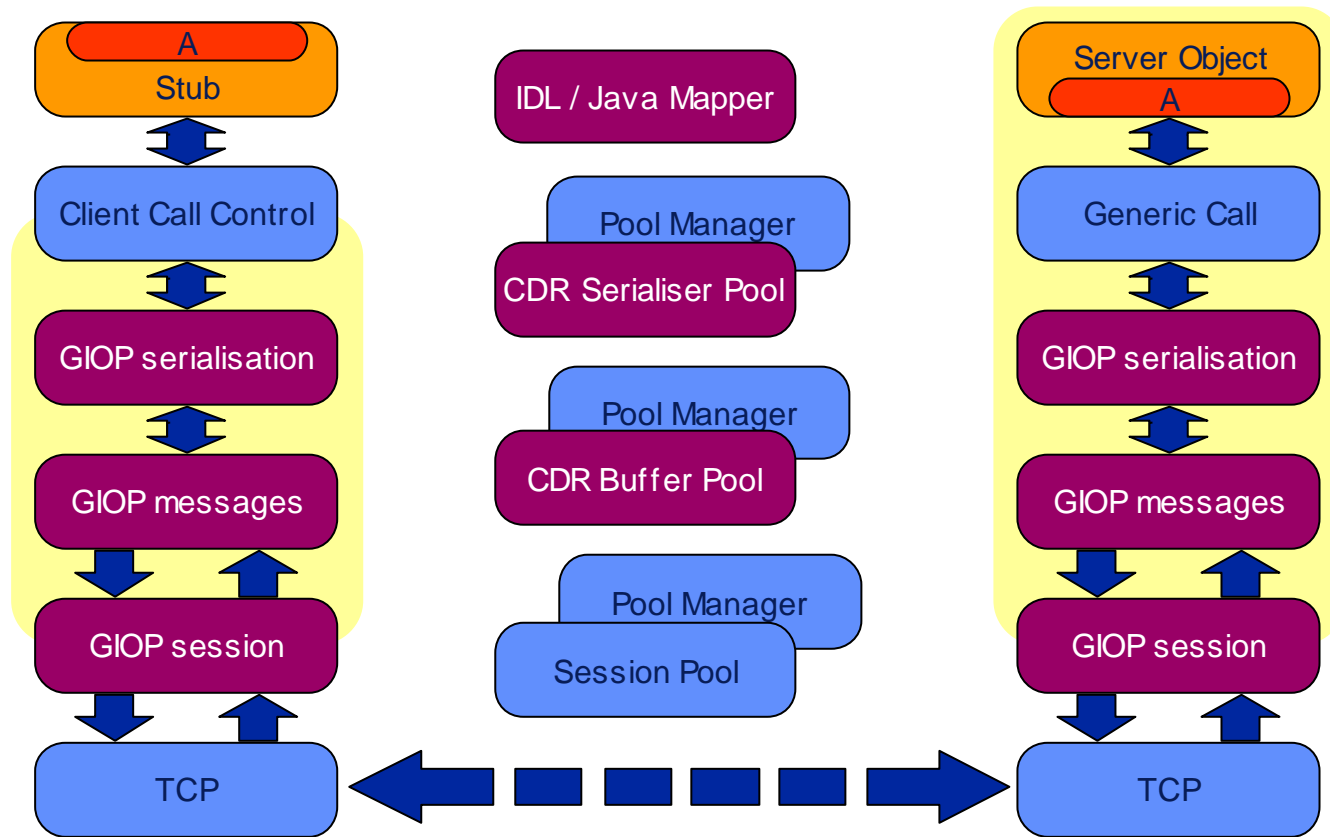
  - IIOP has no concept of sessions

Making it fit

- Use reflection to examine types and signatures

- Represent mappings as classes
  - allow alternative mappings on a per-binding basis

- Protocol can be layered to some extent...
  - …but there is a little "leakage" between layers

- Message ids can double as session ids

Putting it together

A **Stub**	IDL / Java Mapper	Server Object **A**
Client Call Control	Pool Manager / CDR Serialiser Pool	Generic Call
GIOP serialisation		GIOP serialisation
GIOP messages	Pool Manager / CDR Buffer Pool	GIOP messages
GIOP session	Pool Manager / Session Pool	GIOP session
TCP		TCP

What does it do?

- Full GIOP 1.1/IIOP 1.1
- Full support for CORBA2.2 IDL/Java mapping
  - except unions
- Support for objects by value RFP
- Multiple possible Java/IDL mappings (two provided)
- Transparent to application programmer whether using REX/UDP or IIOP/TCP
- Piggy-back FlexiNet names

What doesn't it do?

- unions

- code set negotiation

- CORBA meta-information support (interface repository, _get_interface calls etc.)

- POA (i.e. CORBA application portability)

What was difficult?

- Most problems were with the CORBA side
  - Java->IDL spec not easily reversible
  - important information buried in awkward places in the wire format
- Fitting into FlexiNet relatively easy
  - main changes were to make some classes interfaces, e.g. Buffers, Serializers, Names, SessionLayer

Lessons learnt for FlexiNet

- We needed to organise our code more logically

- Key abstractions should always be interfaces

- Package protection should not be relied upon

- The basic FlexiNet architecture is well founded
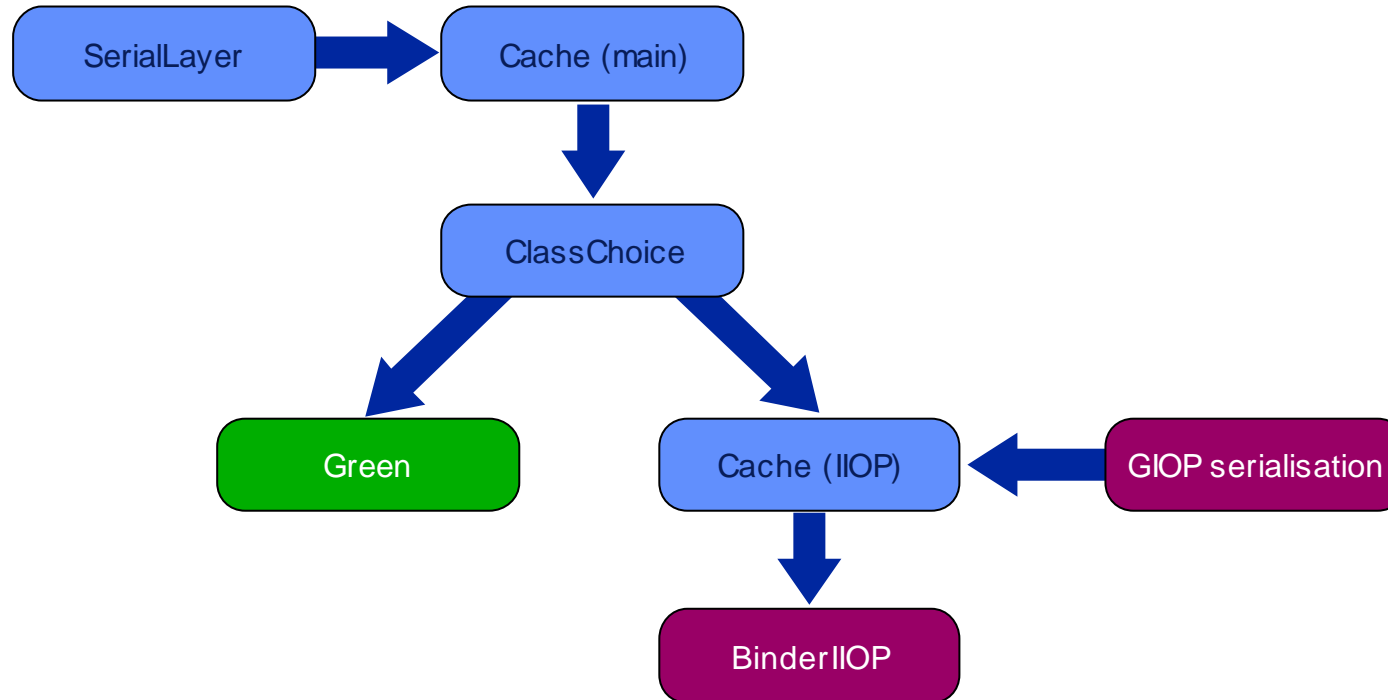
Mappers

- Basic mapper
  - for use when contacting existing CORBA servers
  - characters and strings treated as narrow
  - classes encoded as structs

- Object by value mapper
  - for use as a full feature transport for FlexiNet
  - characters and strings treated as wide
  - classes and strings encoded as values

Using the IIOP binder

Further possibilities

- **Integrating other FlexiNet concepts into CORBA worldview e.g. mobility, transactions, security**

- **Investigate alternative approach to marshalling of "intelligent" buffers**

- **Interface repository**

  - which would also allow a more efficient and universal mapper

- **Optimisation**

- **POA**