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

Trader Presentation to OMG

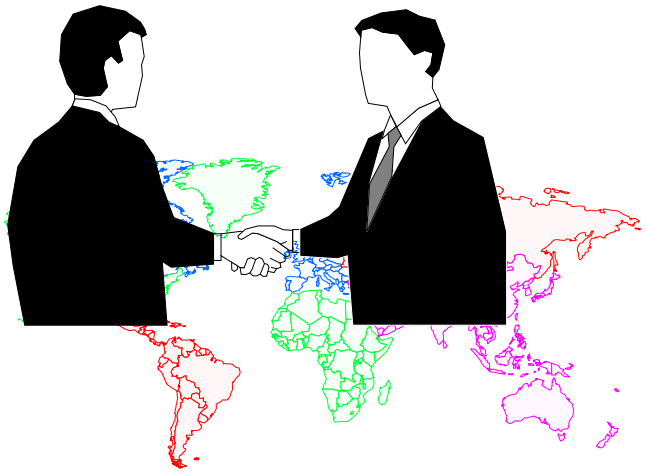
Andrew Herbert

Abstract

Presentation of Trader concepts to OMG Object Services Task Force meeting, Cambridge, England, March 1995.

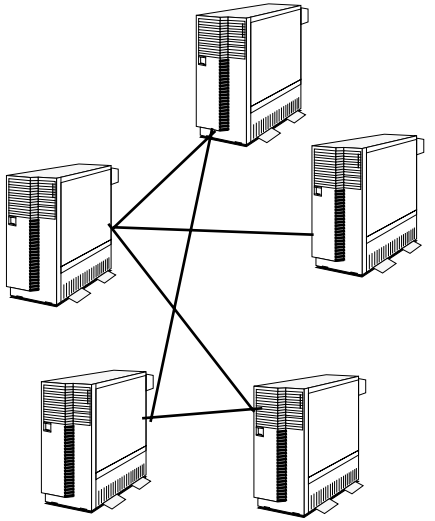
APM.1447.01 **Approved** 23rd March 1995
External Paper

Distribution:
Supersedes:
Superseded by:



TRADING

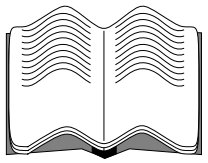
AN ANSA PERSPECTIVE



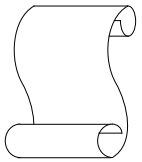
Andrew Herbert

Technical Director, APM

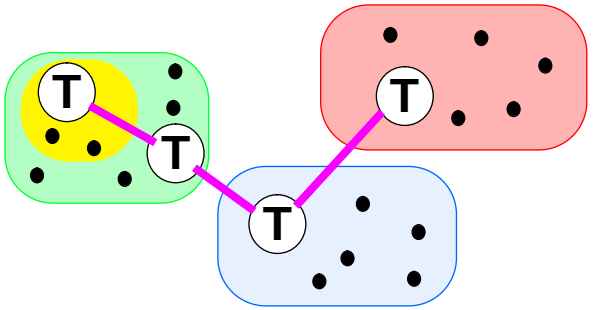
What is trading?



A *"Yellow Pages"* or *"Discovery"* function

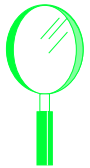


Which finds objects providing a service described in terms of both its type and its properties



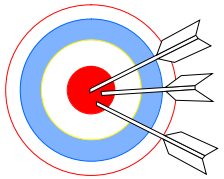
And enables domain based management of services

Trading is an Object Service, not an ORB!



Trading finds a suitable object instance.
Trading fails if there is no such object.
Trading yields an *object pointer*

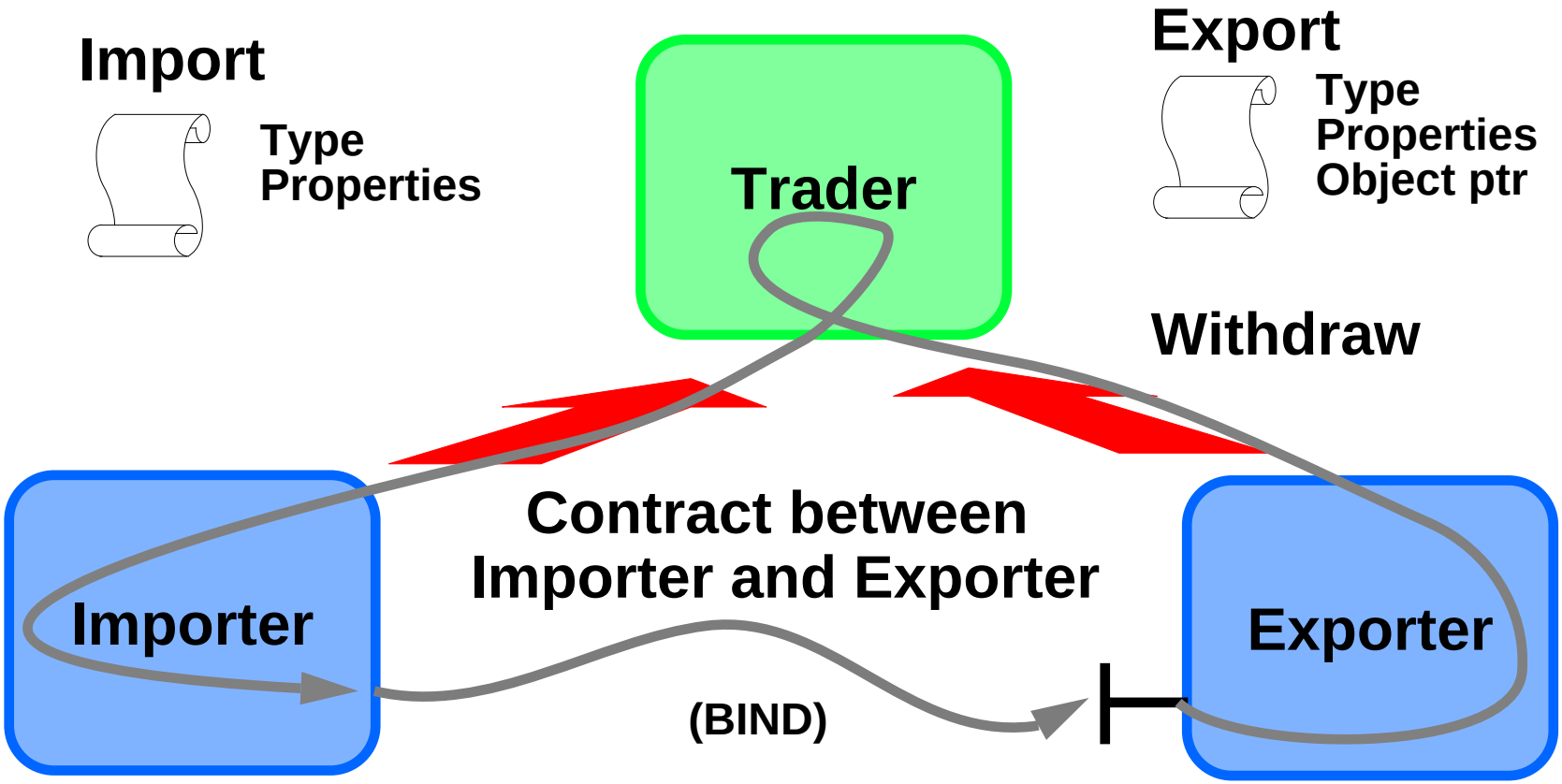
ORBs *bind* object pointers; binding only fails if the target object or the network fails



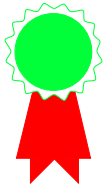
***Rebinding* gets you to the *same* object;**
***re-trading* might get you to another object**

Anyone can trade by passing object pointer parameters - traders are useful, not essential!

Key Trading Operations



Type Matching



Type safety is a good thing
Compilers type check *objects*
Traders type check *systems*

Minimum useful check is interface signature compatibility

Signatures are available from the *Interface Repository*

Any other checks should allow servers and clients to evolve separately

Trading and Properties

Signatures are not sufficient to choose a service

Who owns the service?

What will it cost to use?

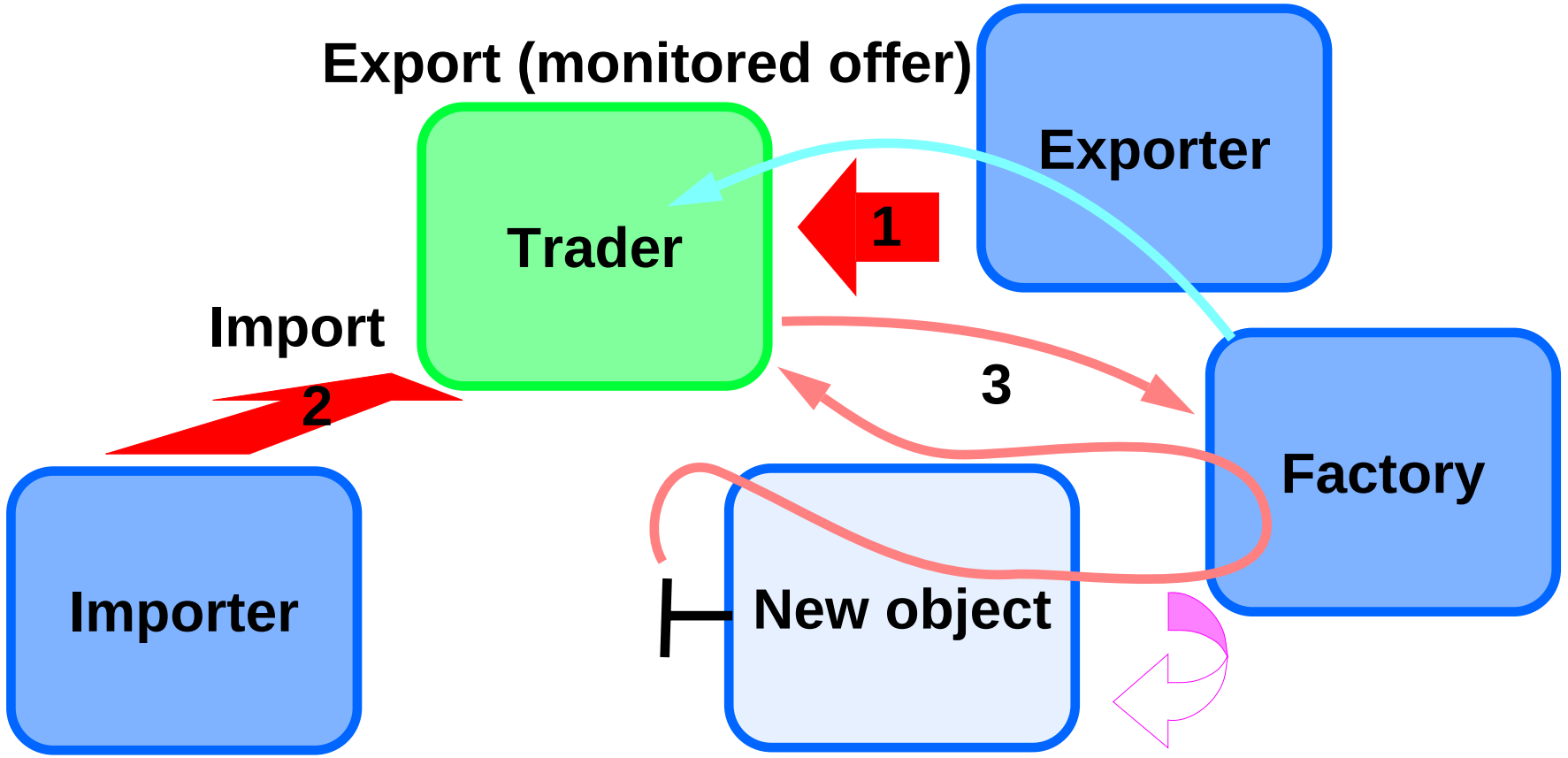
Where is the service?

Is it high quality?

Can I bind (bridge) to it?

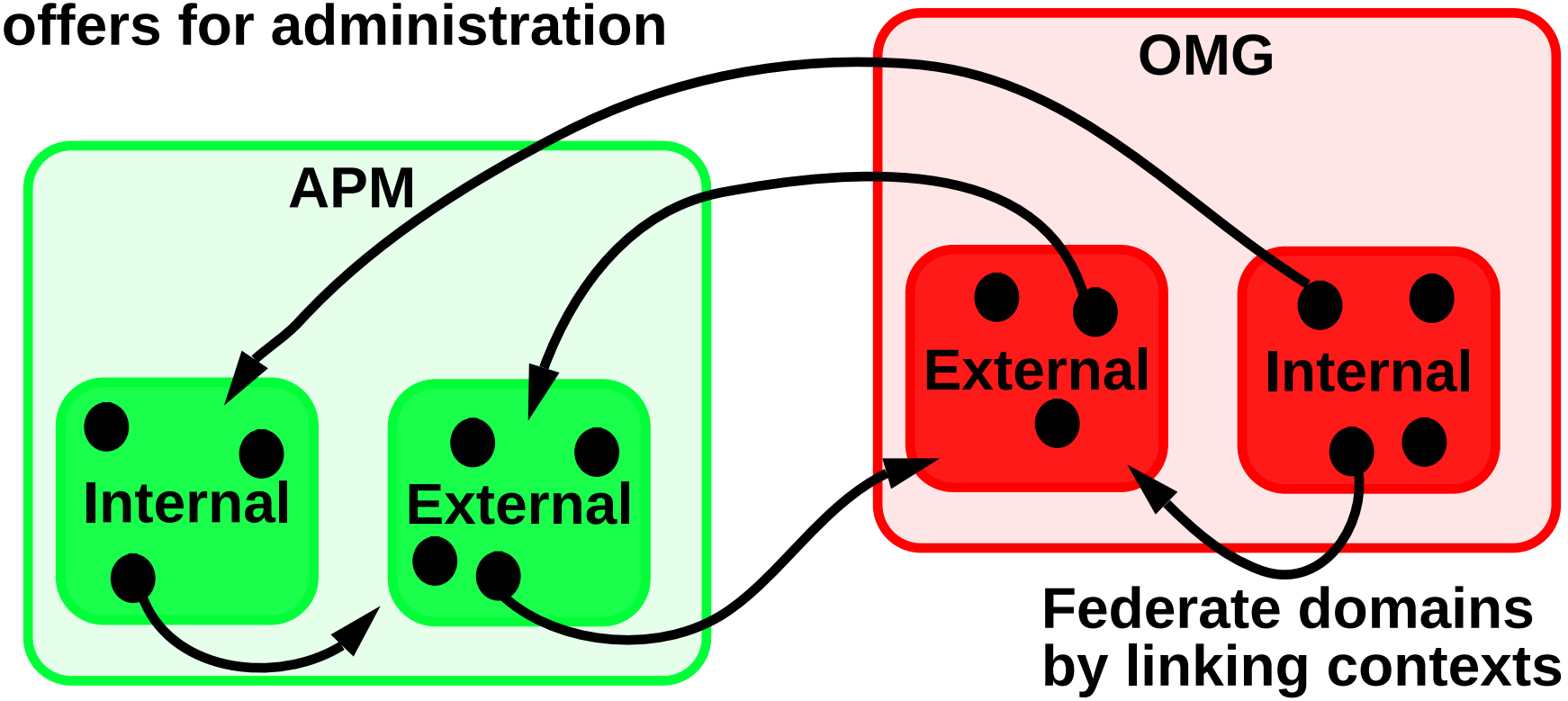
Qualify import with desired properties.

Trading and Life Cycle



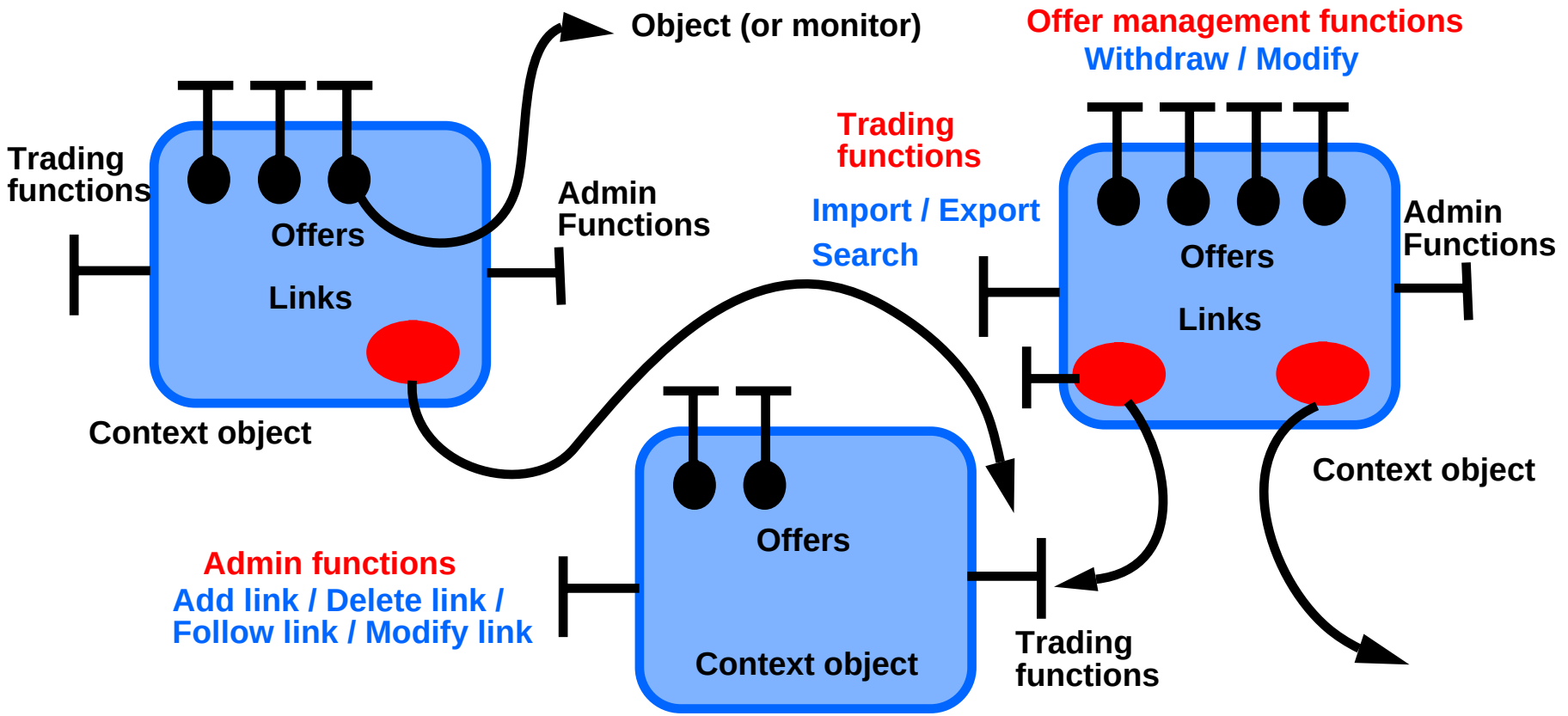
Trader Administration

Use *contexts* to partition offers for administration



Federate domains by linking contexts

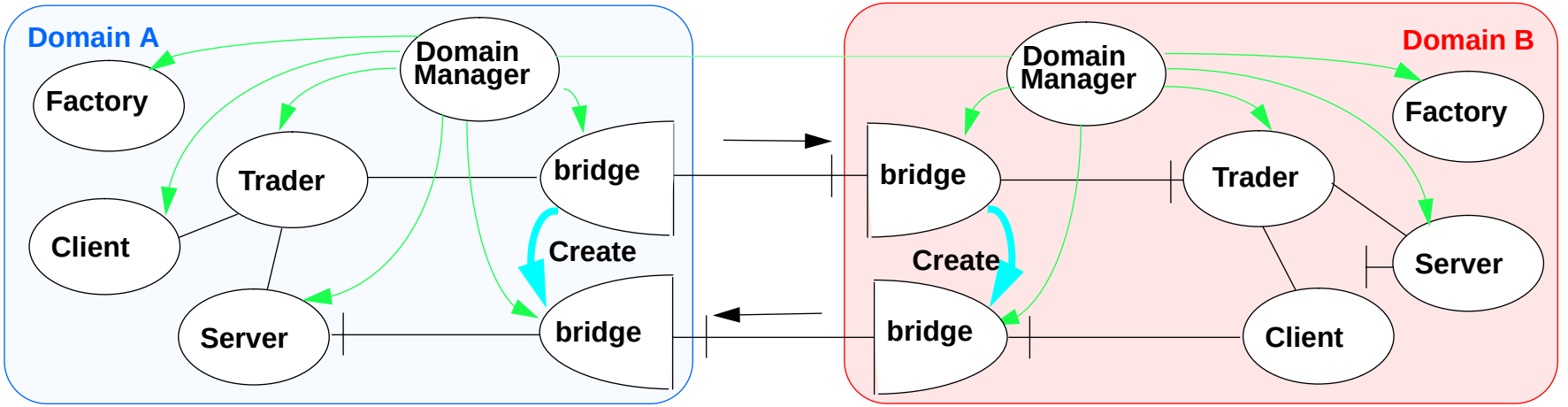
System Model



Trader Implementation

- **Self-contained object**
  - optimized for locality and speed
  - federated through implementing the system model
  - enables explicit navigation
- **Using a name service**
  - trader becomes local agent of distributed name service
  - exploit distribution of name service to handle federation
  - transparency prevents explicit navigation
  - lazy update can give inconsistent views
- **using a database**
  - deal with large numbers of offers, sophisticated querying facilities
  - linked to other repositories

Trading and UNO Bridging



As object references cross bridges

- **Detect**
- **Recognize**
- **Mark**
- **Resolve**
  - leave
  - translate
  - create gateway

Practical Traders

- ICL DAIS Trader, BNR ODS Trader
  - derived from ANSAware Trader by Joe Sventek of HP
  - in memory, explicit type name graph
- Bellcore INASoft Trader
  - Trades in a DCE and CMIS/CMIP environment
- ANSA Orbix Property Repository
  - CORBA based, uses a relational database, available via WWW
- Research traders
  - Australia - federation, integration with other “repositories”
  - Germany - trading as focus for resource management

ODP Trader Standard

- At CD status
  - ballot comments to be processed in May
  - editing being lead by Tom Rutt of ATT
- Enterprise specification -- who uses trading to do what?
  - major roles and responsibilities
  - identification of scope of policies
- Information specification -- definition of concepts and processes
  - information model for offers, properties, contexts, etc
- Computational specification -- the service interfaces
  - interfaces in CORBA IDL
- Enginering specifications -- implementation strategies
  - mapping onto X.500

Issues for OMG Trading Service

- Which interfaces to define -- user, administration, federation?
- Define standard property sets?
- Define property selection language, c.f., life cycle services?
- Relationship to naming service, query service, interface repository?
- Select an implementation strategy?
- Links to resource management -- e.g., “stale offers”?