

MLS Interoperability

using CORBA

Myong H. Kang
Naval Research Laboratory

CORBA

- promises interoperability among applications
 - ◆ written in many different languages
 - ◆ running on multiple operating systems
 - ◆ using a variety of networking protocols
- can be a
 - ◆ set of libraries
 - ◆ set of daemon processes
 - ◆ server machine
 - ◆ part of an operating system

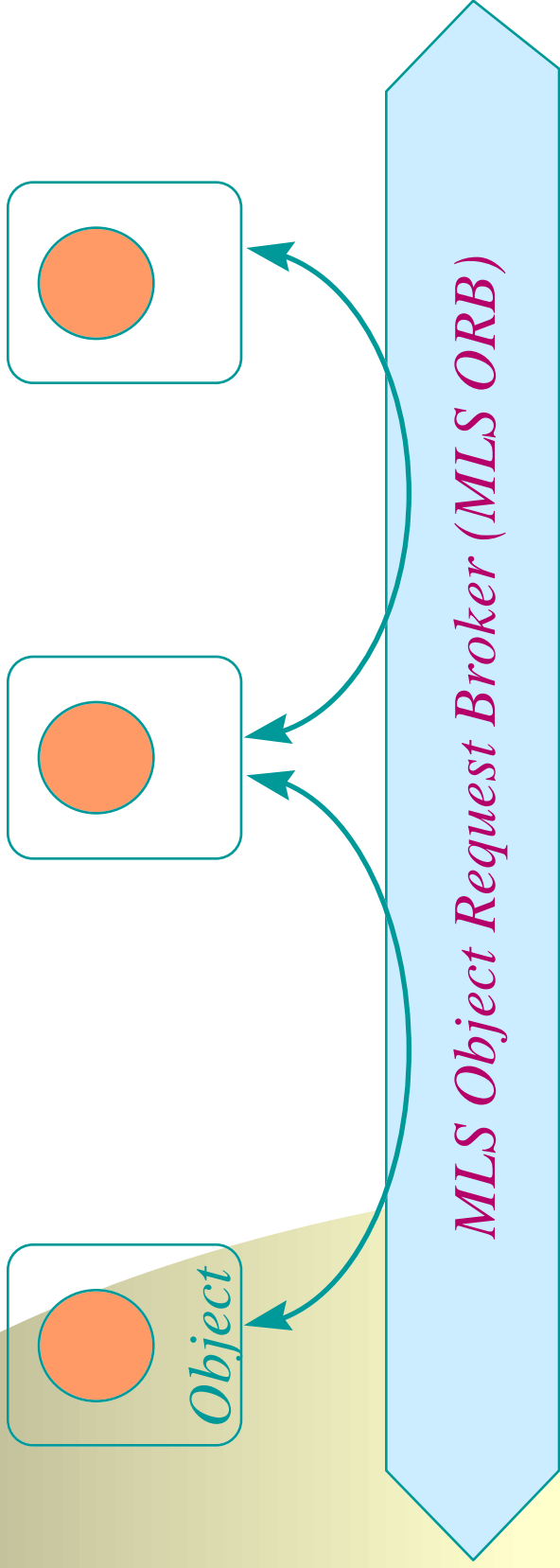
Multilevel Security

- MLS systems should guard against
 - ◆ untrusted applications
 - ◆ unintentional user mistakes
 - ◆
- MLS systems depend on
 - ◆ physical separation
 - ◆ logical separation (e.g., OS)
 - ◆

What is MLS ORB?

MLS ORB allows trusted objects to communicate to other objects at different security levels

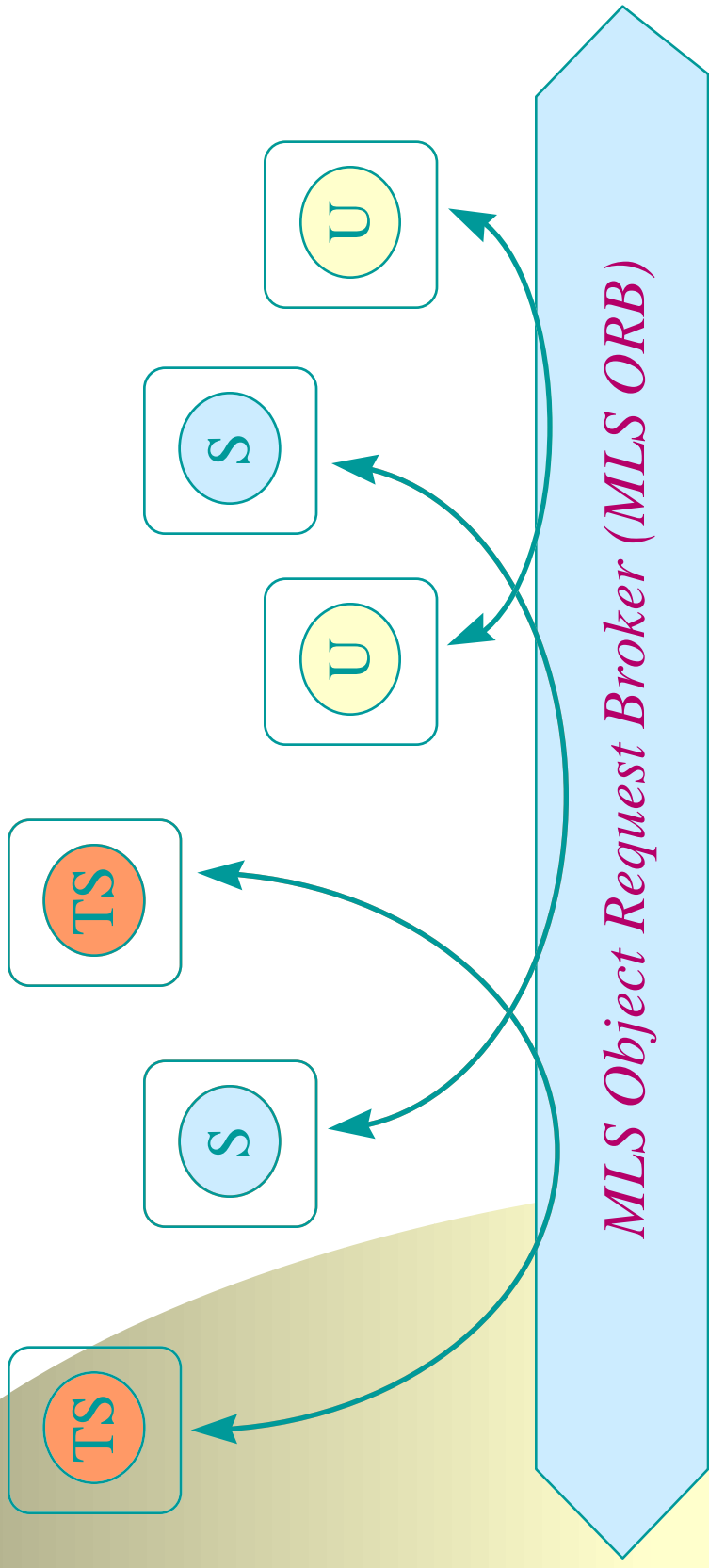
Producer (TS) Consumer / Producer (S) Consumer (U)



All objects that directly or indirectly communicate to objects at lower levels must be trusted !

What is MLS ORB ? (cont)

MLS ORB allows objects to communicate to other objects at the same security level



No information sharing among objects at different security levels

Requirements for High-assurance MLS ORB

- MLS ORB must be non-by-passable
 - ◆ (persistent) objects should not be modified by unauthorized users or unauthorized mechanisms
 - ◆ Without help from underlying OSs, it is difficult to enforce this property
- An MLS ORB should communicate to other MLS ORBs
 - ◆ composition of different assurance classes (vulnerability of the whole system is determined by the most vulnerable part)

Is high-assurance MLS ORB practical ?

- Customer basis for high-assurance MLS ORB is limited
- Enforcing non-by-passable property is difficult because not many high-assurance OSs exist and are deployed
- High-assurance MLS ORB idea has the same weakness as building distributed MLS systems
 - ◆ expensive and difficult to build applications (objects) --- trusted objects
 - ◆ assurance level of the whole system is no greater than that of the lowest assurance ORB

MLS revisited

- Users

- ◆ users at different security levels should be physically separated

- Information

- ◆ lower level information can be located in higher level systems but higher level information should not be located at lower level systems
- ◆ Computing resources
- ◆ either logically or physically separated
- ◆ MLS systems should be managed by users of highest level information on the system

We propose low-assurance MLS ORB

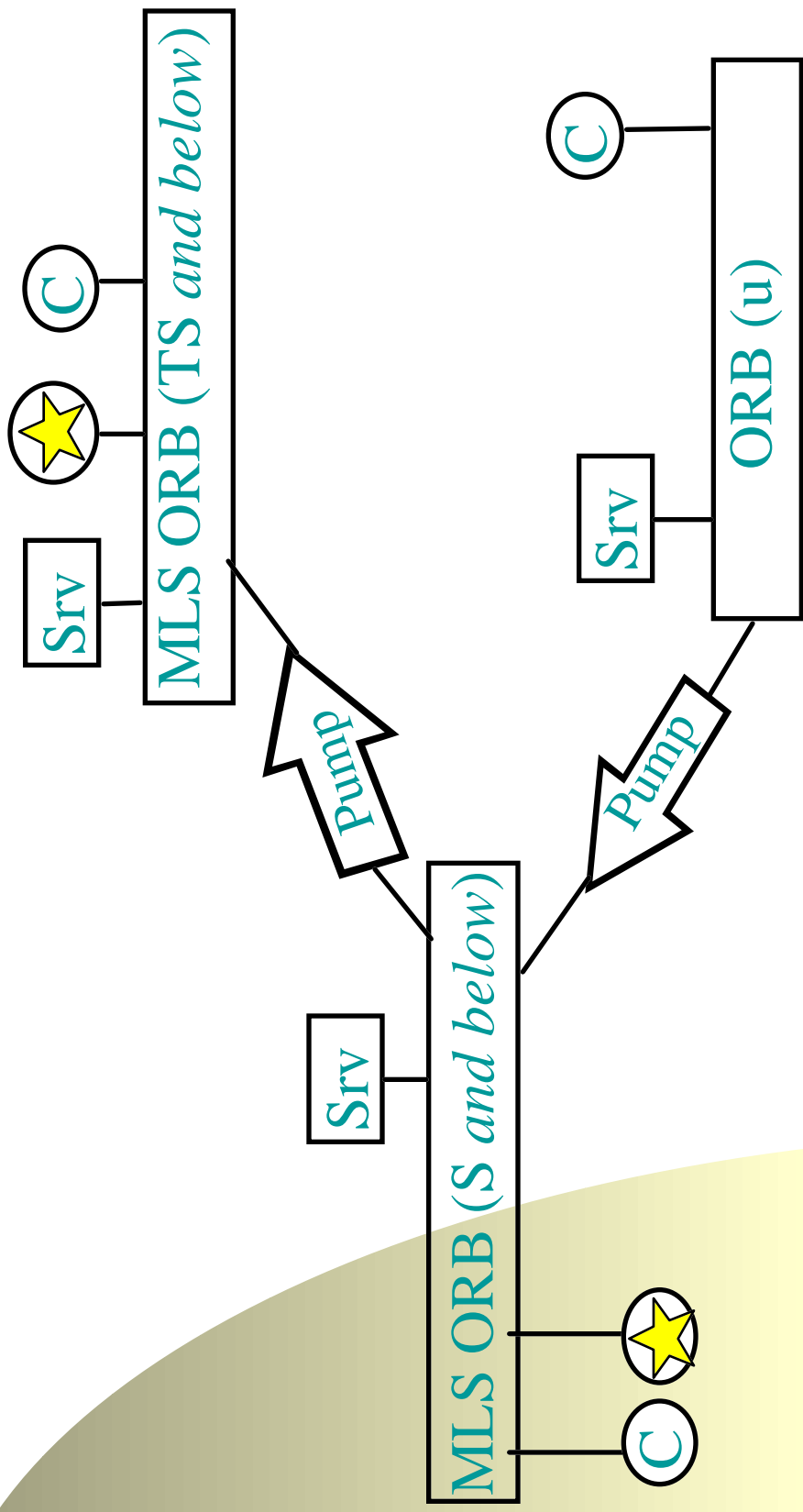
- Provide MLS functionality with moderate assurance
 - ◆ security labels and access control
 - ◆ I&A and non-repudiation
 - ◆ confidentiality and integrity of messages
 - ◆ enforce flexible security policies (different ORBs may enforce different security policies)

Potential customers of low-assurance MLS ORB

- Financial institutions
 - ◆ only authorized users can access the system (object)
- Commercial organization
 - ◆ insiders and outsiders
- Government customers
 - ◆ many security levels and compartments

***Propose a high-assurance
distributed architecture with
moderate-assurance ORBs and
high-assurance security devices!***

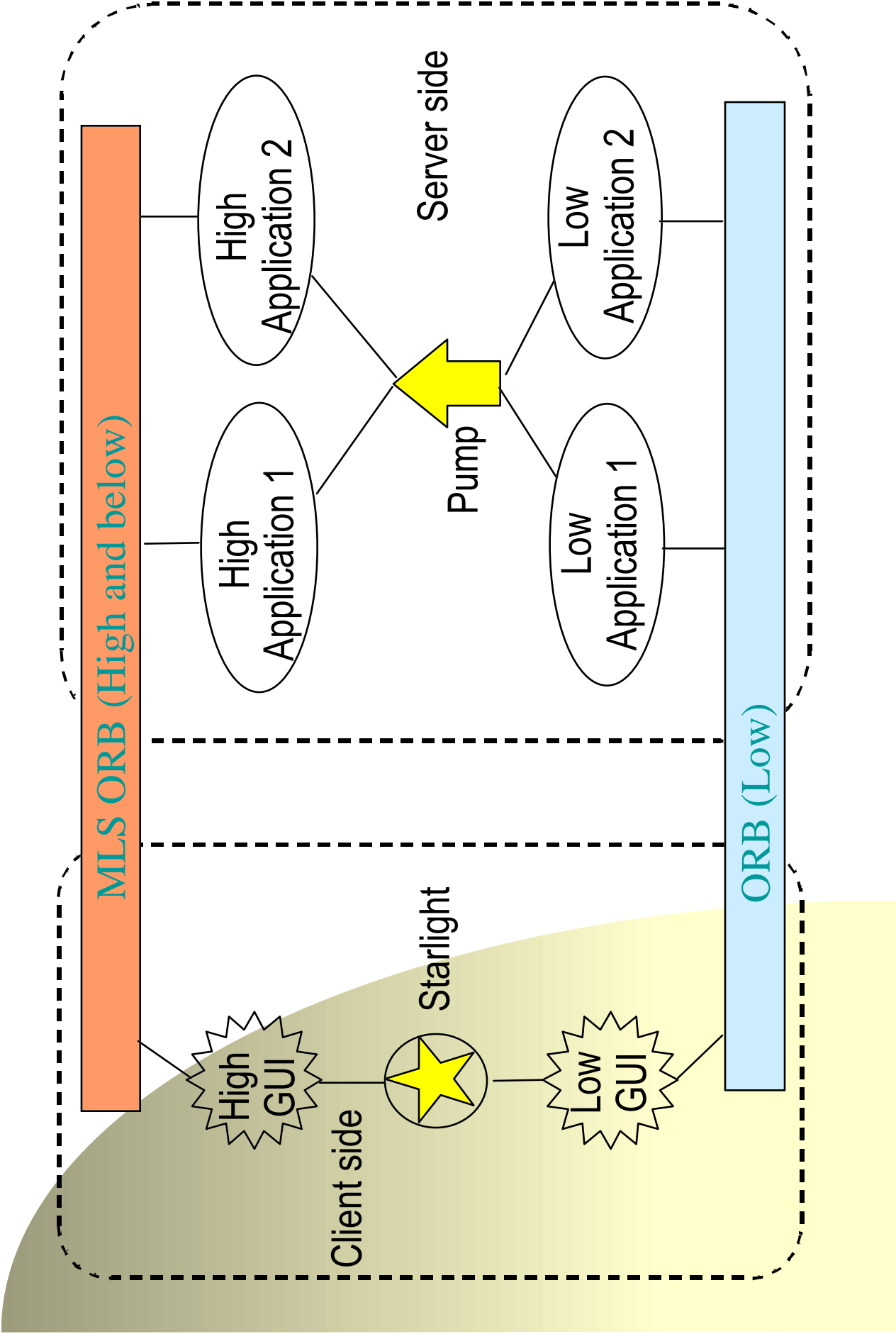
Architecture-based Security



C : single-level client

Starlight : Starlight (multilevel workstation)

Srv : Server

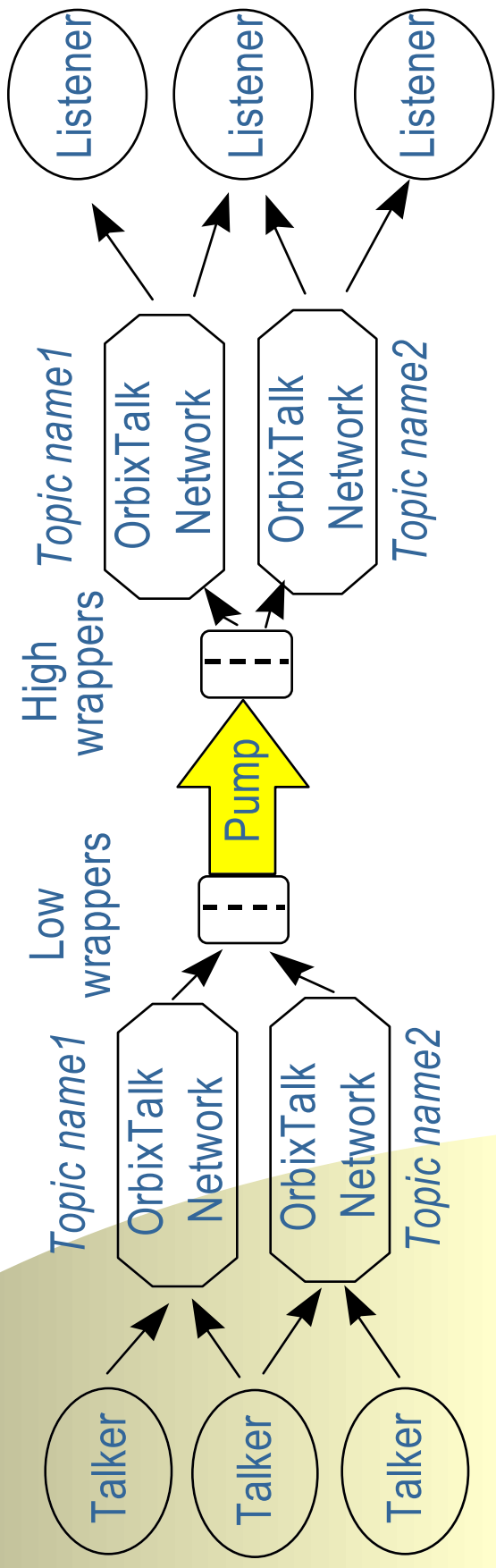


Distributed MLS Paradigm

- High level users can access lower level information and resources through multilevel workstations (e.g., starlight)
 - ◆ ad-hoc access and requests
- High applications access lower level information through replication of lower level information to high level systems through secure one-way devices (e.g., NRL Pump)
- MLS business processes through cooperative applications (objects)
 - ◆ reduce ad-hoc access and requests
- Downgrading can be done through specialized downgraders

Example:

Supporting MLS event-driven programming and multicasting



Advantages of Architecture-based Security

- Isolate security enforcement into simple mechanisms
 - ◆ easier to build high-assurance systems
 - ◆ ability to use existing security mechanisms
 - ◆ ability to develop operational solutions independent of security solutions
- Use of unmodified COTS CORBA-compliant client and servers
 - ◆ facilitate migration to new standards, and exploitation of technology advances
 - ◆ reduce training, maintenance and system cost

Summary

- Focus on security functionality with moderate-assurance
 - ◆ security labels and access control
 - ◆ I&A and non-repudiation
 - ◆ confidentiality and integrity of messages
 - ◆ supporting flexible security policies
- *Investigate high-assurance distributed architectures with moderate-assurance ORBs and high-assurance security devices*
-