A Plug-in Transport with Dissimilar ORBs and a Connectionless Network

Nathan Scandella

OMG Embedded Object-Based Workshop

19 January, 2001

Santa Clara, CA

Honeywell

Washington
WASHINGTON · UNIVERSITY · IN · ST · LOUIS

BBN Technologies
a part of verizon

BOEING

Plug-in Transports for CORBA

- Introduction to DARPA/Air Force Project
- Design Motivators
  - system architecture
  - description of our transport
- CORBA Solution
- How to Plug In a Transport
- Lessons Learned
  - recommendations for RT CORBA

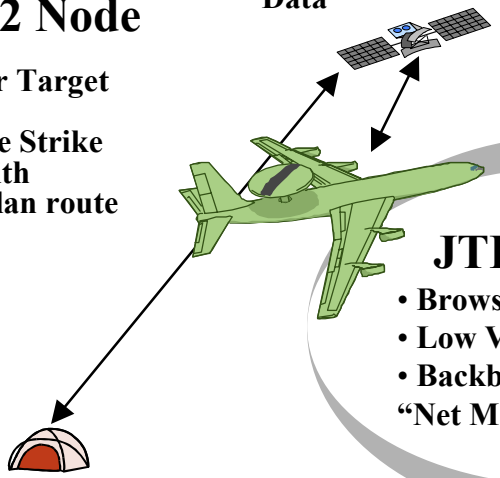
WSOA Project Concept

Commercial Satellite / GBS

- Broadcast of Time Critical Data

Airborne C2 Node

- Collaborates for Target Validation
- Retasks Enroute Strike
- Net Meeting with Warrior to replan route



JTIDS RF Net

- Browser Requests
- Low Volume Imagery
- Backbone for Tactical "Net Meeting"

Virtual Data Archive

- IPL Provides Target Imagery
- Generates Mission Preview
- Injects Time Critical Tactical Data into Broadcast

Warrior

- "Browser" Requests for Target and Imagery data
- Net Meeting with C2 Node for Target Review and Mission Replan
- "Browser" Requests for Mission Rehearsal
- Pulls Imagery and Rehearsal Data
- Previews Updated Mission Enroute

C2 = Command & Control (e.g. AWACS)

Honeywell

Washington University in St. Louis

BBN Technologies a part of Verizon

Boeing

WSOA Introduction

- Case Study - DARPA/Air Force sponsored flight demonstration project
- Weapons System Open Architecture (WSOA)
- Standards-based middleware desired
- Secure tactical datalink available between aircraft

punch line:

- TAO and ORBexpress real-time ORBs chosen
- Datalink transport plugged-in to ORBs

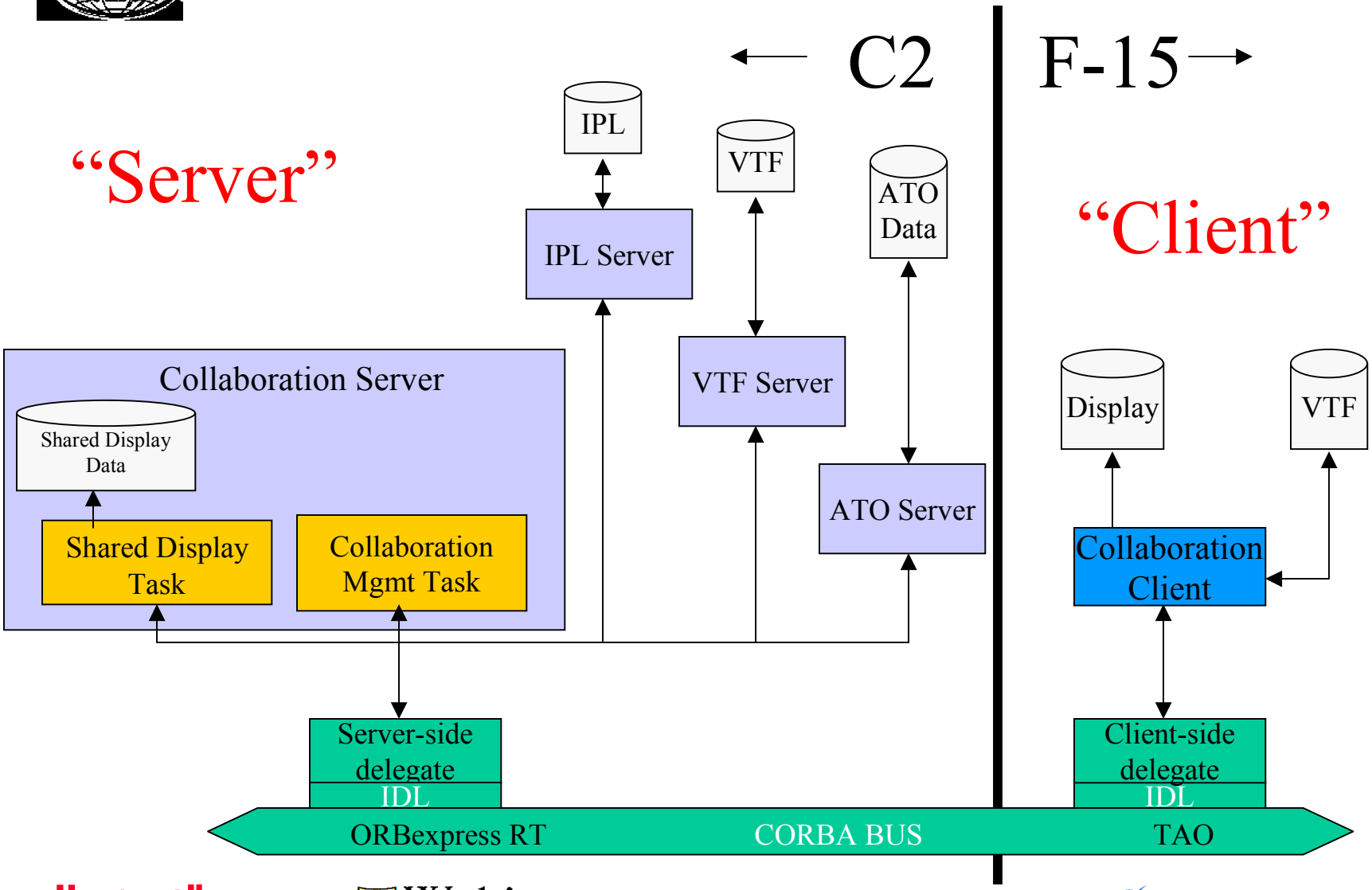
Honeywell

Washington
WASHINGTON·UNIVERSITY·IN·ST·LOUIS

BBN Technologies
a part of verizon

BOEING

WSOA Software Architecture



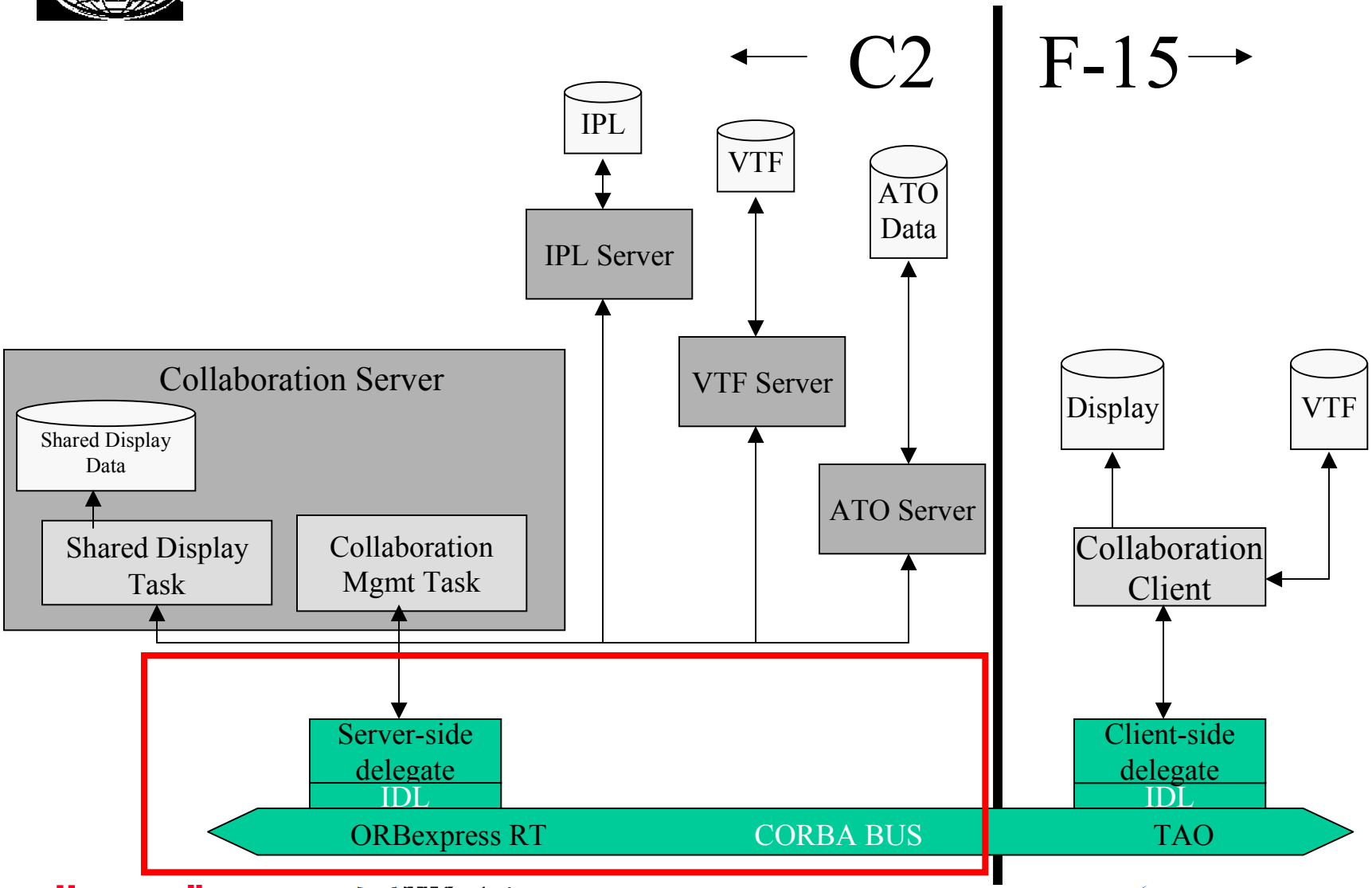
Honeywell

Washington University in St. Louis

BBN Technologies a part of Verizon

BOEING

WSOA Software Architecture



Honeywell

Washington University in St. Louis

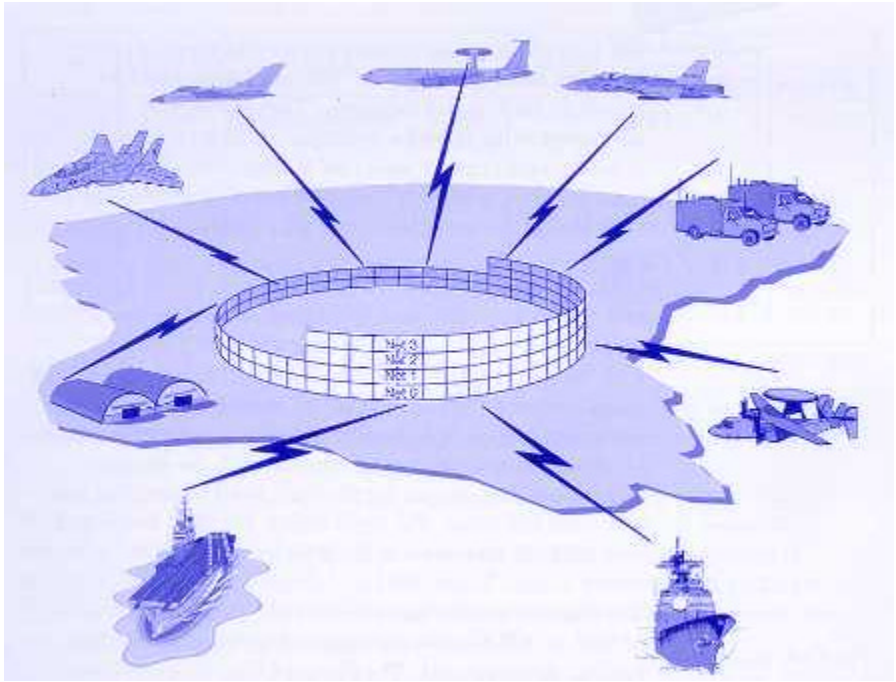
BBN Technologies a part of Verizon

BOEING

Link 16 Transport

Mil-Std-6016 Tactical Digital Information Link - Joint Tactical Information Distribution System (JTIDS)

- Time Division Multiple Access (TDMA)
- Nodeless
- Message and Transmission Encryption
- Frequency Hopping
- UHF, L_X Band
- Line of Sight



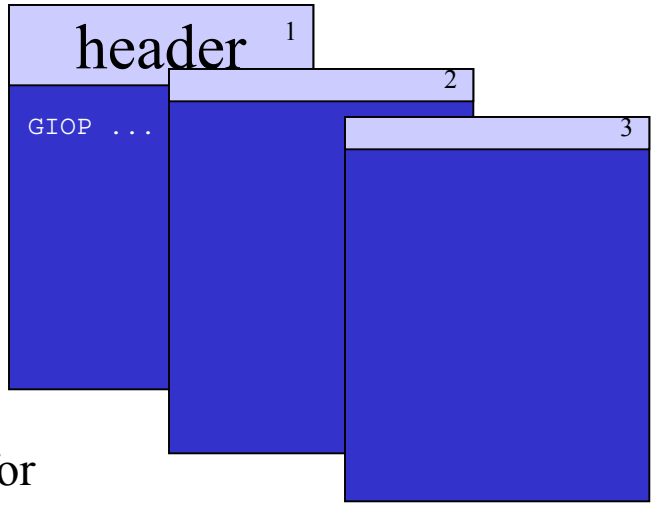
Link 16 Transport

- Secure RF broadcast network
  - no connections!
- Reliable delivery, error detection and correction
- Most messages are content specific, except JTIDS free text message
  - this message used to transmit secure digital voice data
  - compressed imagery not tolerant of transmission errors
- Free text (~byte stream) effective data rate < 16kbps
- "Transport" is really a network application
  - GIOP could be tunneled thru JTIDS messages, though

Free Text Message Transport Layer

- message sequence number
- subaddress
  - ala TCP “port”
  - currently, no host identifier
- message status
  - start, during, end
- message byte count
- overhead
  - approx. 4 bytes per GIOP message + 2 bytes for each 54 bytes of GIOP message length
- compatible with other free text data applications

`“link16://00041:12”`



Aside: Why Else to Replace TCP/IP?

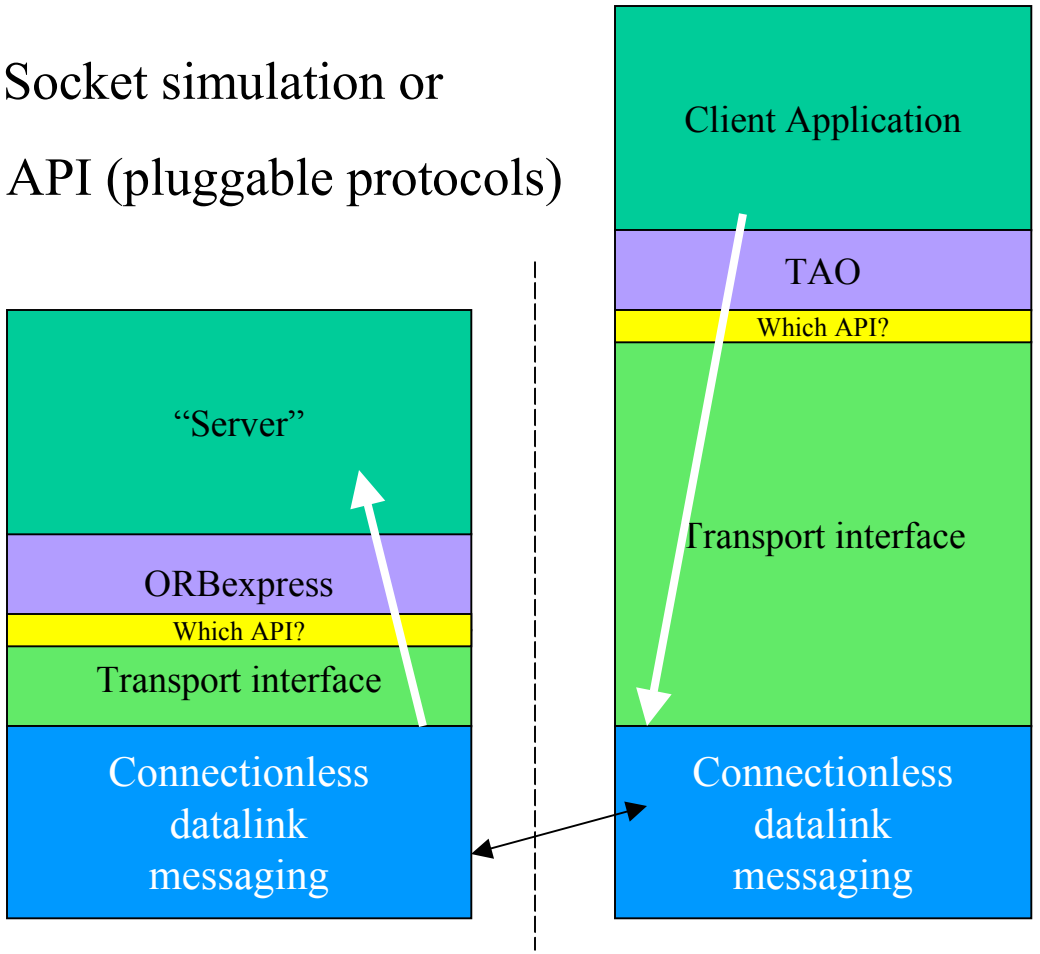
TCP/IP Feature	Protocol or Mechanism	Drawbacks and Limitations
Connection-Oriented		Unsuitable, or at least inefficient, for use in multi-point communication.
Reliable, Ordered	Acknowledgements	Unsuitable or inefficient for simplex, half duplex, or multipoint communication media. Inefficient for high-latency communication connections.
	Retransmission after time-out	Ignores limited time value of some data, e.g. audio, video, track updates
	Source quenching flow control	Unbounded blocking of source may result in unbounded priority inversions
Stream-Oriented		Limits preemption to account for priority

From OIS, CORBA Programming with ORBexpress in Ada 95

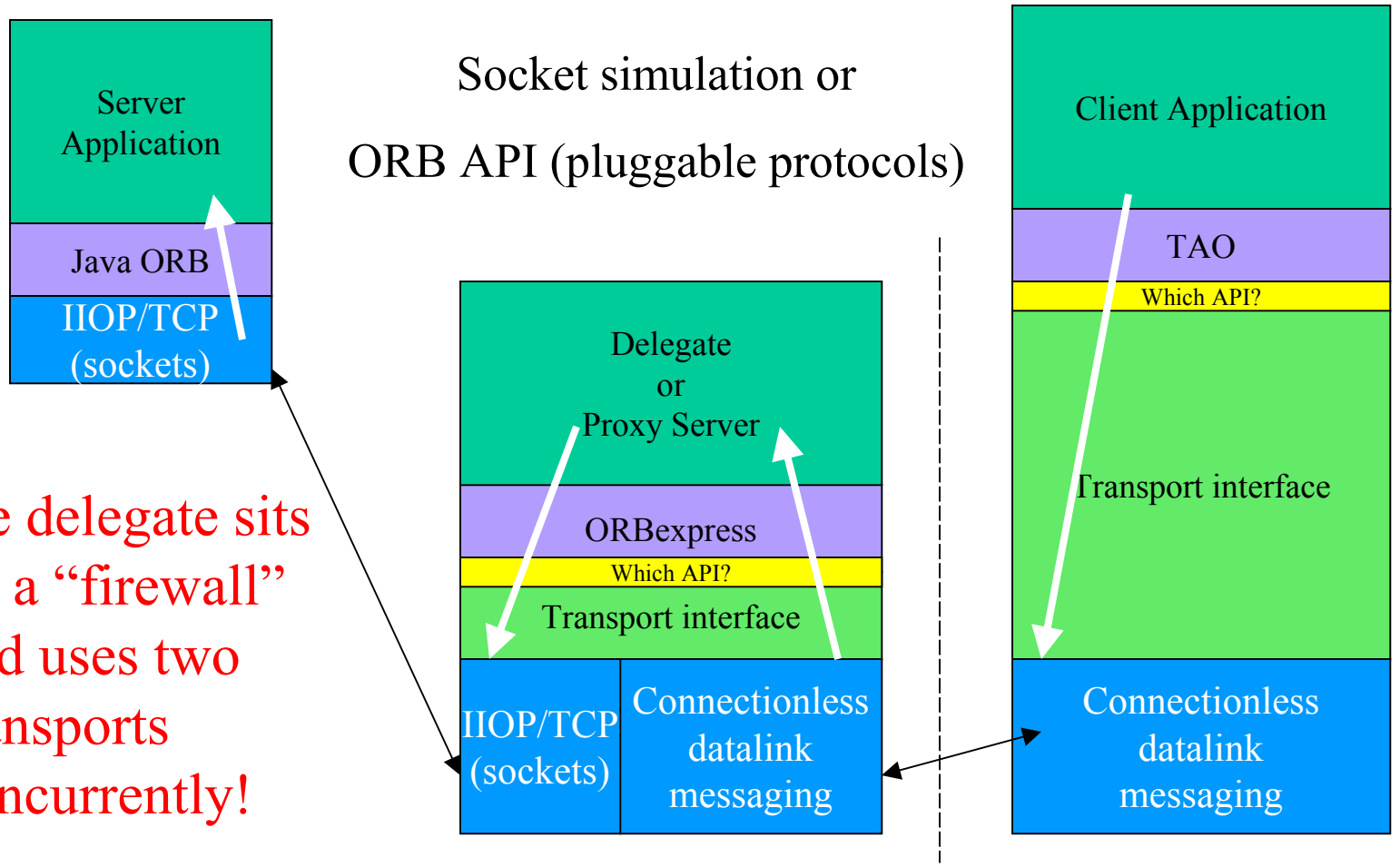
CORBA Implementation - Inline Bridge

Socket simulation or

ORB API (pluggable protocols)



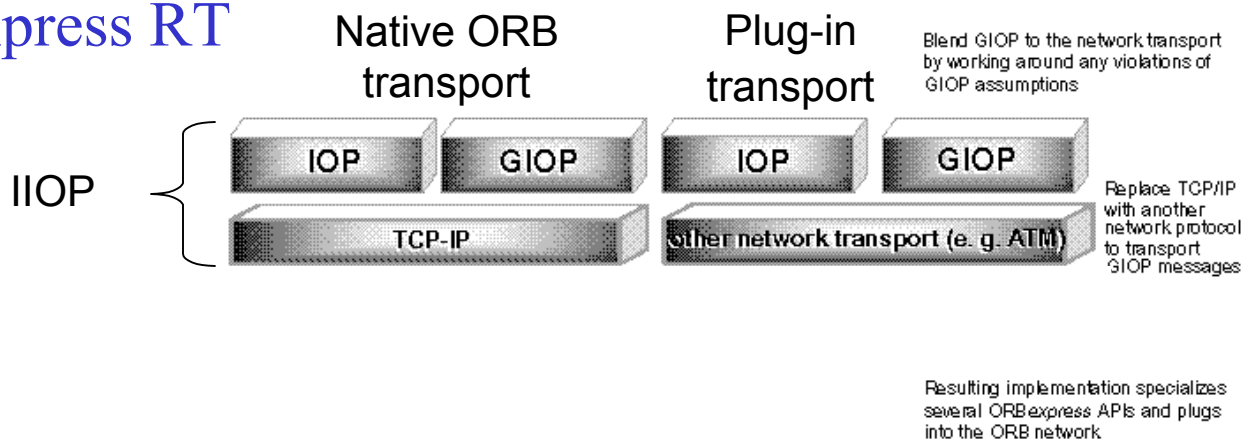
CORBA Implementation - Inline Bridge



the delegate sits on a "firewall" and uses two transports concurrently!

CORBA Implementation - Transport

- Pluggable transport protocol between TAO and ORBexpress RT



- Each ORB has unique transport interface
- Future CORBA specification should standardize this

From OIS, CORBA Programming with ORBexpress in Ada 95

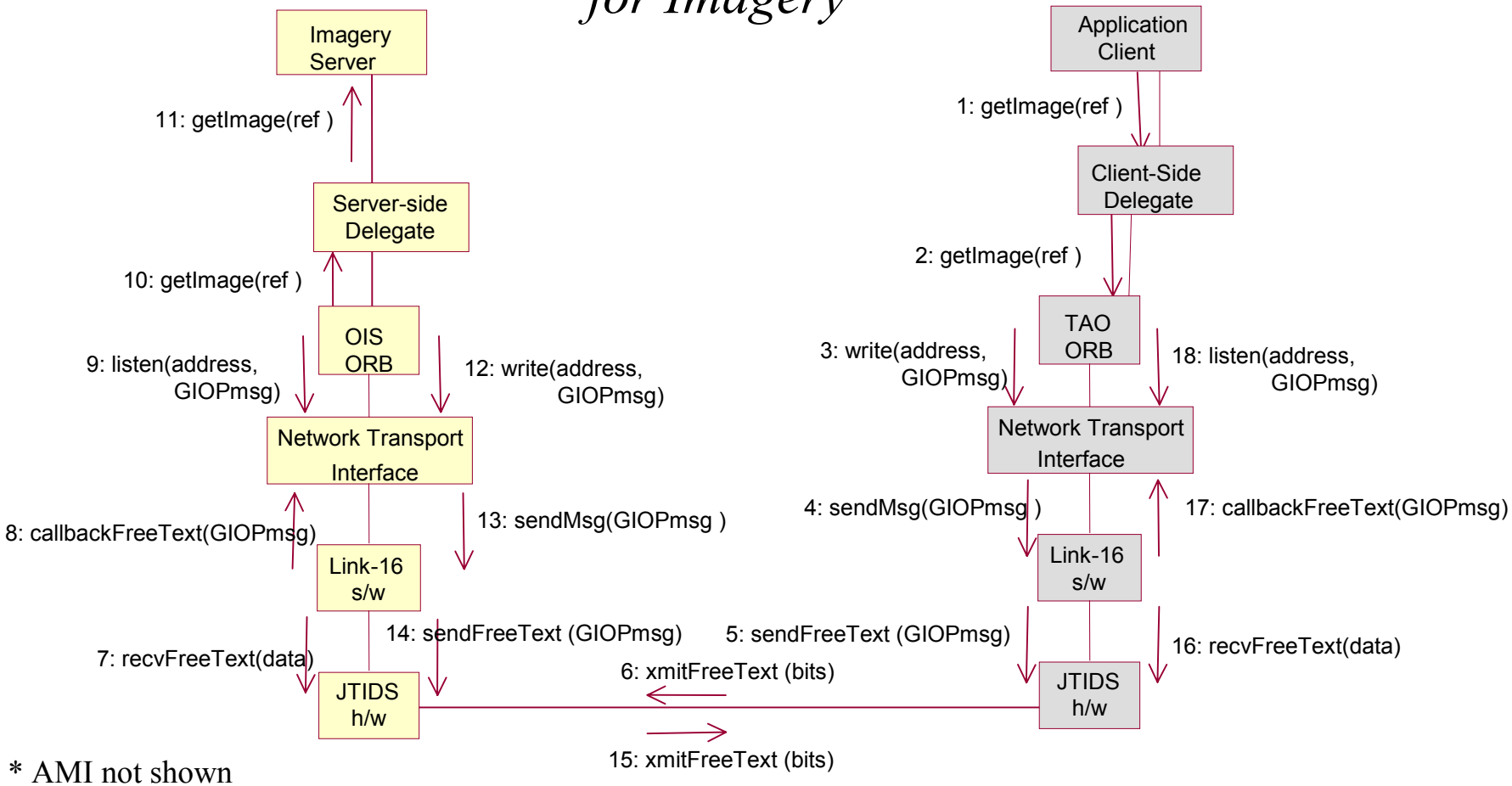
CORBA Implementation - Inline Bridge

Which Network API?

- ORB Pluggable Protocol
  - CORBA recommended method for inline bridging
  - in CORBA spec in the future?
  - uses GIOP
  - clean integration into ORB
  - multiple transports OK
  - simple high level API for ORBexpressRT
  - experience with this method
- Socket API
  - standard network interface for other network apps
  - uses IIOP (GIOP over ~TCP)
  - requires namespace resolution for using real IPC sockets AND socket emulation
  - may have complex system calls to emulate true POSIX sockets
  - experience with this method

CORBA Implementation

C2 — *Client-Server Request for Imagery* — F-15E



* AMI not shown

Honeywell

Washington University in St. Louis

BBN Technologies a part of Verizon

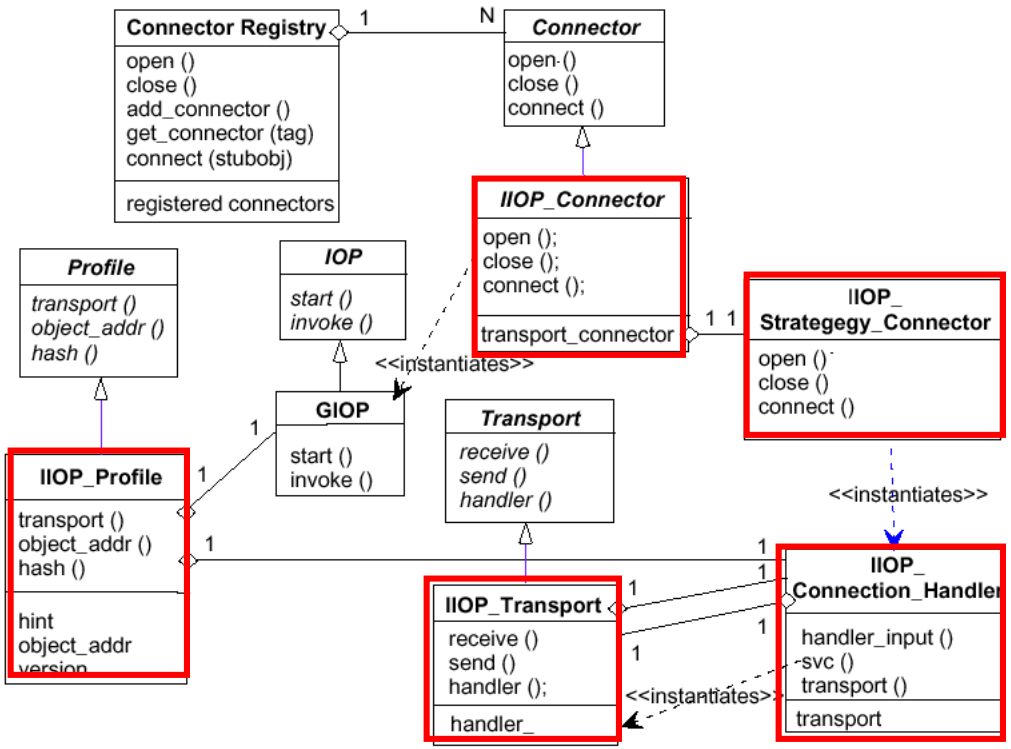
BOEING

GIOP Transport Requirements

Requirement	Comments
• Connection-oriented	• Wrap Link-16 addressing info into transport messages.
• Reliable	• Transport includes orderly delivery.
• Stream-based	• Implemented by free text transport layer.
• Notification of connection-loss	• Asynchronous messaging used to avoid timeouts.
• Connection-initiation model	• Dummy `connect()` call implemented. IOR strings used.

From OIS, CORBA Programming with ORBexpress in Ada 95

Plug-in Transport Framework



TAO Client Class Diagram

classes to implement

from Kuhns et al., *The Design and Performance of a Pluggable Protocols Framework for Real-time Distributed Object Computing Middleware*

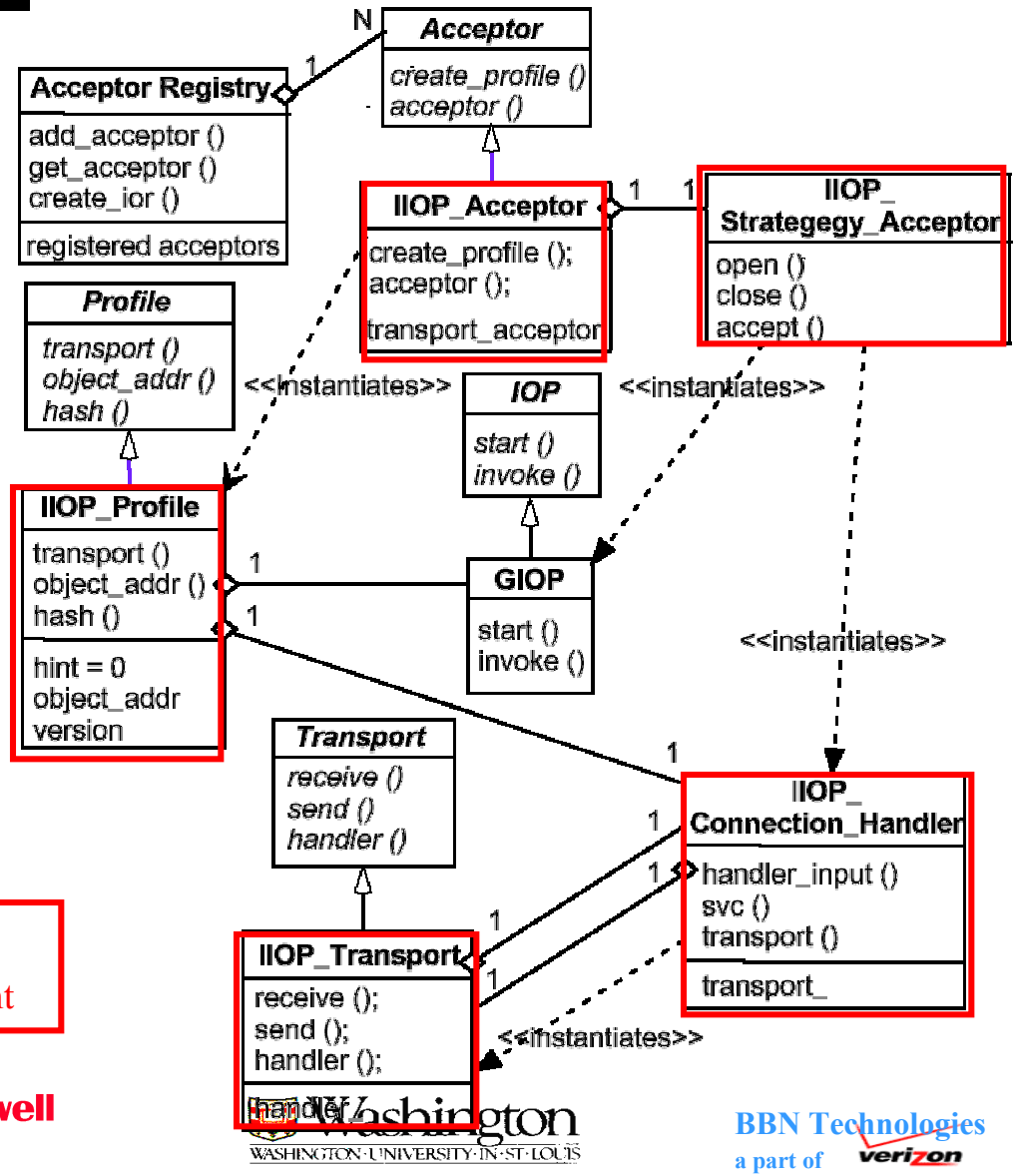
Honeywell

Washington
WASHINGTON UNIVERSITY IN ST LOUIS

BBN Technologies
a part of verizon

BOEING

Plug-in Transport Framework



TAO Server Class Diagram

classes to implement

Honeywell

Washington University in St. Louis

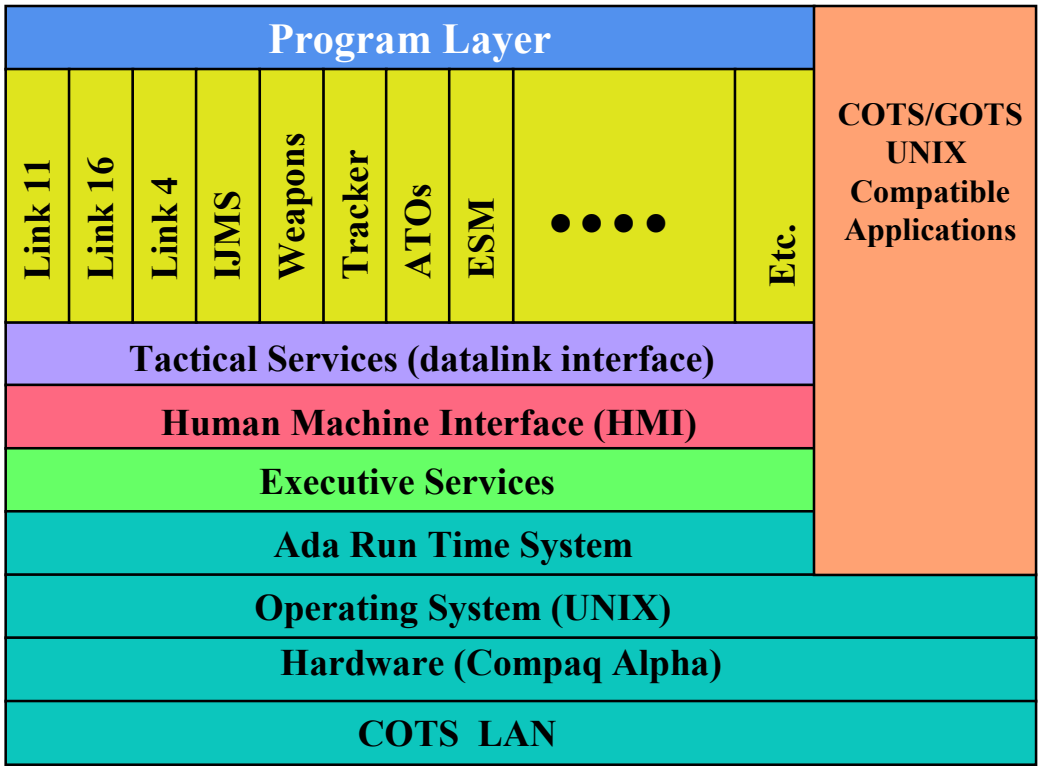
BBN Technologies a part of Verizon

from Kuhns et al., *The Design and Performance of a Pluggable Protocols Framework for Real-time Distributed Object Computing Middleware*

BOEING

WSOA Server Side Architecture

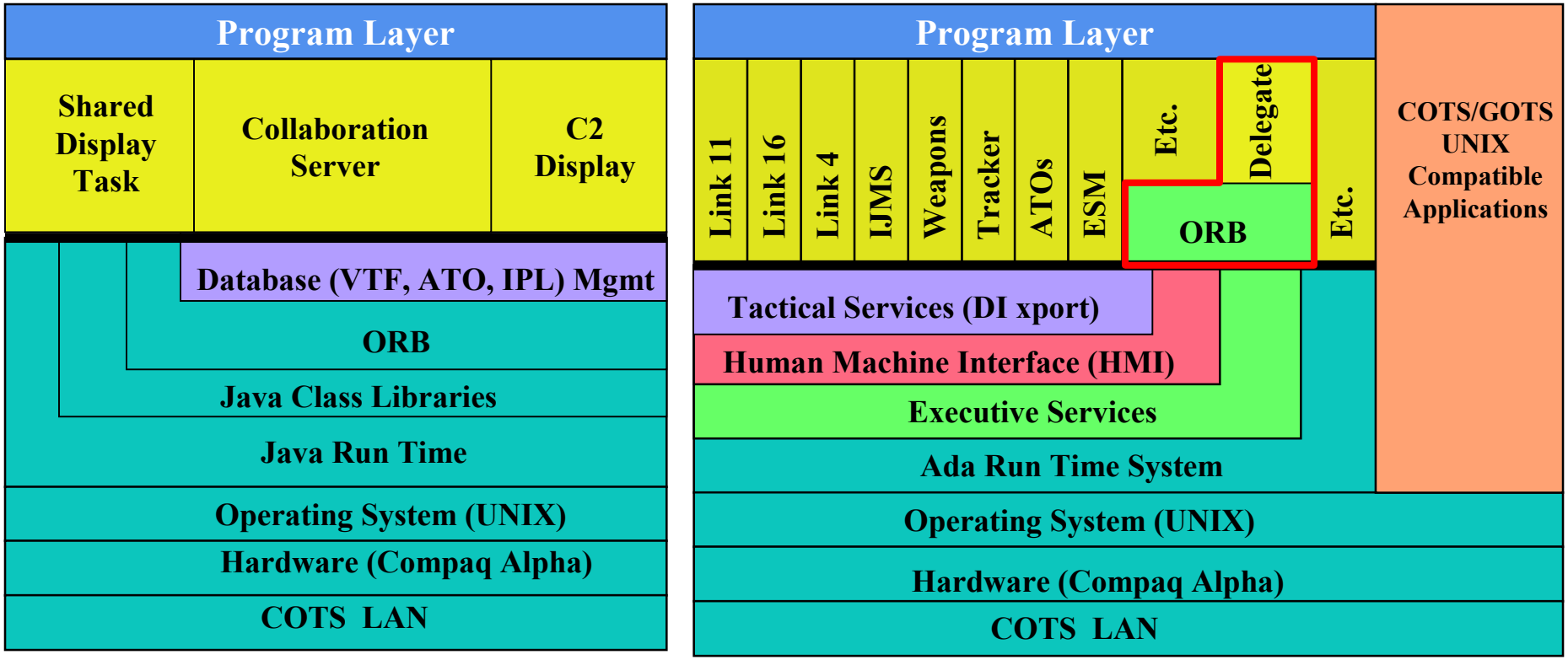
Existing Command and Control Architecture



Mission Computer Program

WSOA Server Delegate Architecture

Detailed WSOA CORBA Architecture



WSOA Server Application

Mission Computer Program

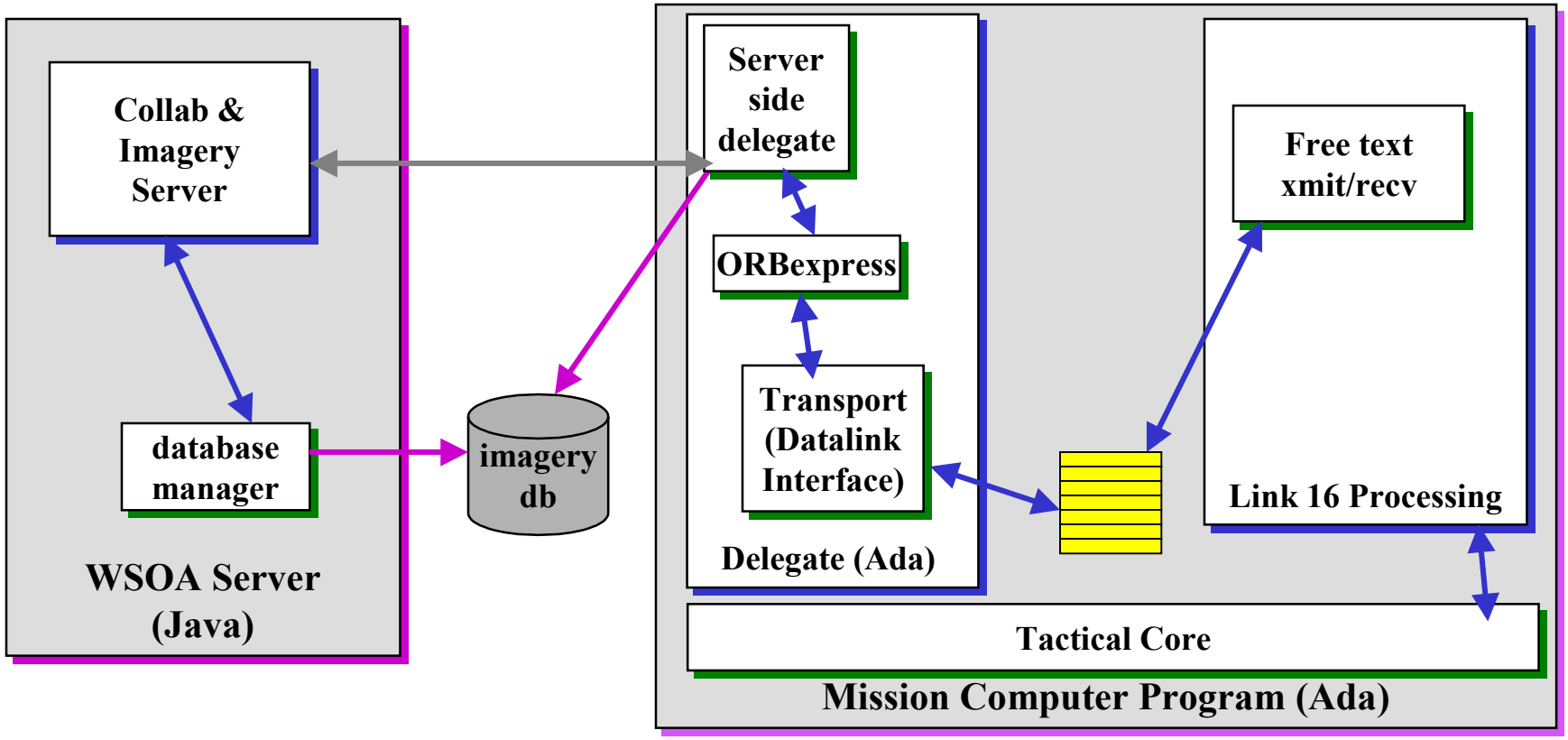
Honeywell

Washington University in St. Louis

BBN Technologies a part of Verizon

BOEING

Concurrency Model - C2 (Server)



Honeywell

Washington University in St. Louis

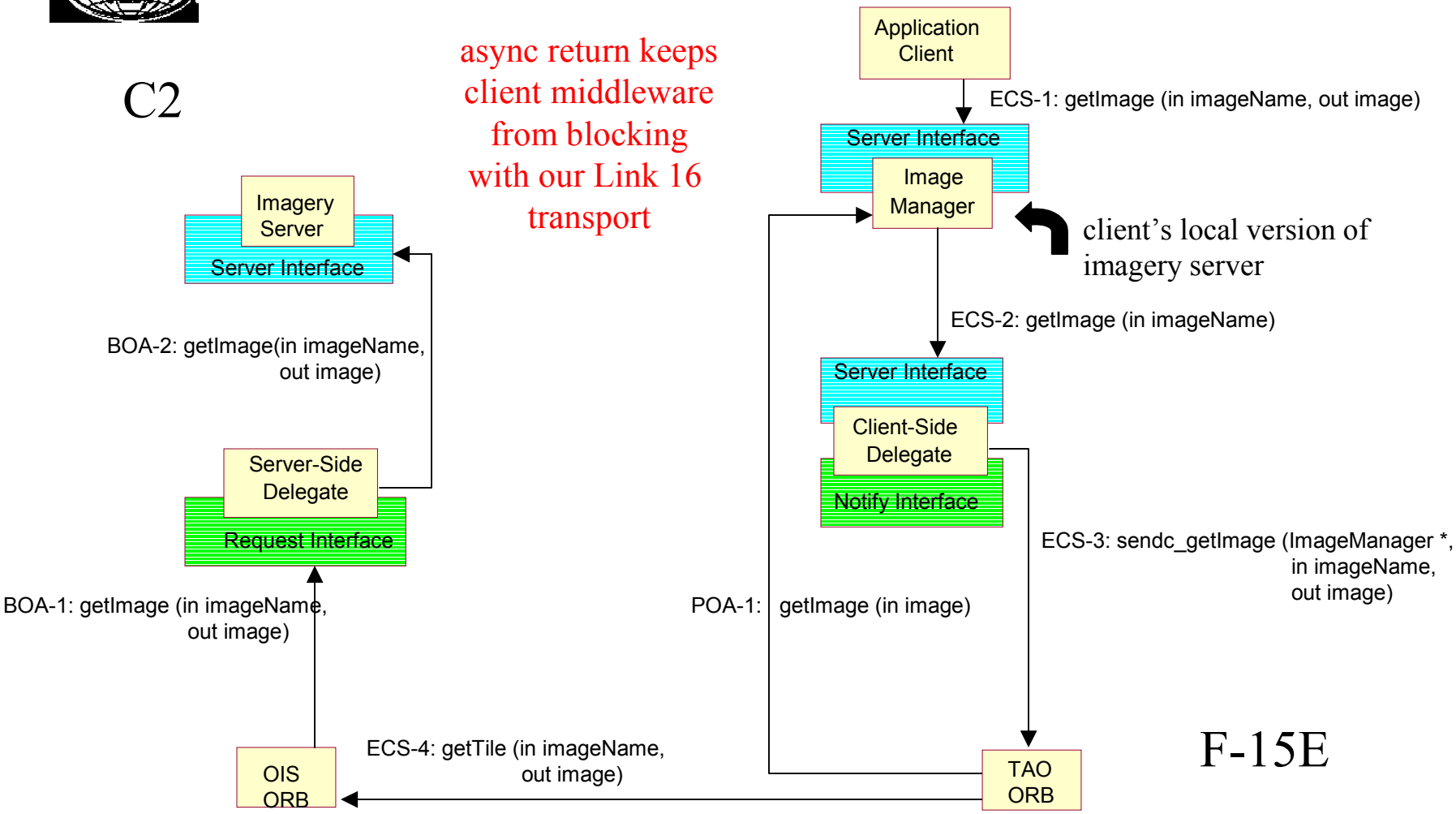
BBN Technologies a part of Verizon

Boeing

Concurrency Model - Client/Server

C2

async return keeps client middleware from blocking with our Link 16 transport



F-15E

Asynchronous Callback AMI Programming Model

Honeywell

Washington University in St. Louis

BBN Technologies a part of Verizon

Boeing

Integration Issues and Lessons Learned

- Full performance data and results available late 2001
  - two way interoperability verified
- Separate transport mapping for each ORB
  - my OE mapping layer:
    ~ 500 SLOC so far
- _IOP Profiles not standard
  - TAO flexible enough to match
  - recommend standardizing on OE's UIOP profile

standard profile for IIOP

```
module IIOP{        // IDL
 struct Version{
  char      major;
  char      minor;
 };
 struct ProfileBody{
  Version     iiop_version;
  string      host;
  unsigned short    port;
  sequence    object_key;
 };
};
```

Honeywell Washington University in St. Louis BBN Technologies a part of Verizon Boeing

Integration Issues and Lessons Learned

- Simple IDL interfaces work well
  - don't push on all sides of the envelope at once!

```
module WSOA
{
  typedef string token;
  typedef sequence <octet> rawImageData;

  exception INVALID_TOKEN
  {
    string explanation;
  };

  interface ClientImage
  {
    rawImageData getImage (in token imageID)
      raises (INVALID_TOKEN);
  };
};
```

- A RT CORBA standard for plug-in transports would be extremely valuable
- Pluggable Protocol Services, too?
  - we built our own COS Name Service wrapper