CORBA

The Good, the Bad, and the Ugly

Douglas C. Schmidt

Washington University, St. Louis

http://www.cs.wustl.edu/~schmidt/

schmidt@cs.wustl.edu

Motivation

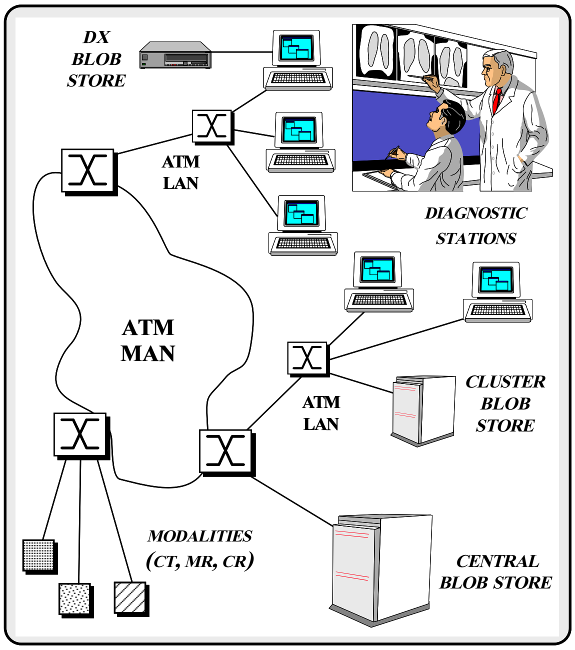
- Typical state of affairs today is the "Distribution Crisis"
  * Computers and networks get faster and cheaper
  * Comm. software gets slower, buggier, more expensive

- Much time wasted on "accidental complexity", *e.g.*,
  - Incompatible software infrastructures
  - Continuous rediscovery and reinvention of core concepts and components

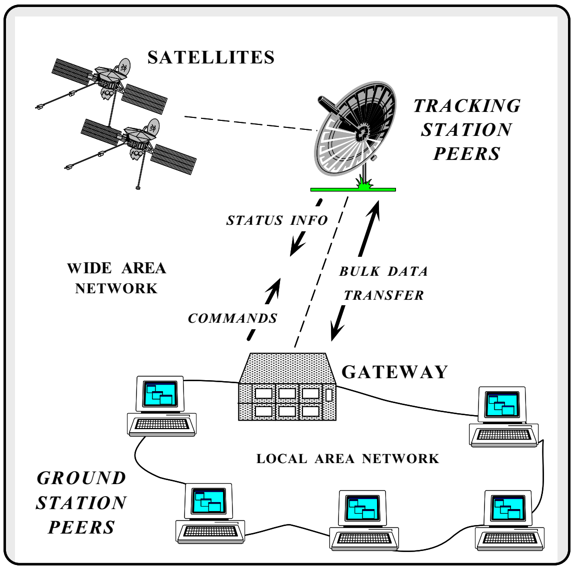
- Also, "inherent complexity" beyond reach of most programmers
  - *e.g.*, latency, partial failures, partitioning, causal ordering, etc.

- CORBA has become the "Holy Grail" of Distributed Object Computing (DOC)
  - Promise to slay the daemons of software complexity, cost, unreliability, etc.

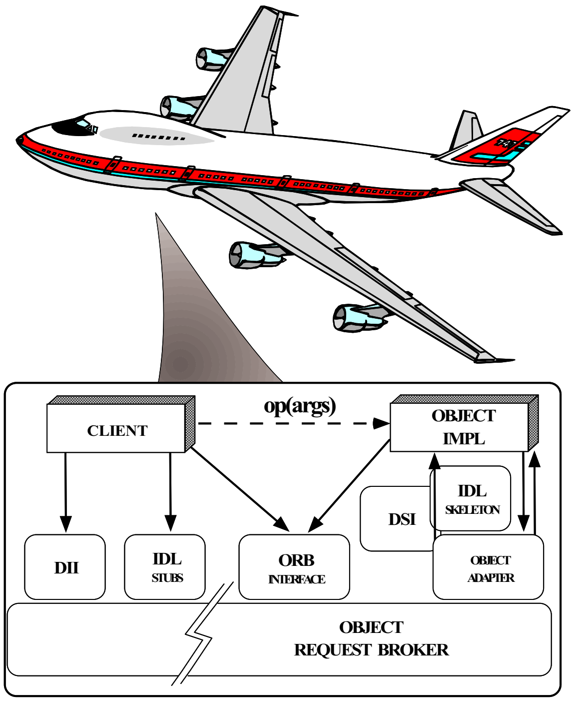
Distributed Medical Imaging



Telecommunications



Avionics



Good News

- ORB infrastructure is stablizing
  - Good ORBs are widely available on most OS platforms
- CORBA Object Services architecture is a useful metaphor
  - *e.g.*, Events, Naming, Lifecycle, Trader, etc.
- User community is forming rapidly
  - *e.g.*, many R&D projects "testing the waters"
- Less "Not Invented Here" syndrome
  - *e.g.*, due to increased complexity and competition

Bad News

- Lack of maturity
  - *e.g.*, many performance, reliability, portability, and interoperability problems
- Lack of integration with existing communication tools
  - *e.g.*, incompatible event loops, name space pollution, often hard to support legacy apps using sockets
- Lack of experience and training
  - *e.g.*, most developers still can't handle inherent complexity, which is not solved by CORBA

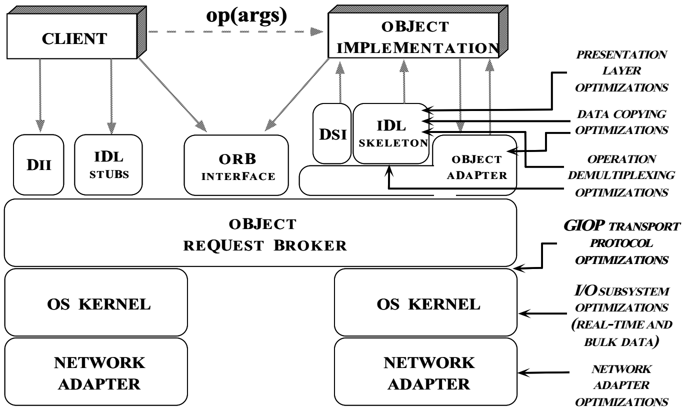
Ugly News

- Lack of standardized *semantics* and *protocols*
  - *e.g.*, layers above the ORB lack meaningful standardized *semantics* and *protocols*
- Lack of truly open standard solution
  - *e.g.*, leads to proprietary systems sold under guise of open systems

Recommendations

- Understand requirements before adopting a communication model
  - *e.g.*, not all applications require high-performance
- Understand ORB performance issues
  - *e.g.*, increase size of socket queues to largest value supported by OS
- Reuse ORBs, reuse COSS *patterns* and *architecture*, but be prepared to build domain-specific services...
- Don't settle for proprietary "open systems"
  - *e.g.*, force the OMG to improve CORBA specifications

CORBA Optimizations



For More Information

- More information about CORBA is available on-line at the following WWW URLs (prefix http:// before each of these)
  - Doug Schmidt's CORBA page
    - ▹ www.cs.wustl.edu/~schmidt/corba.html
    - ▹ www.cs.wustl.edu/~schmidt/corba-research.html
  - LANL's OMG Page
    - ▹ www.acl.lanl.gov/CORBA
  - OMG's WWW Page
    - ▹ www.omg.org/