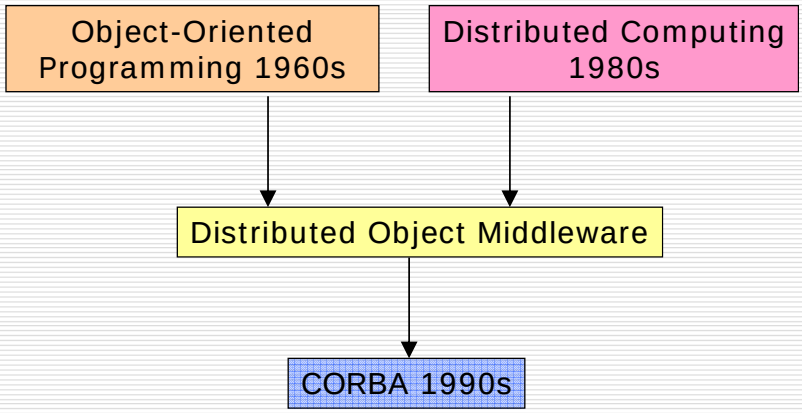
Fault-Tolerant CORBA, A Closer Look

Hadi Salimi
M.S. Seminar
School of Computer Engineering,
Iran University of Science and Technology,
hsalimi@iust.ac.ir
Advisor: Dr. Mohsen Sharifi

Agenda

- CORBA Overview
- FT-CORBA
- Strategies
- Other Paradigms
- Challenges

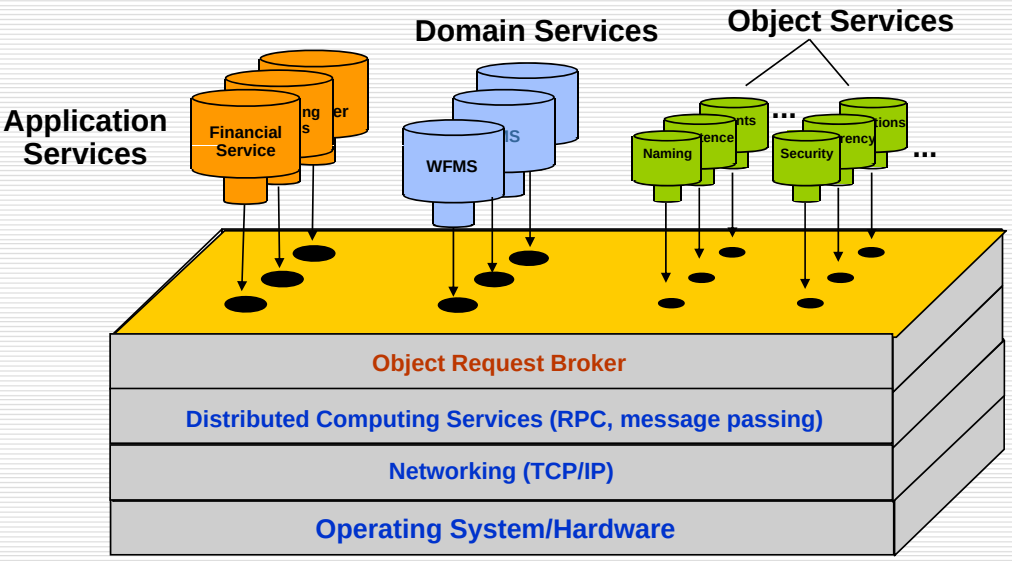
History



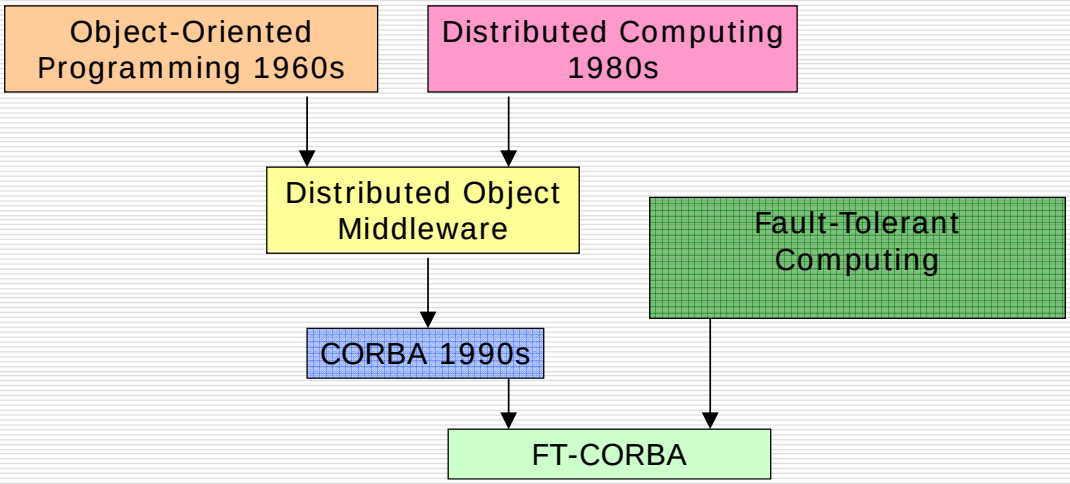
CORBA

- It Stands for Common Object Request Broker Architecture.
- OMG, (A consortium over 800 companies) has produced the CORBA standard; scores of vendors conform.
- The CORBA allows objects to invoke services from other objects, hiding differences in location, programming language or Platform.

CORBA Architecture



FT-CORBA



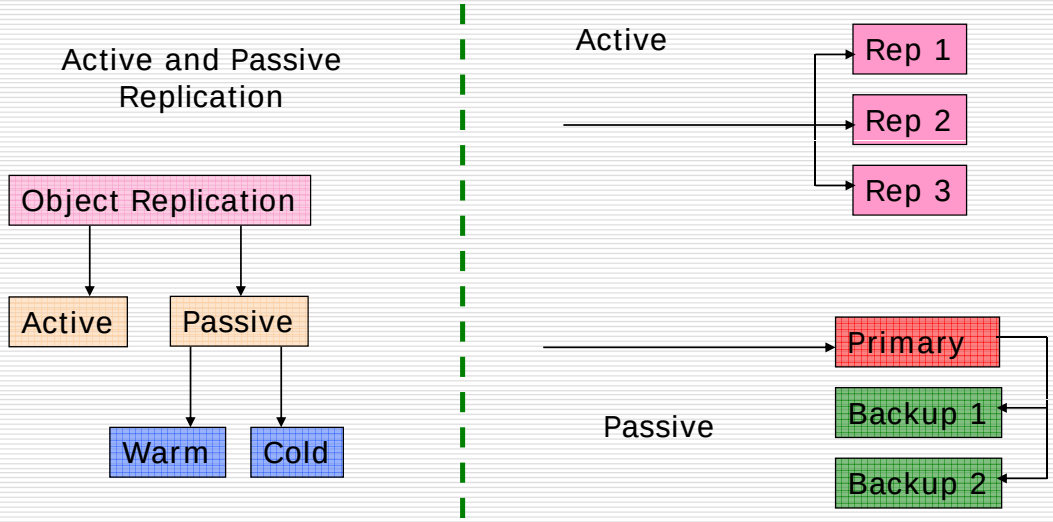
FT-CORBA

- At 1998 OMG issued the RFP
- In early 2000 the first version released
- The last version, December 2001
- using
  - Replication
  - Object Groups

Replication

- The various approaches to fault-tolerant CORBA are alike on their use of replication.
- The behind idea is to mask the failure of an object from a client.

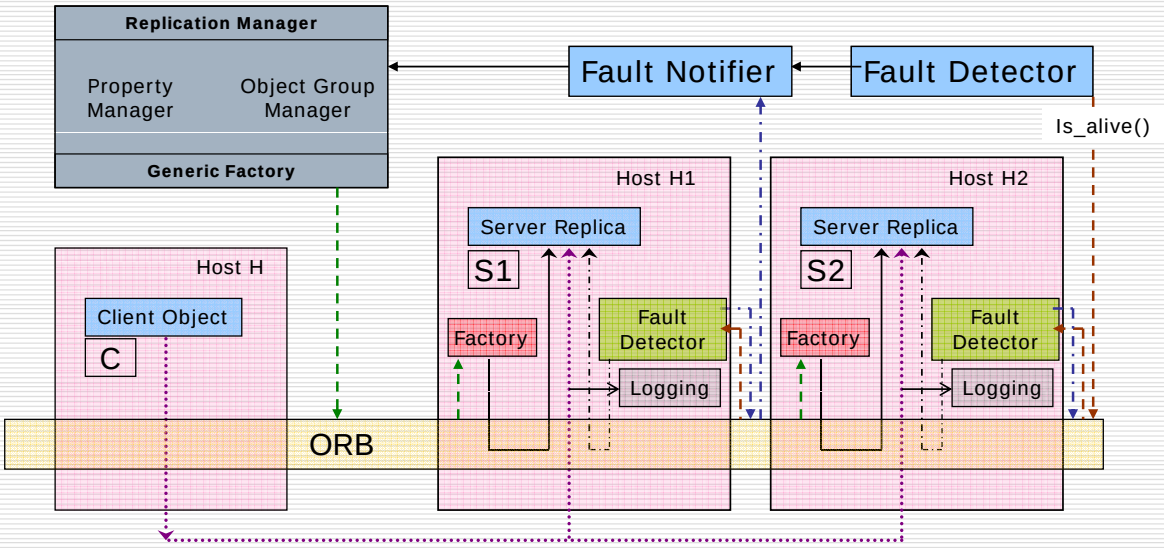
Replication



Object Groups

- Object group represents a replicated object and the group members represent the individual replicas of the object.
- Each object group has an Interoperable Object Group Reference (IOGR).

FT-CORBA Architecture



Other Systems

- Delta-4
  - 1990 – supported by CEC through ESPRIT Project
- Arjuna
  - 1994 – Newcastle University
- Orbix-Isis
  - 1994 – IONA Technologies Co.
- Electra
  - 1995 – Zurich University
- DOORS
  - 1997 - Bell Labs Research
- IRL
  - 1999 – Rome University
- Eternal
  - 2001 – US Air force research Lab.

Strategies

- The Integration Approach
  - Where support for replication is integrated into the ORB.
- The Interception Approach
  - Where support for replication is provided transparently underneath the ORB.
- The Service Approach
  - Where support for replication is provided through an object service.

Integration Approach

- Replace CORBA's TCP/IP based protocol IIOP with Group Communication Protocol.
- So this is non-compliant with standard ORB.
- There is no need to change the clients and the servers on the ORB.
  - Electra
  - Orbix-Isis.

Service Approach

- The ORB is not going to be changed.
- But the application must be aware of the service so the application code will change.
- Every interaction is managed by ORB, So this will be performance poor.
  - DOORS
  - IRL
  - Friends

Interception Approach

- If the interceptions are specific to the operating system, then they need to be ported to the target OS.
  - Eternal

Comparison

Approach	Advantages	Disadvantages	Systems
Integration	Transparent to the Application	Needs porting modified ORB for every object	Electra, Orbix+Isis
Service	CORBA-Compliant can exploit CORBA's Interoperability to work with any ORB	Not always Transparent to Application	OGS, FTS, DOORS, FRIENDS, NewTop
Interception	Transparent to Application and ORB	Needs to be ported to every operating system	Eternal
FT-CORBA Standard	Standardized, configurable support	Leaves Implementation details open, Requires extensions to the standard ORB Core	Eternal, DOORS

FT-CORBA Standard

- Based on Eternal, Doors.
- Parameters:
  - Replication Style: Active, Warm or Cold
  - Application-Controlled or Infrastructure-Controlled
    - Membership Style: Adding and Removal of replicas
    - Consistency Style: Checkpointing, Logging
  - Initial Number of Replicas
  - Minimum Number of Replicas
  - Checkpoint Interval

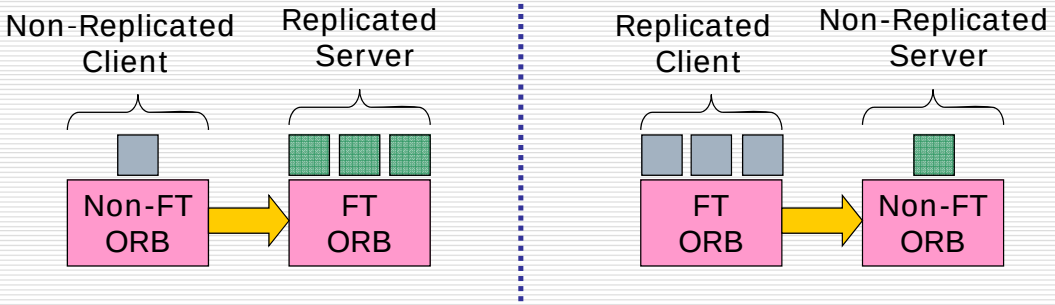
FT-CORBA: A Closer Look…

Server and Client Transparency

- With the FT-CORBA Standard, group specific information can be embedded into object references.
- Client: needs to be recompiled/relinked with new libraries.
- Server: Chackpointable Interfaces for state retrieval must be realized.

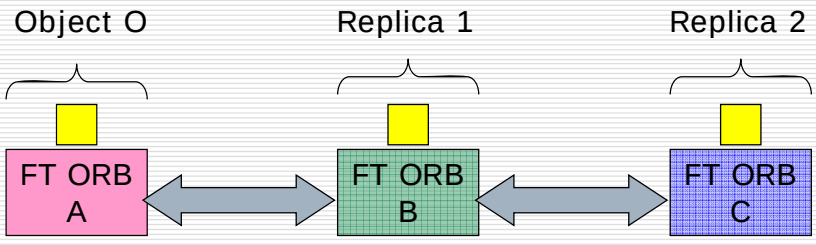
Integration with non-replicated Objects

- The communication of a replicated
  - client with non-replicated servers
  - server with non-replicated clients



Replica Deployment

- Replicas of a CORBA object can not be supported across different FT-CORBA implementations.



Open Issues

- Supporting the CCM
- FT/RT CORBA
- FT and Security
- Replication and Transactions

References

[1] Object Management Group, “Fault Tolerant CORBA (Final Adopted Specification),” OMG Technical Committee Document formal/01-12-29, Dec. 2001.

[2] Object Management Group, “The Common Object Request Broker: Architecture and Specification, 2.6 Edition,” OMG Technical Committee Document formal/02-01-02, Jan. 2002.

[3] P. Felber, P. Narasimhan, “Experiences, Strategies, and Challenges in Building Fault-Tolerant CORBA Systems”, IEEE Transactions in Computers, May 2004.

[4] Narasimhan P. “*Transparent Fault Tolerant for CORBA*”, PHD thesis, University of California, December 1999.

[5] Friedman R. and Hadad E., “*FTS: A High-Performance CORBA Fault-Tolerance Service*”, In the proceedings of the 7th International Workshop on Object-Oriented Real-Time Dependable Systems (WORDS 2002).

References (Cont.)

[6] Slama D. Garbis J. and Russell P. "*Enterprise CORBA*", Prentice Hall, 1999

[7] P. Felber, "The CORBA Object Group Service: A Service Approach to Object Groups in CORBA," PhD thesis, Swiss Federal Inst. Of Technology, Lausanne, 1998.

[8] P. Felber and P. Narasimhan, "Reconciling Replication and Transactions for the End-to-End Reliability of CORBA Applications," Proc. Int'l Symp. Distributed Objects and Applications (DOA '02), Oct. 2002.

[9] IONA and Isis, An Introduction to Orbix+Isis, IONA Technologies Ltd. and Isis Distributed Systems, Inc., 1994.

[10] B. Natarajan, A. Gokhale, S. Yajnik, and D.C. Schmidt, "Doors: Towards High-Performance Fault Tolerant CORBA," Proc. Second Int'l Symp. Distributed Objects and Applications (DOA '00), pp. 39- 48, Feb. 2000.

Fault-Tolerant CORBA, A Closer Look

Hadi Salimi
M.S. Seminar
School of Computer Engineering,
Iran University of Science and Technology,
hsalimi@iust.ac.ir
Advisor: Dr. Mohsen Sharifi