A Management Framework for CORBA-based Distributed Services and Applications

Ji-Young Kong
Dept. of Computer Science and Engineering
POSTECH
Email: konga@postech.ac.kr
http://www.postech.ac.kr/~konga

Contents

- Introduction
- Related Work
- Mgmt Architecture
- Mgmt Interface Object
- CDS MIB
- Mgmt Services
- Prototype Implementation
- Conclusion & Future Work

Introduction

- Explosive growth of CORBA-based Distributed Services(CDS) on Internet & telecommunication networks
- CDSs need to be monitored and controlled to provide reliable and efficient services.
- There does not yet exist international standards for managing CORBA-based services or applications.

Goal of Thesis

- Design a mgmt framework for managing CORBA-based distributed services and applications, including
  - Mgmt Architecture
  - Mgmt Interface
  - Mgmt Information
  - Mgmt Services

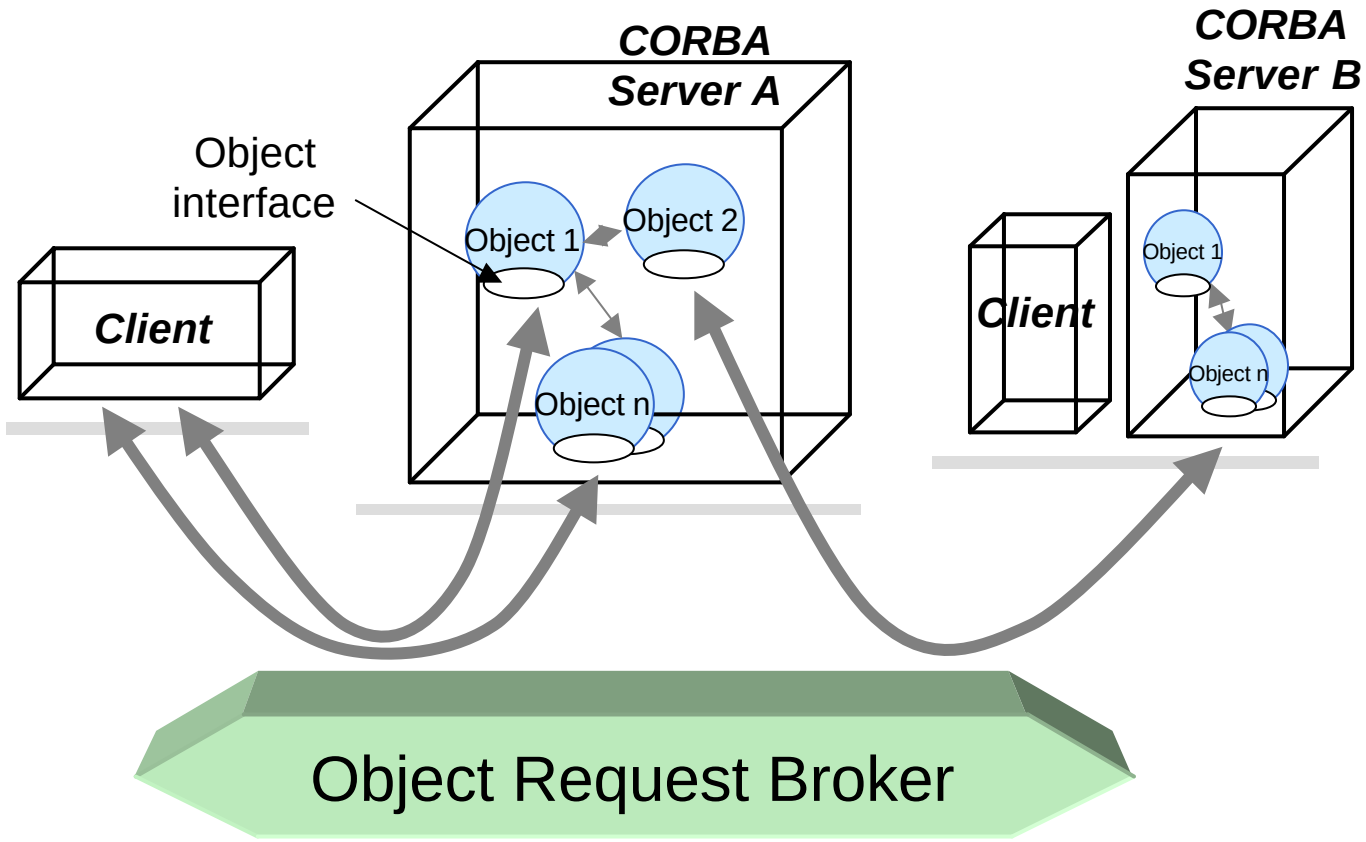
Related Work (1)

- IETF's MIBs for applications
  - Network Services Monitoring MIB
  - System Application MIB
  - Application MIB
  - Mail Monitoring MIB
  - X.500 Directory Monitoring MIB
  - WWW MIB

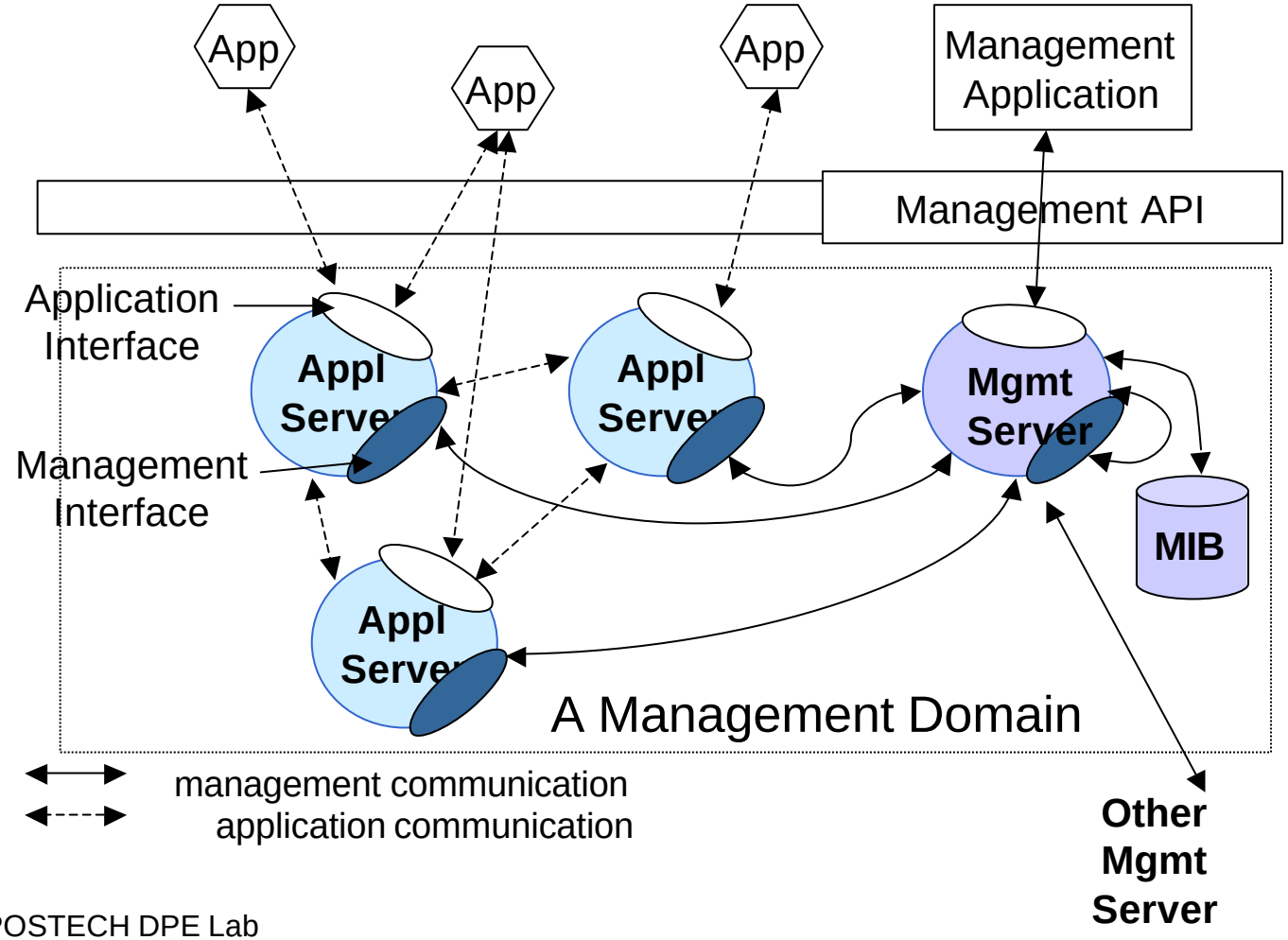
Related Work (2)

- MAScOTTE Project
  - Part of ESPRIT project
  - Aims to provide a set of management services for object-oriented distributed systems.
  - Focuses on
    - Extension of CORBA-based systems for management
    - Management facilities
    - External management platform to CORBA Gateway

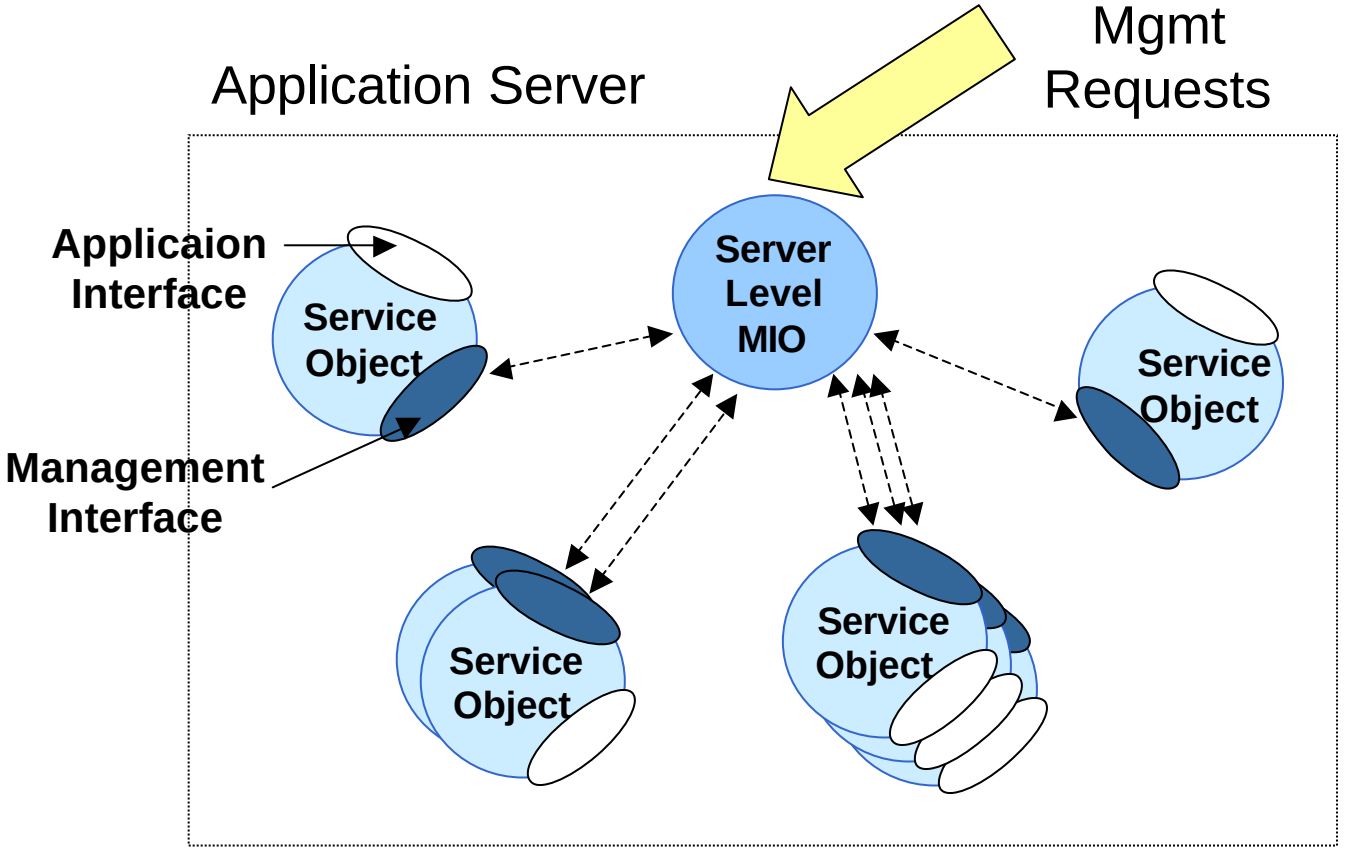
CORBA-based Service Model



Mgmt Architecture



Mgmt Interface Object (1)



Mgmt Interface Object (2)

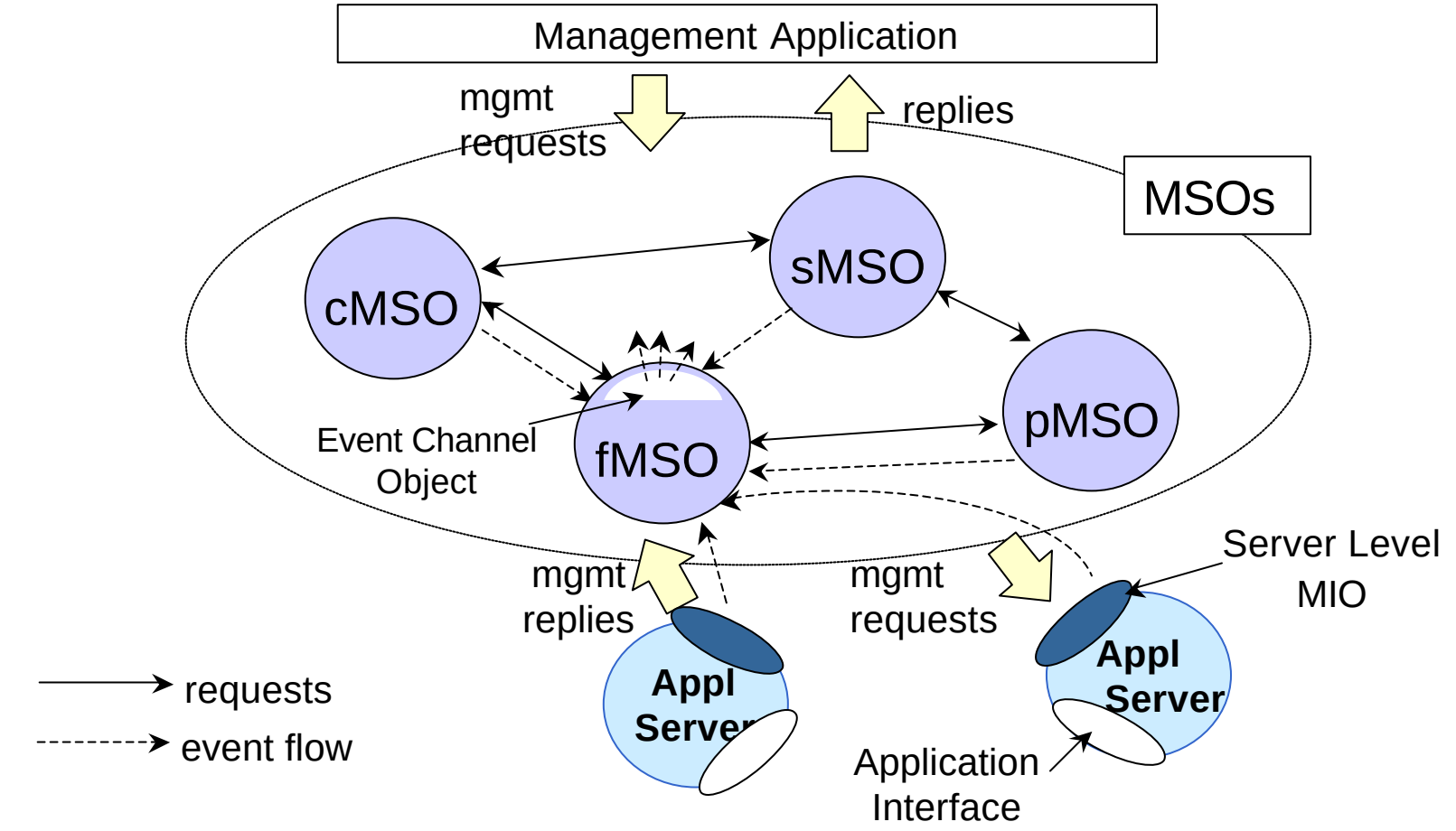
- Enables MSO to monitor and control servers
- Instrumented in every managed server (SO)
- Registered by server and maintained by cMSO
- 2-Level MIOs
  - Sever Level MIO : general information about server
  - Object Level MIO : specific information about each object
- Functions
  - Allowing MSO to access mgmt information
  - Allowing MSO to update mgmt information if necessary
  - Reporting events or faults to MSO

CDS MIB

General Information Group	Server Name, Host Name, Executable, Owner, Protocol, Version, Last Update Time, Last Start Time, Status, Platform, Language
Operational Statistics Group	# of Current In-bound Associations, Requests, Errors, Received/Sent Bytes
Object Information Group	# of Objects, ObjMIO Reference Table
Resource Information Group	CPU Utilization, Memory Space, # of Child Processes, Threads, IPCs

- Provided through MIO interfaces
- Gathered from OS, ORB, and SO by MIO
- Can be specialized using the inheritance feature of the object-oriented technique

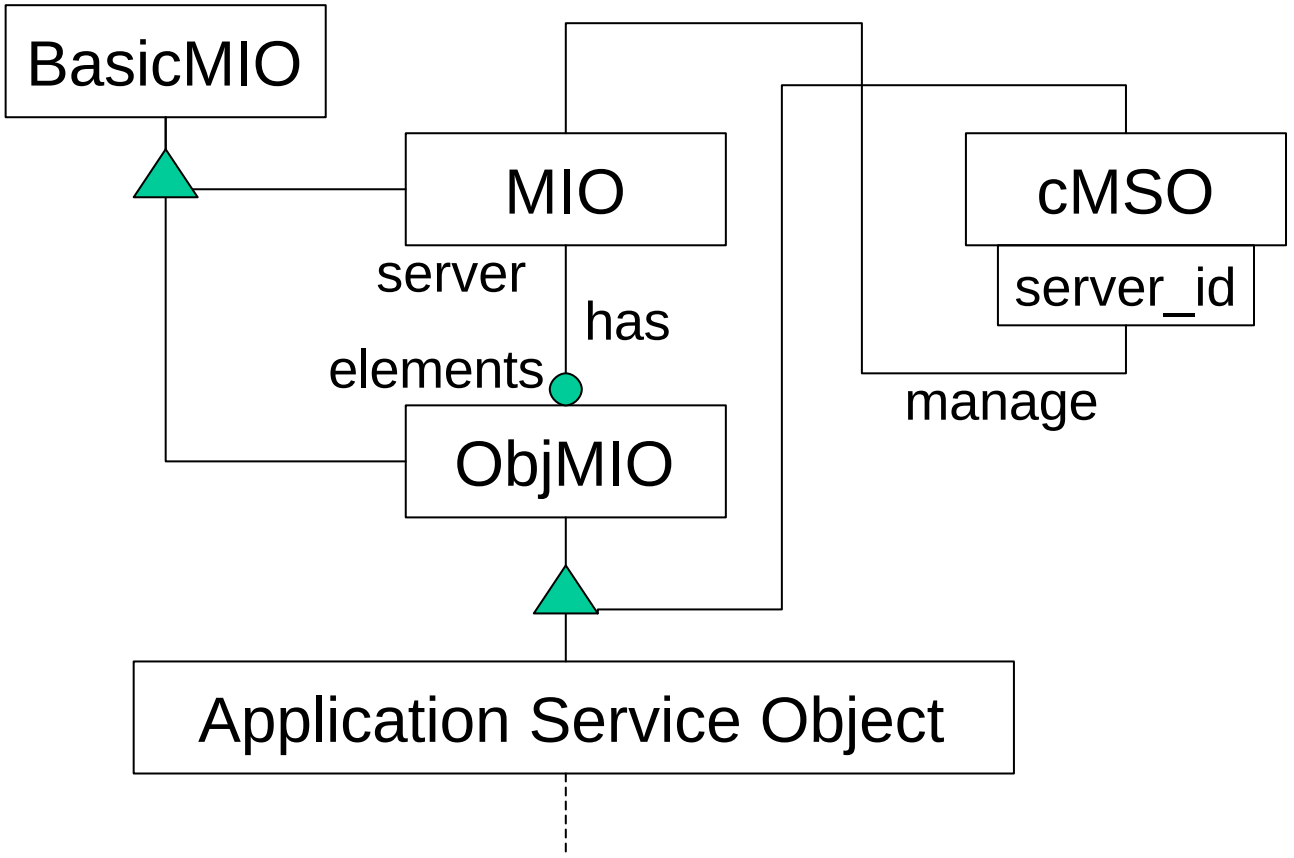
Mgmt Service Architecture



Configuration Mgmt Service

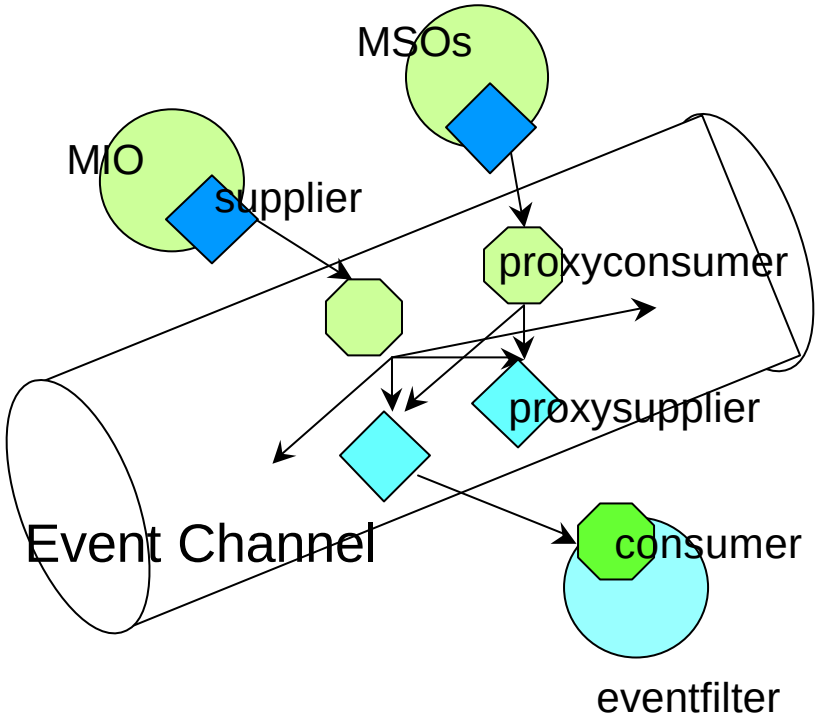
- cMSO is responsible for managing containment, relationships among servers, objects and clients, and attributes of each server and each object.
- Functions
  - Finding servers in a management domain
  - Providing a server's current status
  - Providing information of servers
  - Changing information of servers
  - Initializing or terminating the operation of a server
  - Providing relationships between servers and clients

Object Model for MIO & cMSO

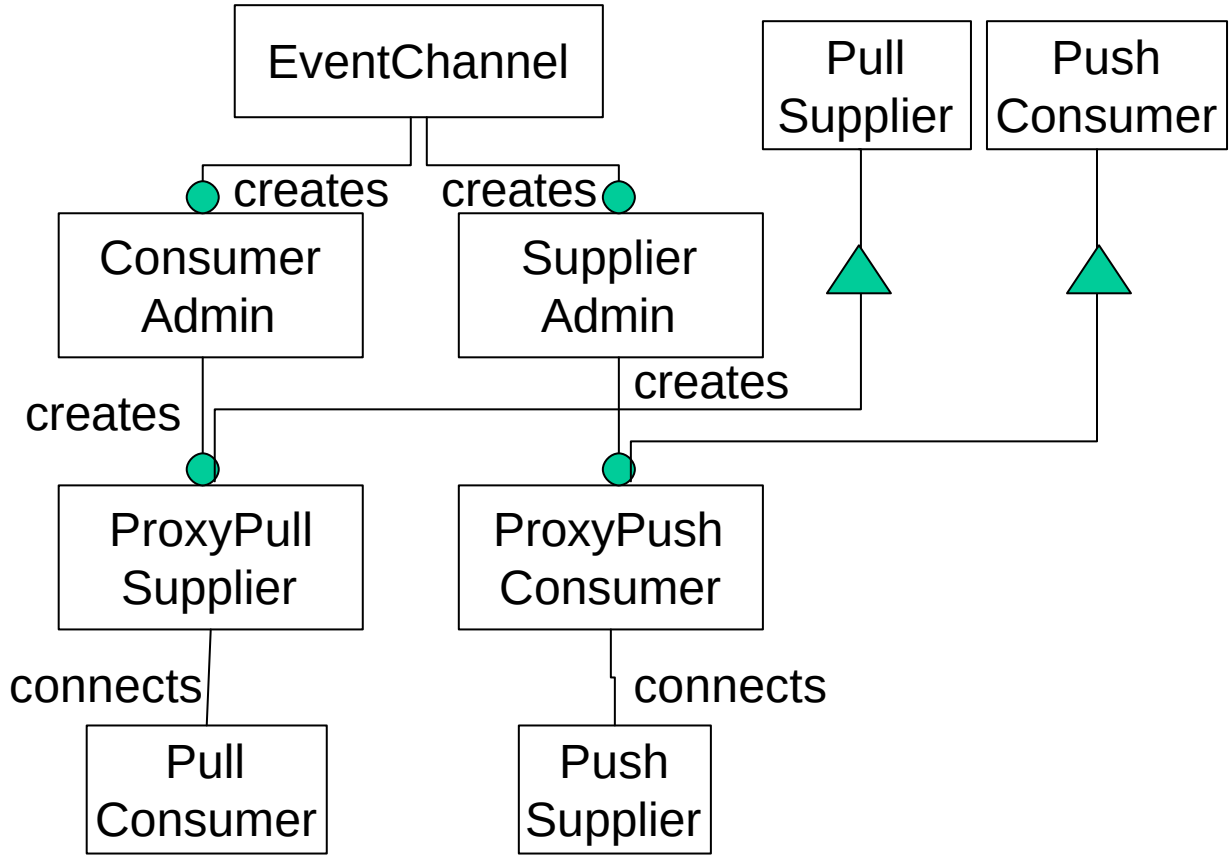


Event Service

- Adopted the OMG CORBA Event Service Specification
- Every object can act as supplier/consumer by including supplier/consumer objects.
- Event Channel creates proxies and manages proxy supplier's event queue lists.
- Incoming events are multicasted through event queue lists.



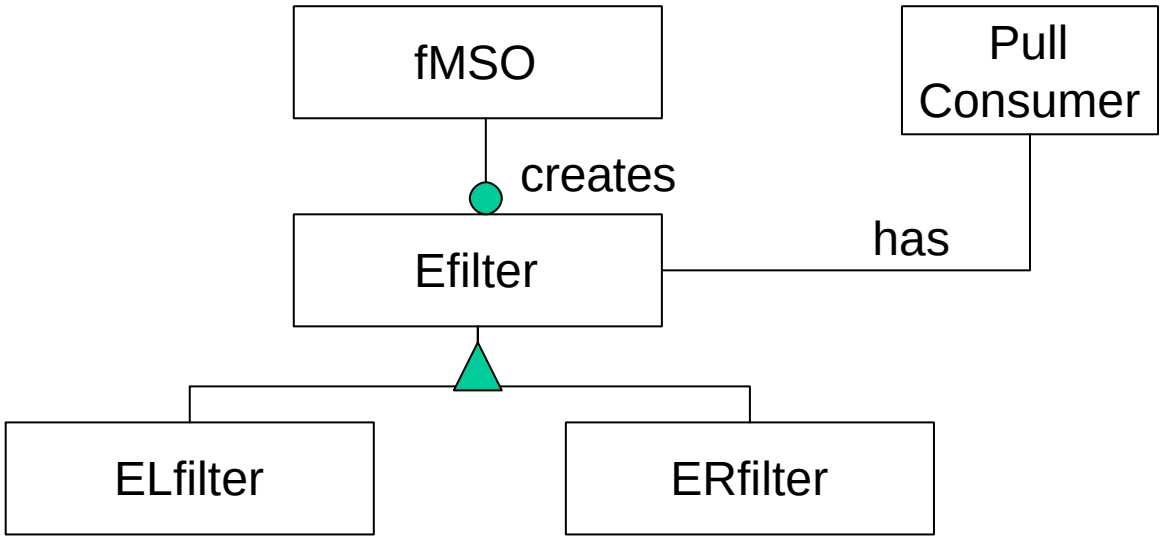
Object Model for Event Service



Fault Mgmt Service

- fMSO is responsible for providing a reliable service environment by handling events and faults.
- Functions
  - Providing logging/reporting event filter objects
    - Allowing users to set scope of events and information logged or notified, and name of a logging file
    - Allowing users to start and stop logging or notifying process
  - Fault Diagnosis
    - Analyzing events and detects failures in servers
    - Localizing the causes of the failure
    - Notifying the suspicious causes of the failure

Object Model for fMSO



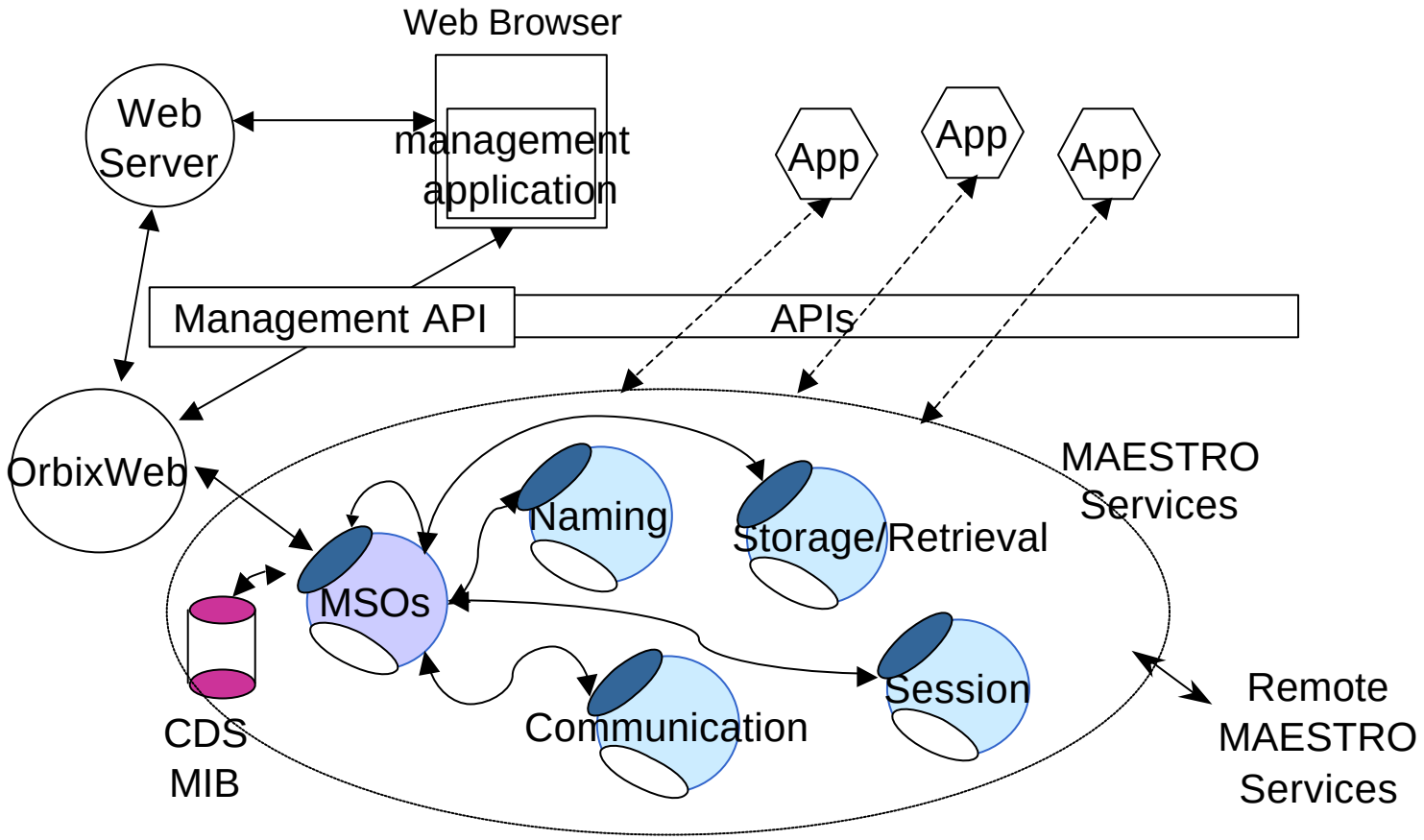
Performance Mgmt Service

- pMSO is responsible for providing an efficient service environment by monitoring the behaviors of servers.
- Functions
  - Providing performance related information
    - Average response time, requests, errors, throughput, bytes
    - Minimum, Maximum, Current etc
  - Logging performance data
  - Setting performance thresholds
  - Notifying when a dangerous condition happens

Security Mgmt Service

- sMSO is responsible for providing a secure environment by controlling the access on services and information.
- Functions
  - Authentication
    - Checking valid users of the management system (adding user, deleting user, login, unlog)
  - Authorization
    - Confining access on services and information to authorized users (setting ACL, getting ACL)

Prototype Implementation Architecture

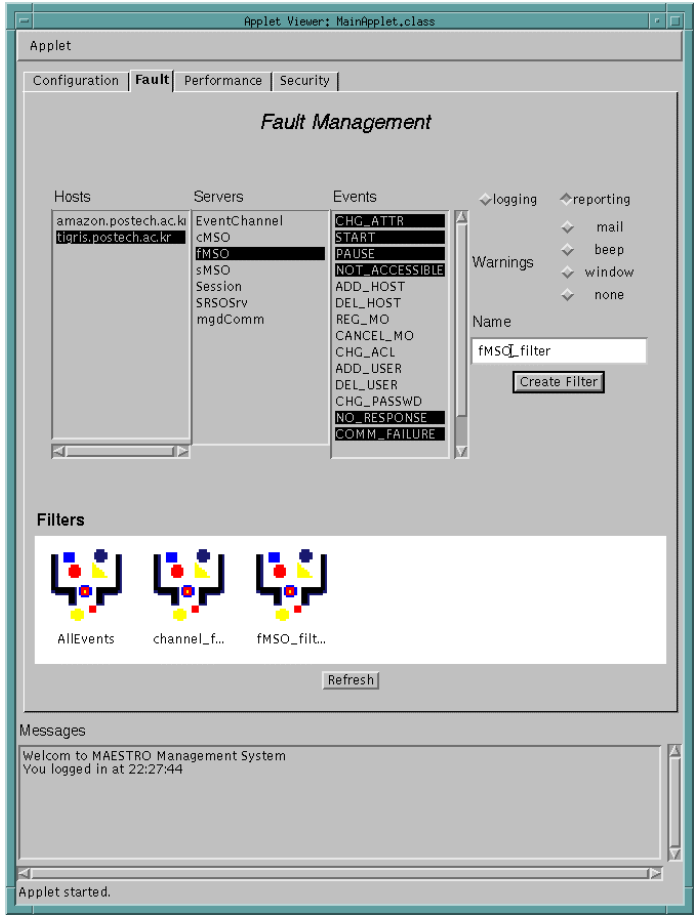


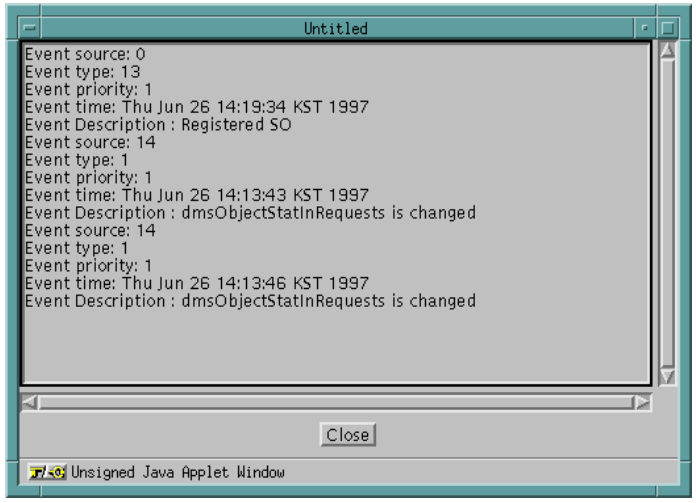
Implementation Environment

- Management Servers
  - Orbix 2.2 & Sparc C++ 4.0
  - Solaris 2.4, 2.5
- Management Application
  - written in Java
  - OrbixWeb 2.0.1
  - Web-based Interfaces
  - Platform independent management application

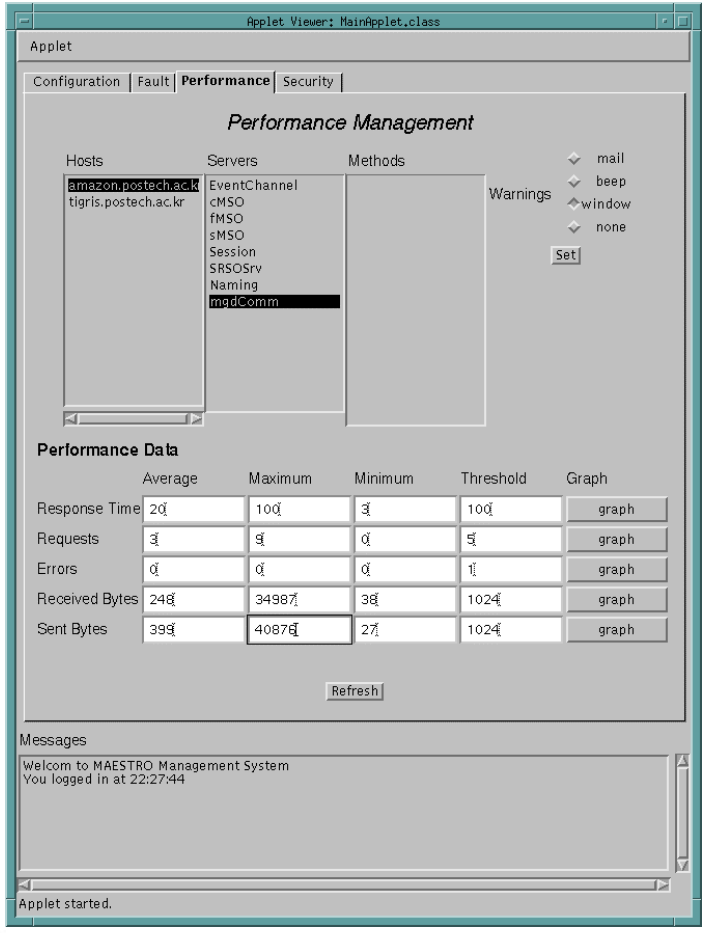
Configuration Mgmt Tool

Event & Fault Mgmt Tool





Performance & Security Mgmt Tool



Conclusion & Future Work

- Management Framework for CORBA-based Distributed Services and Applications
  - CDS MIB, MIO, MSOs, MA
- Our work can be easily extended to manage other CORBA-based services & applications.
- We developed a prototype mgmt system.
- Future Work
  - Integrated web-based management
    - SNMP, CMIP, DMI gateways
  - Management of OMA Components
  - Management of other CORBA-based applications