MONitoring Agents using a Large

Integrated Services Architecture



Iosif Legrand

California Institute of Technology



Distributed Dynamic Services Architecture



- Hierarchical structure of loosely coupled services which are independent & autonomous entities able to cooperate using a dynamic set of proxies or self describing protocols.
- They need a dynamic registration and discovery & subscription mechanism
- For an effective use of distributed resources, these services should provide adaptability and self-organization (aggregation and hierarchical orchestration)
- Reliable on a large scale network distributed environment
 - Avoid single points of failure
 - Automatic re-activation of components and services
- Scalable & Flexible for adding dynamically new services and automatically replicate existing ones to cope with time dependent load



Distributed Object Systems CORBA, RMI, DCOM





The Server and the client code must be created together !!



The client can dynamically generate the data structures and the interfaces for using remote objects based on WSDL

Platform independent



Dynamic Code Loading

Services can be used dynamically

Remote Services Proxy == RMI Stub
Mobile Agents Proxy == Entire Service
"Smart Proxies" Proxy adjusts to the client

Act as a true dynamic service and provide the necessary functionally to be used by any other services that require such information (Jini, interface to WSDL / SOAP)

- mechanism to dynamically discover all the "Service Units"
- remote event notification for changes in the any system
- lease mechanism for each registered unit



JINI – Network Services





A Service Registers with at least one Lookup Service using the same ID.

It provides information about its functionality and the URL addressed from where interested clients may get the dynamic code to use it. The Service must ask each Lookup Service for a lease and periodically renew it.

If a Service fails to renew the lease, it is removed form the Lookup Service Directory. When problems are solved, it can re-register.

The lease mechanism allows the Lookup Service to keep an up to date directory of services and correctly handle network problems.

Monitoring: Data Collection





Registration / Discovery / Remote Notification









Global Client for Farms and Network Connectivity



Mobile Agents and Filters







VRVS Architecture





Reflectors are hosts that interconnect users by permanent IP tunnels.

The active IP tunnels must be selected so that there is no cycle formed.



The selection is made according to the **assumed** network links performance.



Barůvka's Algorithm







4

E

Iosif Legrand

F







INSTITUTE O

ORNIA





ACAT 2003



Global Client / Dynamic Discovery





Iosif Legrand



MonALISA repositories









SUMMARY



- MonALISA is able to dynamically discover all the "Service Units" used by a community and through the remote event notification mechanism keeps an update state for the entire system
- Automatic & secure code update (services and clients).
- **Dynamic configuration for services. Secure Admin interface.**
- Access to aggregate farm values and all the details for each node
- Selected real time / historical data for any subscribed listeners
- Active filter agents to process the data and provided dedicated / customized information to other services or clients.
- Mobile Agents for decision support and global optimization.
- Dynamic proxies and WSDL & WAP pages for services.
- Embedded SNMP support and interfaces with other tools (LSF, PBS, Ganglia, Hawkeye, IEPM-BW...)
- Dedicate pseudo-clients for repository, WAP access or decision making units
- ♦ It proved to be a stable and reliable distributed service system. It is currently running at ~ 100 sites

http://monalisa.cacr.caltech.edu