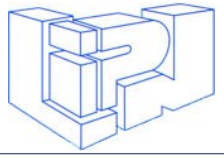
Analysis of Peer-to-Peer Protocols Performance for Establishing a Decentralized Desktop Grid Middleware

Heithem Abbes[1,2], Jean-Christophe Dubacq[2]

[1]Unité de Recherche UTIC
ESSTT, Université de Tunis

[2]Laboratoire LIPN , UMR 7030
CNRS, Université Paris 13



EuroPar, SGS
August, 26-29th 2008

Outlines

- Introduction
  - Desktop Grid
  - Peer-to-peer systems
  - Resource discovery
- Candidates libraries
  - Bonjour
  - Avahi
  - Pastry
- Experimental setup
- Analysis of results
- Conclusion and perspectives

Desktop Grid (1/2)

- Desktop Grid technology consists in exploiting personal computer geographically dispersed:
  - Deliver massive compute power
  - Investigate complex and demanding problems
- XtremWeb (Orsay)
- XW-CH (Genève)
- GTRS (Tunis)

Systems with centralized architecture

Desktop Grid (2/2)

- Number of resources increases:
  - Scalability
  - Auto-organization
  - Dynamic re-configuration
  - Decentralization
  - Performance:
  - Required property
  - Intinsic proterty of P2P systems

Peer-to-peer systems

- P2P tools offer functionalities for the construction of high-level services

  ➡ use P2P tools to construct Desktop Grid systems
- Efficiency of P2P libraries?
  - Simulation
  - Analysis of complexity
  - Very few experimental analyses
- Experimental evaluation ➡ Unavoidable

Resource discovery (1/2)

Fundamental process for the good working of the system

Central:

- global information
- Fast and simple search
- Single point of failure

Decentralized:

- Distributed information (broadcasting)
- Complex research
- No central element

Resource discovery (2/2)

- Three candidates libraries provide the API to construct P2P applications:
  - Bonjour
  - Avahi
  - Pastry
- Experimental evaluation on Grid'5000
  - Registration time of services
  - Discovery time of services.

Objectives

- Choose the best P2P libraries for...
  - Designing and implementing of a decentralized middleware for the Desktop Grid.
  - Decentralized service discovery

Virtualization of the network

High level middleware able to virtualize the network

- no more problems with firewall and NAT

=> Bonjour, Avahi and Pastry on top of such middleware.

Instant Grid / Private Virtual Cluster is one of the candidate for network virtualization.

The main requirements are:

- simple network configuration
- no degradation of resource security
- no need to re-implement existing distributed applications.

Under these assumptions, it is reasonable to check if Bonjour, Avahi and Pastry can scale up.

Outlines

- Introduction
  - Desktop Grid
  - Peer-to-peer systems
  - Resource discovery
- Candidates libraries
  - Bonjour
  - Avahi
  - Pastry
- Experimental setup
- Analysis of results
- Conclusion and perspectives

Bonjour (1/2)

- Bonjour is an implementation by Apple of the ZeroConf protocol.
  - Obtain a functional IP network without DHCP or DNS servers.
- Bonjour is structured around three functionalities:
  - Dynamic allocation of IP addresses without DHCP
  - Resolution of names and IP addresses without DNS
  - Services discovery without directory server.

Bonjour (2/2)

- At a technical level, Bonjour uses
  - Link-Local addresses: An IP address is selected by means of a pseudo-random generator in a defined range of addresses (169.254.1.0-255).
  - mDNS: Machines can still refer the ones with the others by name using the protocol mDNS (multicast DNS).
  - DNS-SD protocol (DNS Service Discovery) to discover the services published in a local area network

Avahi

- Based on Linux implementation of mDNS protocol.
- System that facilitates the service discovery on a local network.
- Allows programs to publish and to discover services and hosts running avahi daemon on a local network without any specific configuration.
- Implementation of DNS-SD specifications and Multicast DNS of ZeroConf.
- Use D-Bus (an asynchronous communication library between process) for the communication between applications.

Pastry (1/2)

- Generic, evolutionary and efficient infrastructure for the P2P applications.
- Fully decentralized, evolutionary, auto-organized.
- Automatic adaptation to the arrival, the departure or the failure of nodes connection.

Pastry (2/2)

- Based on DHT (Distributed Hash Table)
- Pastry nodes form , a decentralized, auto-organized and fault-tolerant, peers network (or logical ring)
  - Creation of bootstrap node.
  - The new arrived node contacts the bootstrap to join the logical ring.
- Pastry node
  - Unique identifier : nodeId
  - State table containing the neighbors
  - A Pastry node routes the message efficiently to the node which is numerically closest, through alive nodes.

Outlines

- Introduction
  - Desktop Grid
  - Peer-to-peer systems
  - Resource discovery
- Candidates libraries
  - Bonjour
  - Avahi
  - Pastry
- Experimental setup
- Analysis of results
- Conclusion and perspectives

- Highly re-configurable and controllable
- Gather 9 sites geographically distributed in France
- All sites are connected by the RENATER network (10 Gb/ses)
- Site of Orsay: machines are connected with a network of 1Gb/s.
- More than 300 machines used.
- All machines have AMD Opterons processors and 1Gb/s network cards.

Specific image for Grid'5000

- Grid'5000 offers an infrastructure with standard images.
- Personalization of an image to support Avahi, Bonjour and Free-Pastry.
- NTP to synchronize all machines
- OAR and Kadeploy
  - Reservation and deployment of the specific image on all machines

Evaluation criterias

- Scalability and response time of the 3 systems
- The maximal number of nodes that can be supported by these tools
- Necessary response time (“latency”) to discover the new node that has just connected on the network (according to the state of the grid).

Service registration

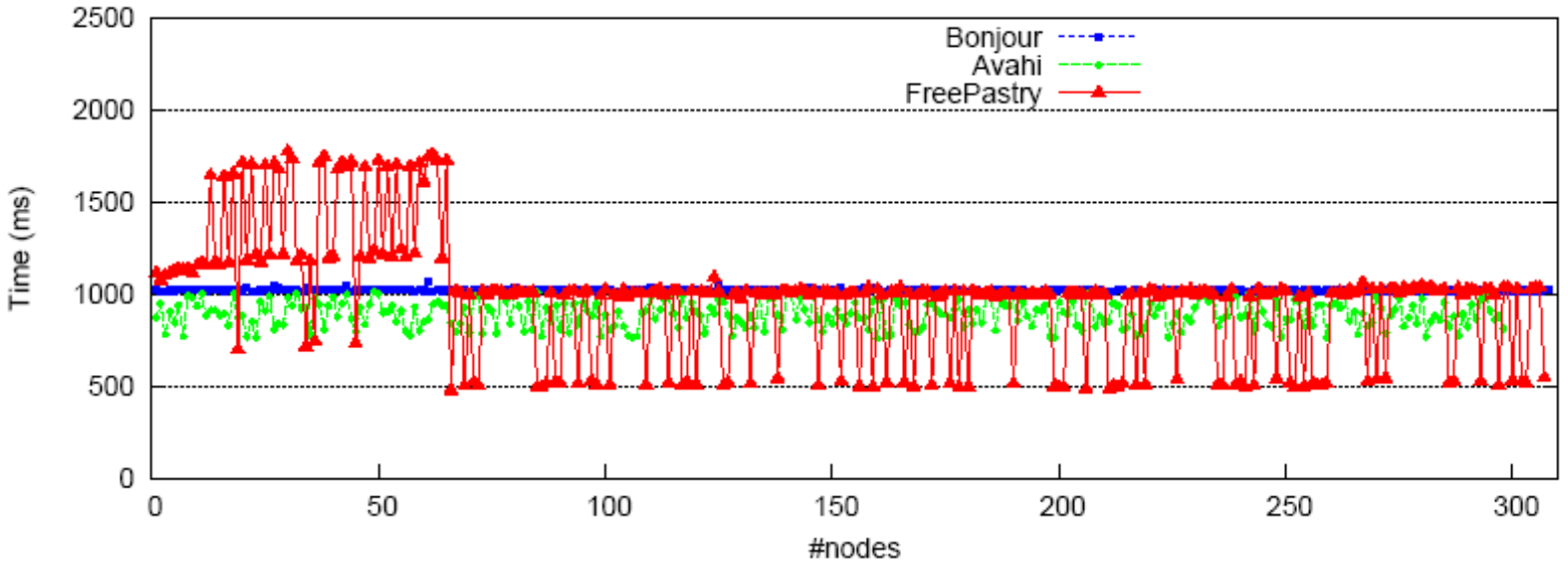
- Registration of one service on each machine
  - If the service is running, the machine is connected on the network.
  - Otherwise (we de-activate the service) the machine is disconnected.
- 2 types of registration
  - Sequential registration
    - Every 60 sec, we activate a service
    - Only multicast per minute (non stressed network)?
  - Simultaneous registration
    - All services are activated on the same instant .
    - N multicasts at a given “tic” (stressed network)
  - Time of registration = Time between the demand and the receipt of notification

Outlines

- Introduction
  - Desktop Grid
  - Peer-to-peer systems
  - Resource discovery
- Candidates libraries
  - Bonjour
  - Avahi
  - Pastry
- Experimental setup
- Analysis of results
- Conclusion and perspectives

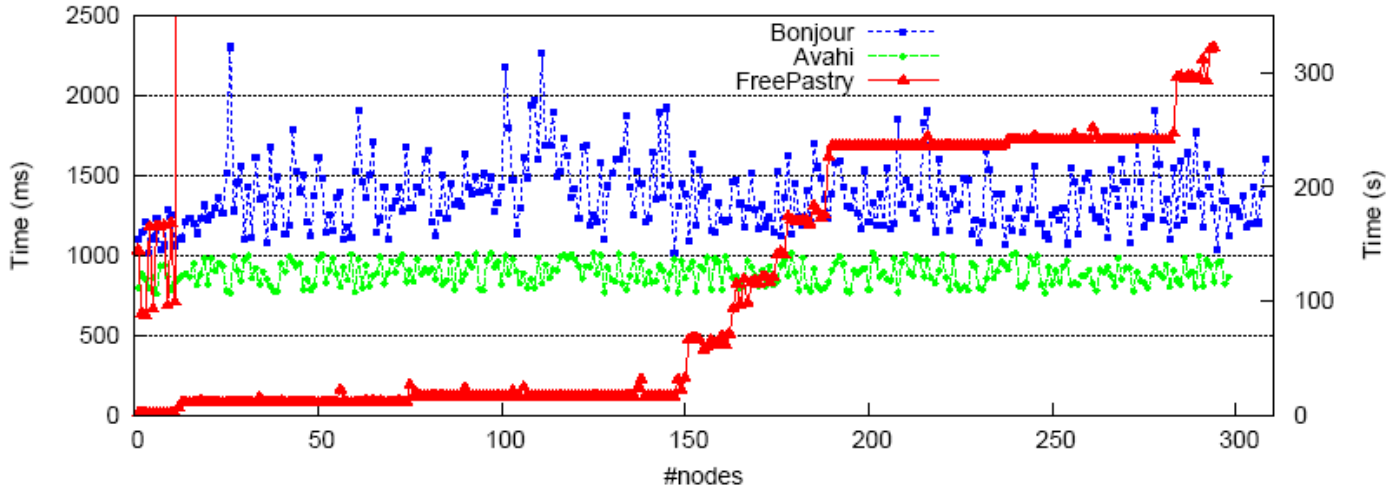
Sequential registration

- There is not a great difference between the three libraries.
- Bonjour and Avahi give similar results .
- Pastry spends almost same time to register 60% of services, needs less times to register first 30% but more times than Avahi and Bonjour for the rest (10%).



Simultaneous registration

- Avahi is the best since it spends less time (last registered service needs 1000 ms).
- Bonjour requires 1300 ms to register the last service.
- Pastry gives times close to those given by Avahi until 160 registrations, beyond this, it spends times definitely larger (until 320 000 ms) that those of Avahi and Bonjour.



Service discovery (1/2)

- Discovery time : the elapsed time between the registration end of a service and its discovery time by the browser..

- Bonjour
  - Bonjour is reliable and very powerful in resources discovery.
  - It discovers all the services with 0% of loss
  - It discovers more than 300 services published simultaneously in less than 1 second.

Services discovery (2/2)

- Avahi
  - loses 60% of registered services (simultaneous version)
  - Improvement in the sequential version (loses just 4 services)
- Pastry
  - 3% of lost services (among 300 registered services) in the two versions (Sequential and Simultaneous)?
  - Discovery time <= 1 second

Outlines

- Introduction
  - Desktop Grid
  - Peer-to-peer systems
  - Resource discovery
- Candidates libraries
  - Bonjour
  - Avahi
  - Pastry
- Experimental setup
- Analysis of results
- Conclusion and perspectives

Conclusion

- The three protocols showed high performances except that Avahi failed to discover all services in simultaneous version and Pastry spends a considerably long time to register all services simultaneously.
- Bonjour is very powerful in registration and discovery in both sequential and simultaneous versions.
- Pastry also proved its performance in sequential registration and also in simultaneous case provided when we does not exceed 160 registrations at a given moment.

Perspectives

- We choose Bonjour to construct a decentralized middleware for Desktop Grid: BonjourGrid.
- BonjourGrid Vision:
  - Publish/subscribe system based on Bonjour to monitor the machine status.
  - Conceive and implement a decentralized and multi-coordinators platform of local DesktopGrid.
  - Build an infrastructure which does not depend on a central element.
  - No static coordinator
  - Create the coordinator in a dynamic, automatic way and without any intervention of a system administrator.
  - Each coordinator asks and seeks, in a decentralized way, idle machines to participate in the execution of a given application.

MERCI