From Multi-Cores to Clouds with ProActive Parallel Suite:
UC in Biotech, IT, Finance, and Engineering

D. Caromel, et al.

**Agenda**

1. Background: INRIA, ActiveEon
2. CLOUD Computing
3. ProActive Parallel Suite *Programming, Scheduling, Resourcing*
4. Use Cases & Demos
5. Conclusion: Cloud Revolution ?

Cloud Computing Revolution ?
This cloud computing I don't understand it at all
CLOUD Revolution?

- 1990: PCs
- 2000: Internet for Companies
- 2010: Cloud for Companies

Concept: John McCarthy in 1961 originally coined the expression “Utility Computing” (Electricity, Water, Gas)

Today: How could we do without Internet and Google Search?

In 2020: we will not imagine working without Clouds

Today: We buy Network, Hardware, Software, Services
Tomorrow: Cloud Services (hiding N, H, S)
1. Background
OASIS (HC: 35)

- Researchers (5):
  - D. Caromel (UNSA, Det. INRIA)
  - E. Madelaine (INRIA)
  - F. Baude (UNSA)
  - F. Huet (UNSA)
  - L. Henrio (CNRS)

- PhDs (11):
  - Antonio Cansado (INRIA, Conicyt)
  - Brian Amedro (SCS-Agos)
  - Cristian Ruz (INRIA, Conicyt)
  - Elton Mathias (INRIA-Cordi)
  - Imen Filali (SCS-Agos / FP7 SOA4All)
  - Marcela Rivera (INRIA, Conicyt)
  - Muhammad Khan (STIC-Asia)
  - Paul Naoumenko (INRIA/Région)
  - Viet Dung Doan (FP6 Bionets)
  - Virginie Contes (SOA4ALL)
  - Guilherme Pezzi (AGOS, CIFRE SCP)

- Visitors + Interns
  - PostDoc (1):
    - Regis Gascon (INRIA)
  - Engineers (10):
    - Elaine Isnard (AGOS)
    - Fabien Viale (ANR OMD2, Renault)
    - Franca Perrina (AGOS)
    - Germain Sigety (INRIA)
    - Yu Feng (ETSI, FP6 EchoGrid)
    - Bastien Sauvan (ADT Galaxy)
    - Florin Alexandru Bratu (INRIA CPER)
    - Igor Smirnov (Microsoft)
    - Fabrice Fontenoy (AGOS)
    - Open position (Thales)
  - Trainee (2):
    - Etienne Vallette d’Osia (Master 2 ISI)
    - Laurent Vanni (Master 2 ISI)
  - Assistants (2):
    - Patricia Maleyran (INRIA)
    - Sandra Devauchelle (I3S)

Located in Sophia Antipolis, between Nice and Cannes, Visitors Welcome and PhD Scholarship Avail.!
8 INRIA’s Research Centres

3 800 HC, 217 M Euro

2 900 Scientists
1200 Researchers, Faculty members
1200 Doctoral students
500 Post-Doct & Visiting scientists

1 000 Engineers, Technicians and Staff

8 Research Centres in France
68 Associated Teams worldwide

4 000 Scientific Publications / year
230 Active patents

89 Innovative companies created
Co-developing, Support for ProActive Parallel Suite

Worldwide Customers: Fr, UK, Boston USA
2. Cloud Computing
The CLOUD Solution

Source: ScienceDaily
Clouds: Basic Definition

- Dynamically **scalable**, often **virtualized** resources
- Provided **as a service** over the **Internet**
- Users need not have knowledge of, expertise in, or control over the technology infrastructure

**XaaS: Anything as a Service**

- **Software as a service** (SaaS), CRM, ERP
- **Platform as a service** (PaaS), Google App Engine
- **Infrastructure as a service** (IaaS), Amazon EC2
Clouds in Picture

From Joseph Kent Langley
Grid Computing
- Several administrative Domains
- Virtual Organizations
- Trading not based on Currency

(Too) Hard

Cloud solves the issue:
- Pay as you Go

Distributed, //, & Grid Technologies for Clouds
Multi-Core Push
Symmetrical Multi-Core: 8-ways Niagara II

- 8 cores
- 4 Native threads per core
- Linux see 32 cores!
Today Off The Shelf Multi-Cores, 3 GHz

Intel Xeon 5670, 6 cores

AMD's Opteron 6174, “Magny-Cours”, 12 cores
Multi-Cores: A Few Key Points

- Moore’s Law rephrased: Nb. of Cores double every 18 to 24 months

- Key expected Milestones: Cores per Chips (OTS)
  - 2012: 32 to 64
  - 2014: 64 to 128

- 1 Million Cores Parallel Machines in 2014
- 100 M cores coming in 2020

- Multi-Cores are NUMA, and turning Heterogeneous (GPU)
- They are turning into SoC with NoC
Virtualization
Virtualization

Source: http://www.apac.redhat.com
Virtualization

Sun, Blog Marc Hamilton
Virtualization
What we Used to do as Syst. Admin.
3. ProActive Parallel Suite
Cloud Solution: ProActive Parallel Suite

ProActive Parallel Suite

Java Parallel Toolkit
Multi-Platform Job Scheduler
Resource Manager

Strong Differentiation:
- Java Parallel Programming + Integration
- Portability: Linux, Windows, Mac
- Versatility: Desktops, Cluster, Grid, Clouds

Used in Production Today:
50 Cores → 300 Cores 2010

Perfect Flexibility
ProActive Programming:
Active Objects
ProActive Programming View
ProActive Programming View

GPU nodes
ProActive: Active objects

- A ag = newActive ("A", [...], VirtualNode)
- V v1 = ag.foo (param);
- V v2 = ag.bar (param);
- ...
- v1.bar(); //Wait-By-Necessity

**Java Object**  **Active Object**  **Future Object**  **Proxy**

**Req. Queue**  **Thread**  **Wait-By-Necessity**

is a
Dataflow Synchronization
Standard system at Runtime: No Sharing

NoC: Network On Chip

Proofs of Determinism
ASP: Asynchronous Sequential Processes

\[
(a, \sigma) \rightarrow_S (a', \sigma') \quad \text{(LOCAL)}
\]

\[
\alpha[a; \sigma; i; F; R; f] \parallel P \rightarrow \alpha[a'; \sigma'; i; F; R; f] \parallel P
\]

- \(\gamma\) fresh activity
- \(i' \notin \text{dom}(\sigma) \quad \sigma' = \{i' \mapsto AO(\gamma)\} :: \sigma\)
- \(\sigma_\gamma = \text{copy}(i'', \sigma) \quad \text{Service} = (\text{if } m_j = \emptyset \text{ then } \text{FifoService} \text{ else } i''.m_j())\)

\[
\alpha[R[\text{Active}(i'', m_j)]; \sigma; i; F; R; f] \parallel P
\rightarrow \alpha[R[i']; \sigma'; i; F; R; f] \parallel \gamma[\text{Service}; \sigma_\gamma; i''; \emptyset; \emptyset; \emptyset] \parallel P
\]

- \(\sigma_{\alpha}(i) = AO(\beta) \quad i'' \notin \text{dom}(\sigma_\beta) \quad f_{i}^{\alpha \rightarrow \beta} \text{ new future} \quad i_f \notin \text{dom}(\sigma_{\alpha})\)
- \(\sigma'_{\beta} = \text{Copy}\&\text{Merge}(\sigma_{\alpha}, i'; \sigma_\beta, i'') \quad \sigma'_{\alpha} = \{i_f \mapsto \text{fut}(f_{i}^{\alpha \rightarrow \beta})\} :: \sigma_{\alpha}\)

\[
\alpha[R[i.m_j(i')]; \sigma_{\alpha}; i_\alpha; F_{\alpha}; R_{\alpha}; f_{\alpha}] \parallel \beta[a_\beta; \sigma_\beta; i_\beta; F_\beta; R_\beta; f_\beta] \parallel P \rightarrow
\]

\[
\alpha[R[i_f]; \sigma'_{\alpha}; i_\alpha; F_{\alpha}; R_{\alpha}; f_{\alpha}] \parallel \beta[a_\beta; \sigma'_{\beta}; i_\beta; F_\beta; R_\beta; \beta_m; i''; f_i^{\alpha \rightarrow \beta}; f_j] \parallel P
\]

\[
R = R' :: [m_j; i_\gamma; f'] :: R'' \quad m_j \in M \quad \forall m \in M, m \notin R'
\]

\[
\alpha[R[\text{Serve}(M)]; \sigma; i; F; R; f] \parallel P \rightarrow \alpha[i.m_j(i_\gamma) \uparrow f, R[\emptyset]; \sigma; i; F; R' :: R''; f'] \parallel P
\]

- \(i' \notin \text{dom}(\sigma) \quad F'' = F :: \{f \mapsto i'\} \quad \sigma' = \text{Copy}\&\text{Merge}(\sigma, i; \sigma, i')\)

\[
\alpha[i \uparrow (f', a); \sigma; i; F; R; f] \parallel P \rightarrow \alpha[a; \sigma'; i; F'; R; f'] \parallel P
\]

- \(\sigma_{\alpha}(i) = \text{fut}(f_{i}^{\gamma \rightarrow \beta}) \quad F_\beta(f_i^{\gamma \rightarrow \beta}) = i_f \quad \sigma'_{\alpha} = \text{Copy}\&\text{Merge}(\sigma_\beta, i_f; \sigma_{\alpha}, i)\)

\[
\alpha[a_\alpha; \sigma_{\alpha}; i_\alpha; F_{\alpha}; R_{\alpha}; f_{\alpha}] \parallel \beta[a_\beta; \sigma_\beta; i_\beta; F_\beta; R_\beta; f_\beta] \parallel P \rightarrow
\]

\[
\alpha[a_\alpha; \sigma'_{\alpha}; i_\alpha; F_{\alpha}; R_{\alpha}; f_{\alpha}] \parallel \beta[a_\beta; \sigma_\beta; i_\beta; F_\beta; R_\beta; f_\beta] \parallel P
\]
Standard system at Runtime: No Sharing

NoC: Network On Chip

Active Object  Synchronous Call  Sub System
Passive Object  Asynchronous Call  Address Space
Distributed Objects On Chip
Distributed Objects On Chip, Boards, Clouds
Broadcast and Scatter

Broadcast is the default behavior
Use a group as parameter, Scattered depends on rankings

ag.bar(cg);  // broadcast cg
ProActive.setScatterGroup(cg);
ag.bar(cg);  // scatter cg
Dynamic Dispatch Group

ag.bar(cg);
IC2D: Optimizing
IC2D

Monitoring View

Job Monitoring View

Virtual nodes
- Render
- DefaultVN
- Dispatcher
- User

Display topology
- Proportional
- Ratio
- Filare

15:09:15 -> NodeObject id=Node-455186381 already monitored, check for new active objects
Video 1:
IC2D Optimizing
Monitoring, Debugging, Optimizing
Scheduling & Resourcing
ProActive Scheduling

ProActive Scheduling
Multi-Platform Job Scheduler

ProActive Scheduling

ProActive Parallel Suite

ProActive Parallel Suite
ProActive Scheduling Big Picture

- Multi-platform Graphical Client (RCP)
- File-based or LDAP authentication
- Static Workflow Job Scheduling, Native and Java tasks, Retry on Error, Priority Policy, Configuration Scripts, …
- Dynamic and Static node sources, Resource Selection by script, Monitoring and Control GUI, …
- ProActive Deployment capabilities: Desktops, Clusters, Clouds, …

ProActive Scheduler

ProActive Resource Manager
Workflow Example: Picture Denoising

• with selection on native executable availability (ImageMagik, GREYstoration)
  • Multi-platform selection and command generation
• with file transfer in pre/post scripts
ProActive Resourcing

ProActive Resourcing
Desktop, Cluster, Grid & Cloud Resource Manager

ProActive Resource Manager

ProActive Parallel Suite
RESOURCING User Interface
Clusters to Grids to Clouds:
e.g. on Amazon EC2
Private, Public & Hybrid Clouds

ProActive Scheduler
ProActive Resource Manager

Static Policy
LSF

Timing Policy
12/24
Desksops

Dynamic Workload Policy
EC2

Amazon EC2

“Cloud Bursting!”

Dedicated resources
Desksops
Amazon EC2 Execution

Cloud Seeding strategy to mix heterogeneous computing resources:
- External GPU resources
Cloud Seeding with ProActive

User submit its noised video to the web interface
Cloud Seeding with ProActive

Web Server submit a denoising job the ProActive Scheduler

User

Web Interface

ProActive Scheduler + Resource Manager

Amazon EC2

CPU nodes

GPU nodes

ProActive Scheduler

ProActive Parallel Suite
Cloud Seeding with ProActive

CPU nodes are used to split the video into smaller ones

User

Web Interface

ProActive Scheduler + Resource Manager

Amazon EC2

CPU nodes

GPU nodes

ProActive

Parallel Suite
Cloud Seeding with ProActive

CPU nodes are used to split the video into smaller ones.
Cloud Seeding with ProActive

User

Web Interface

ProActive Scheduler + Resource Manager

Amazon EC2

CPU nodes

GPU nodes

GPU nodes are responsible to denoise these small videos
Cloud Seeding with ProActive

GPU nodes are responsible to denoise these small videos.
Cloud Seeding with ProActive

CPU nodes merge the denoised video parts

User

Web Interface

Amazon EC2

ProActive Scheduler
+ Resource Manager

GPU nodes

ProActive
Parallel Suite
Cloud Seeding with ProActive

- User
- ProActive Scheduler + Resource Manager
- Web Interface
- CPU nodes
- GPU nodes
- CPU nodes merge the denoised video parts
- Amazon EC2
- ProActive Parallel Suite

©2010 Google - Inajerix ©2010 Tetamexico, N-Com, Domaines cartographiques ©2010 ANI
Cloud Seeding with ProActive

User

Web Interface

ProActive Scheduler + Resource Manager

Amazon EC2

CPU nodes

GPU nodes

The final denoised video is sent back to the user
Real Live Demo: ProActive Scheduler & Resource Manager
Use Case 1: BioTechs
IPMC Use Case and Collaboration

SOLID machine from Applied Biosystems

Nodes can be dynamically added!

Clusters

Desksops

Clouds
The distributed version with ProActive of Mapreads has been tested on the INRIA cluster with two settings: the Reads file is split in either 30 or 10 slices.

Use Case: Matching 31 millions Sequences with the Human Genome (M=2, L=25)

4 Time FASTER from 20 to 100
Speed Up of 80 / Th.
Sequential : 50 h ➔ 35 mn

EC2 only test: nearly the same performances as the local SOLiD cluster (+10%)

For only $3,2/hour, EC2 has nearly the same perf. as the local SOLiD cluster (16 cores, for 2H30)
UC 2: IT
SOA Analysis of Web Server Logs
Parallel Services

- Separation: BPEL – Parallel Serv. – Task Flow
- Standards et Portable
- Flexibility
AGOS Platform Management

HP- Business Availability Center (HP-BAC)

- Monitoring of entire platform
- Cover all layers in the scope
- Monitoring dashboard and reports

Tasks Scheduler & Resources manager

- Integration with grid
- Indicator on running jobs
- Hypervisor & VM management
UC 3: Acceleration of Financial Valuations

C++ library developed by Pricing Partners
Pricing solution dedicated to highly complex derivatives,
Greek computation
How Does it Work? Price-it Computing Distribution

- Price-it Excel
- ProActive Scheduler
- Regular Price-it Excel Interface
- Pool of shared resources
- Automatic execution via job scheduler
Increased Productivity: Reduces Price-it Execution Time by 6 or more!

Use Case: Bermuda Vanilla, Model American MC

Test conditions:
- One computation is split in 130 tasks that are distributed
- Each task uses 300ko

More than 3 times faster with only 4 nodes!

Even 6 times faster with 9 nodes!
Use Case 4: OMD2
Distributed Multi-Disciplinary Optimizations
Coupling Mechanics, Aerodynamics ...

3D Air Conditioning

10min CPU

2D Air Conditioning

<1min CPU

100h CPU

Cylinder Head

External Aerodynamic

1000h CPU
ProActive OMD2 Demo

1000 Cores
Production Cloud Portal
Use Case 4-Bis: Hydrodynamic with K-Epsilon and FineMarine
Hydrodynamic Optimization: Workflow generated from a GUI

ProActive Studio ➞ Graphical Workflow Editor
Real Live Demo: CFD Distributed // Workflow
Hydrodynamic Optimization: Execution
Hydrodynamic: Remote Steering during execution
Conclusions
Conclusion: Technology Preview

- ProActive Fine Grain CLOUD management:
  ➡ Pricing at the second (like GSM)

- Open Source Cloudware Initiative (OSCi)

➡ Elastic Clouds

http://ProActive.inria.fr
### Industrial (1750) & Cloud Revolution Compared

<table>
<thead>
<tr>
<th></th>
<th>Industrial Revolution</th>
<th>Cloud Revolution</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Concept</strong></td>
<td>Mechanization and centralization of manufacturing activities</td>
<td>Computing as a Utility Centralization of Data Center</td>
</tr>
<tr>
<td><strong>Technology</strong></td>
<td>Supporting new technos (Mechanic, Tool Machines, etc.)</td>
<td>Distributed Computing Virtualization Multi-Cores Network</td>
</tr>
<tr>
<td><strong>Socio Economical Factors</strong></td>
<td>Large new demand was ready to use the new offer. (A change in business attitude &amp; organization)</td>
<td>IT Cost Reduction Pressure CIO Nightmare CEO Out-of-DataCenter CapEx</td>
</tr>
</tbody>
</table>

→ All elements converge for a strong Cloud Revolution

Sources & Inspiration: Simon Wardley (CSC) Scott Stewart
Thank you for your attention!
http://proactive.inria.fr