

Arquitecturas Paralelas I

Computação Paralela



João Luís Ferreira Sobral
www.di.uminho.pt/~jls
jls@...

<http://gec.di.uminho.pt/lesi/ap1>

Computação paralela

Resultados da Aprendizagem

- Desenvolver aplicações que executem de forma eficiente numa gama alargada de plataformas, desde máquinas com hyperthreading, mult-core, multiprocessador a clusters.

Programa (resumido)

- **Arquitectura dos sistemas de computação**
 - sistemas de memória partilhada e sistemas de memória distribuída.
 - introdução à GRID (rede mundial de sistema de computação).
- **Modelos de programação paralela e linguagens**
 - processos comunicantes e objectos activos.
 - distribuição de componentes pelos recursos computacionais
 - mecanismos de especificação de concorrência/paralelismo
- **Projecto de aplicações paralelas**
 - gestão eficiente da distribuição de componentes; medição, análise e optimização do desempenho
 - relação custo/benefício de concorrência/paralelismo e sua quantificação (métricas);
 - adaptação da granularidade da computação/comunicação;
 - estratégias de distribuição da carga de dados/computação/comunicação.
- **Análise de ambientes comerciais/tecnologias**
 - Software de gestão de "GRIDs" (*Torque*)
 - Desenvolvimento de aplicações empresariais em *clusters* de máquinas (J2EE em ambiente de *clustering*)

Computação paralela

Planeamento

Aula	Matéria Teórica	Tecnologia
1 (27/09)	- Introdução às arquitecturas paralelas/Grid	-
2 (04/10)	- Revisão de conceitos de programação	Programação em Java
3 (11/10)	- <i>Middleware</i> para suporte a distribuição	Java RMI/.Net Remoting (em C#)
4 (18/10)	- Programação OO concorrente	<i>Threads</i> em C#/ Java
5 (25/10)	- Metodologias de desenvolvimento de aplicações concorrentes/distribuídas/paralelas	Desenvolvimento de algoritmos em C# + Remoting / Java
6 (01/11)	- Medição e optimização de desempenho	Análise de algoritmos em Java
7 (08/11)	- Desenvolvimento de aplicações paralelas baseadas em passagem de mensagens	Programação em MPI (<i>Message Passing Interface</i>) e C++
8 (15/11)	OpenMP	-
9 (22/11)	- Análise de sistemas de <i>Grid</i>	<i>Torque</i>
10 (29/11)	- Aplicações empresariais em <i>Clusters</i>	<i>J2EE</i>
11 (06/12)	- Projecto PPC-VM	-
12 (13/12)	Folga	-

Computação Paralela

Avaliação

- Um trabalho prático (desenvolvimento aplicação para um ambiente de *cluster*)

Pré-requisitos

- Conhecimentos de Java e de C

Bibliografia (Base)

- Acetatos
- I. Foster. *Designing and Building Parallel Programs*, Addison-Wesley, 1995.
- D. Lea. *Concurrent Programming in Java: Design Principles and Patterns*, Addison-Wesley, 2000.
- G. Andrews. *Multithreaded, Parallel and Distributed Programming*, Addison-Wesley, 2000.

Bibliografia (Adicional)

- R. Gerber, A. Binstock. *Programming with Hyper-Threading Technology*, Intel Press, 2004.
- F. Buschmann, D. Schmidt, M. Stal, H. Rohnert, *Pattern-oriented Software Architecture Vol 2: Patterns for Concurrent and Networked Objects*, John Wiley and Sons Ltd, 2000.
- T. Christopher, G. Thiruvathukal, *High-performance Java platform computing*, Prentice Hall, 2000.
- I. Foster, C. Kesselman (ed). *The Grid2*, Morgan Kaufmann, 2004.

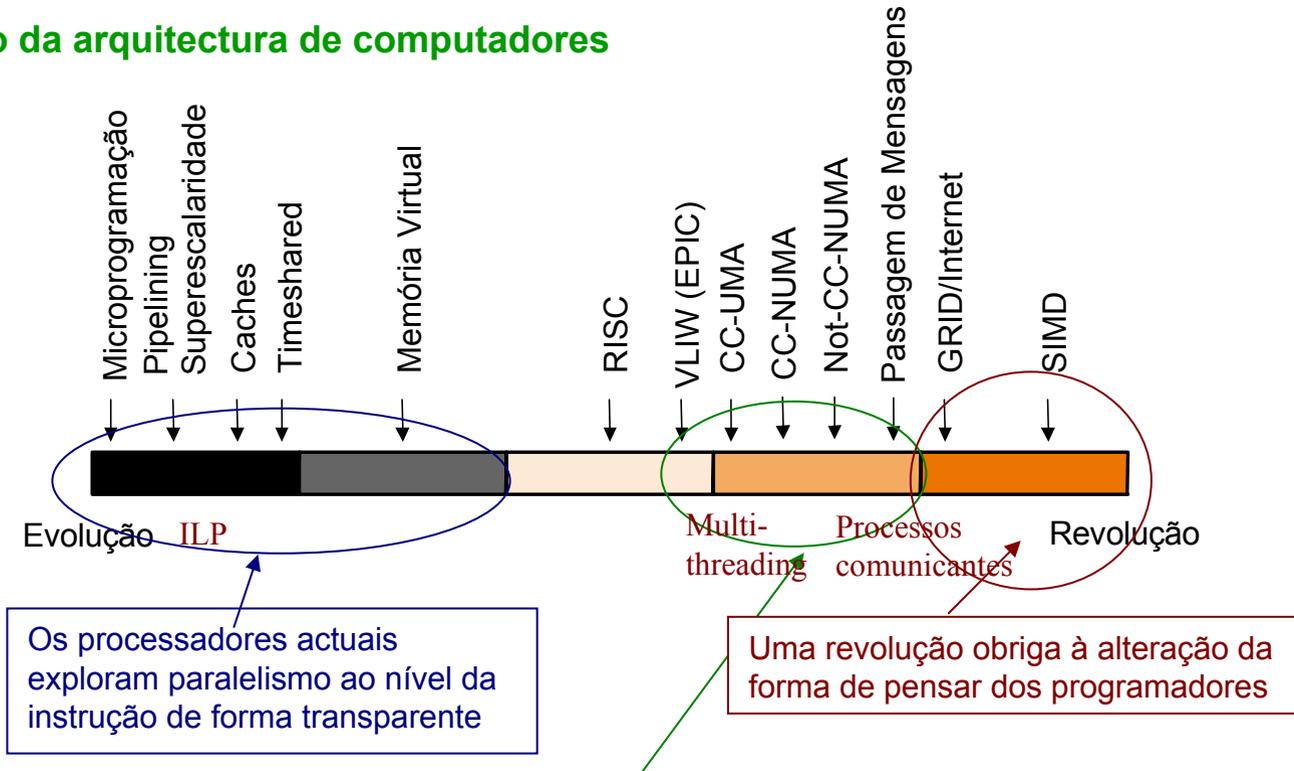
Computação Paralela

www.top500.org (Jun/2005)

Rank	Site Country/Year	Computer / Processors Manufacturer	R_{max} R_{peak}
1	DOE/NNSA/LLNL United States/2005	<i>BlueGene/L</i> eServer Blue Gene Solution / 65536 IBM	136800 183500
2	IBM Thomas J. Watson Research Center United States/2005	<i>BGW</i> eServer Blue Gene Solution / 40960 IBM	91290 114688
3	NASA/Ames Research Center/NAS United States/2004	<i>Columbia</i> SGI Altix 1.5 GHz, Voltaire Infiniband / 10160 SGI	51870 60960
4	The Earth Simulator Center Japan/2002	Earth-Simulator / 5120 NEC	35860 40960
5	Barcelona Supercomputer Center Spain/2005	<i>MareNostrum</i> JS20 Cluster, PPC 970, 2.2 GHz, Myrinet / 4800 IBM	27910 42144
6	ASTRON/University Groningen Netherlands/2005	eServer Blue Gene Solution / 12288 IBM	27450 34406.4
7	Lawrence Livermore National Laboratory United States/2004	<i>Thunder</i> Intel Itanium2 Tiger4 1.4GHz - Quadrics / 4096 California Digital Corporation	19940 22938
8	Computational Biology Research Center, AIST Japan/2005	<i>Blue Protein</i> eServer Blue Gene Solution / 8192 IBM	18200 22937.6
9	Ecole Polytechnique Federale de Lausanne Switzerland/2005	eServer Blue Gene Solution / 8192 IBM	18200 22937.6
10	Sandia National Laboratories United States/2005	Red Storm, Cray XT3, 2.0 GHz / 5000 Cray Inc.	15250 20000

Computação Paralela

Futuro da arquitectura de computadores

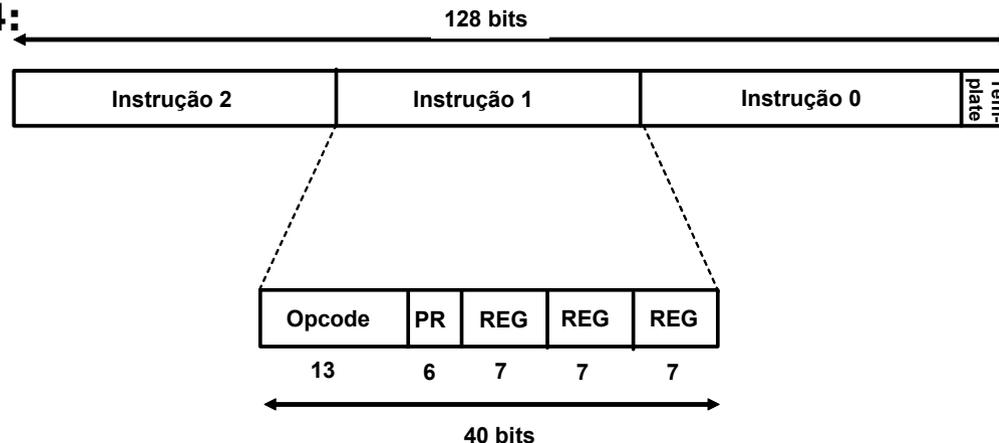


A ênfase de computação paralela é na programação deste tipo de arquitecturas

Computação Paralela

VLIW (Very Long Instruction Word)

- ❑ O escalonamento dinâmico incrementa de forma considerável a complexidade do Hardware.
- ❑ VLIW efectua um escalonamento estático, sendo o compilador responsável por indicar as instruções que podem ser realizadas em paralelo.
=> tem-se revelado um fracasso
- ❑ O formato de instrução indica as operações que são realizadas em paralelo por cada unidade funcional.
- ❑ Exemplo IA-64:



Computação Paralela

VLIW (cont.)

□ Limitações de VLIW

- O código gerado tende a ser de maior dimensão, porque é necessário inserir *nop* nos campos da instrução não preenchidos.
- Compatibilidade de código entre gerações dos mesmo processador uma vez que tende a expor a arquitectura interna do processador
- É mais penalizado com *stalls* que o escalonamento dinâmico

□ EPIC – IA-64 / Itanium

- 64 registos de inteiros + 64 registos FP, ambos com 64 bits
- 3 instruções em 128 bits (LIW?)
 - menos bits que VLIW clássico, produzindo código mais compacto
 - possibilidade de ligação entre os vários grupos de instruções
- Verificação de dependências em HW => compatibilidade de código

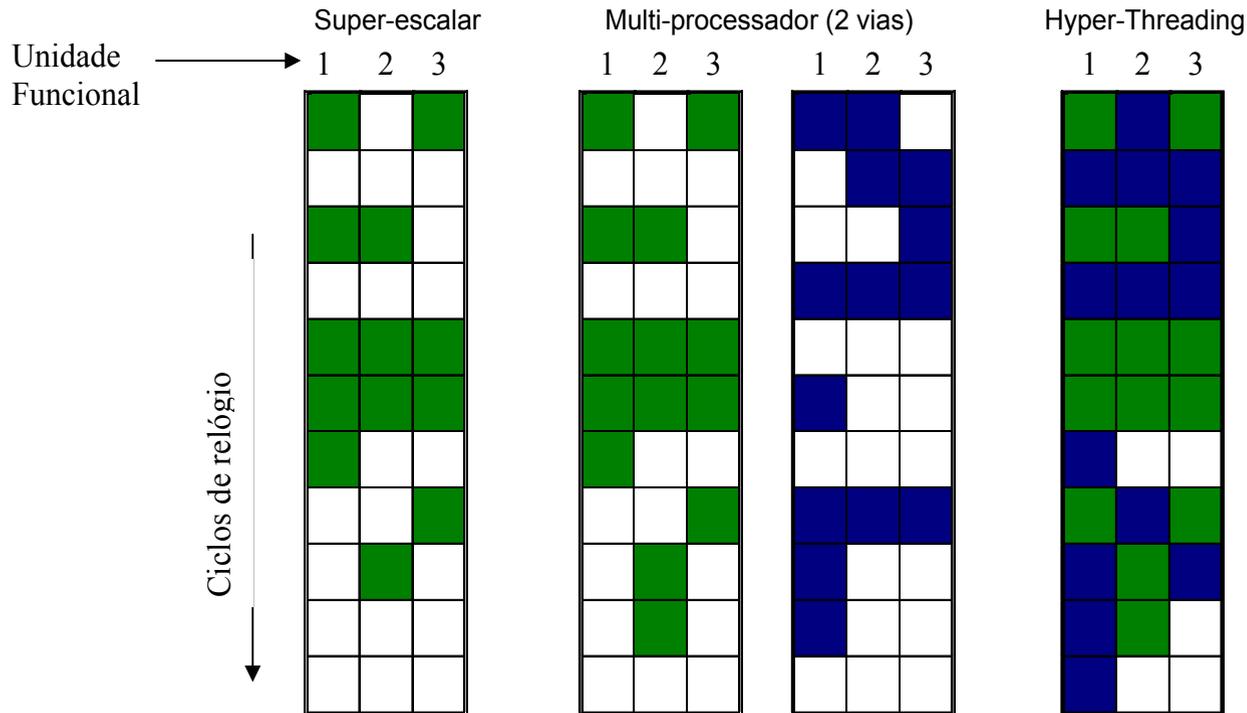
Computação Paralela

Hyper-Threading (Intel)

- ❑ **A adição de mais unidades funcionais nos processadores actuais não conduz a ganhos significativos de desempenho.**
- ❑ **O processador tem a capacidade de simular vários processadores (CC-NUMA) ou pode executar vários fios de execução em simultâneo.**
 - **baseia-se mais no paralelismo ao nível dos fios de execução ou dos processos e menos paralelismo ao nível da instrução.**
- ❑ **Implica um aumento de <5% de transístores por processador e pode conduzir a ganhos no desempenho até 35% (Xeon actuais, com grau 2).**
- ❑ **Os ganhos de desempenho são proporcionados por uma utilização mais eficiente dos recursos**

Computação Paralela

Hyper-Threading (cont.)



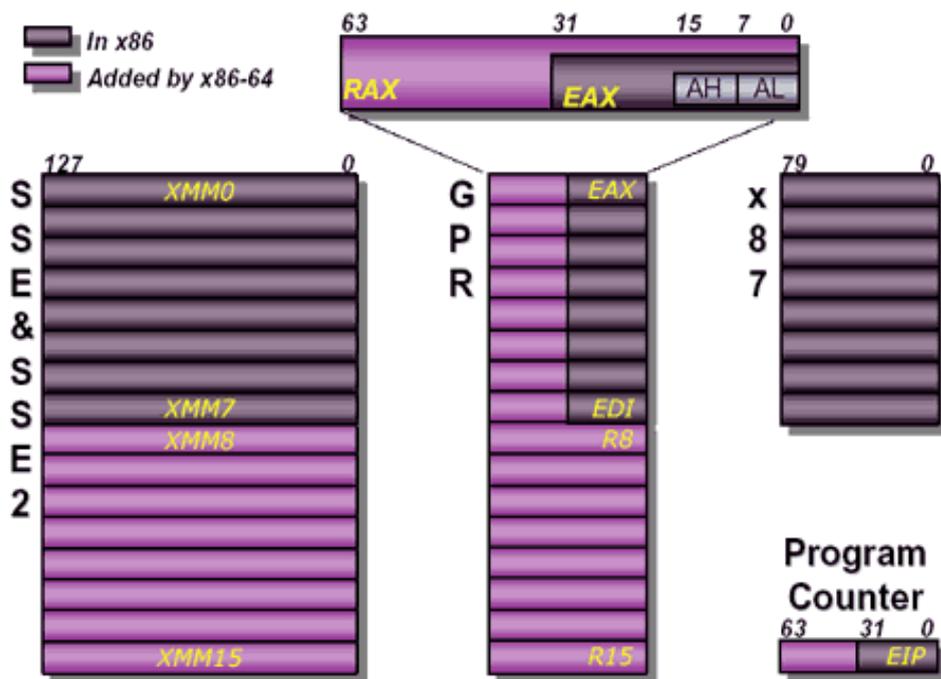
1 processador = 2 processadores ?

Não porque parte dos recursos do processador não são duplicados (i.é., são divididos pelos vários fios de execução) (*caches*, registos internos, *buffers* internos, etc.)

Computação Paralela

Hammer (AMD com 64 bits)

- Estende o IA-32 para 64 bits:

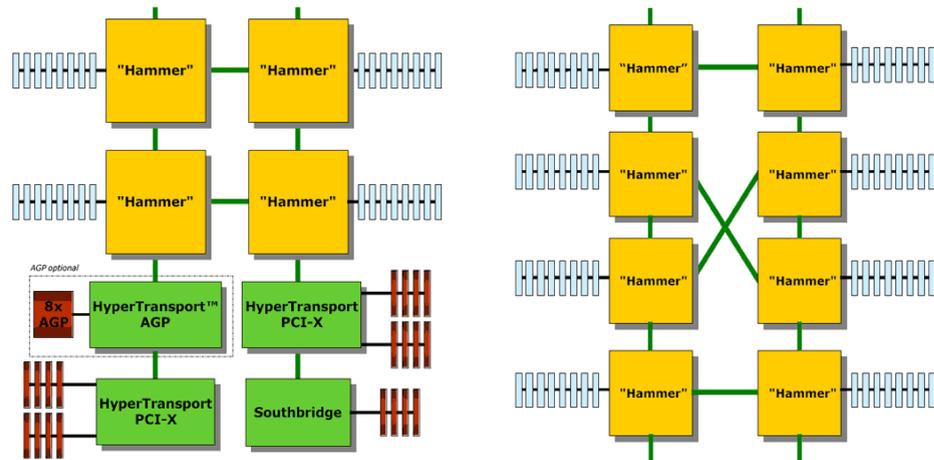


- A *pipeline* possui mais dois estágios (12 contra 10) que a arquitectura K7 (Athlon XP)

Computação Paralela

Hammer (cont.)

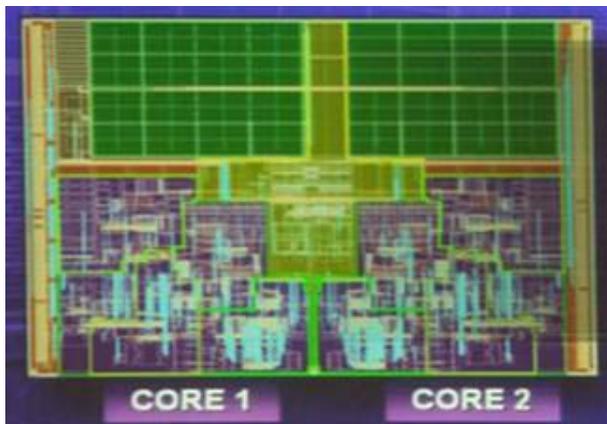
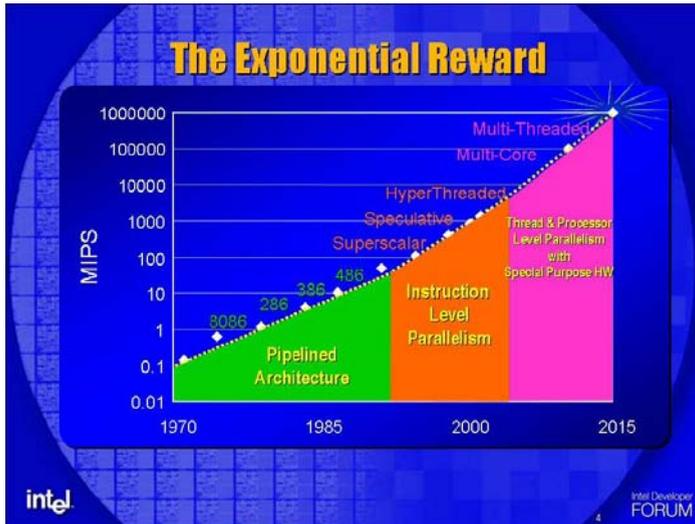
- ❑ Integra o controlador de memória do próprio chip e 2 ligações externas (Hyper Transport) para ligação a outros processadores.
- ❑ O Hyper Transport permite a ligação até 8 processadores sem recurso a componentes adicionais:



- ❑ A configuração multiprocessador assemelha-se mais ao tipo CC-NUMA, uma vez que cada processador possui o seu controlador de memória e a largura de banda disponível para a memória local é superior à disponibilizada pelo Hyper Transport, embora a velocidade do Hyper Transport escale com a velocidade do processador

Computação Paralela

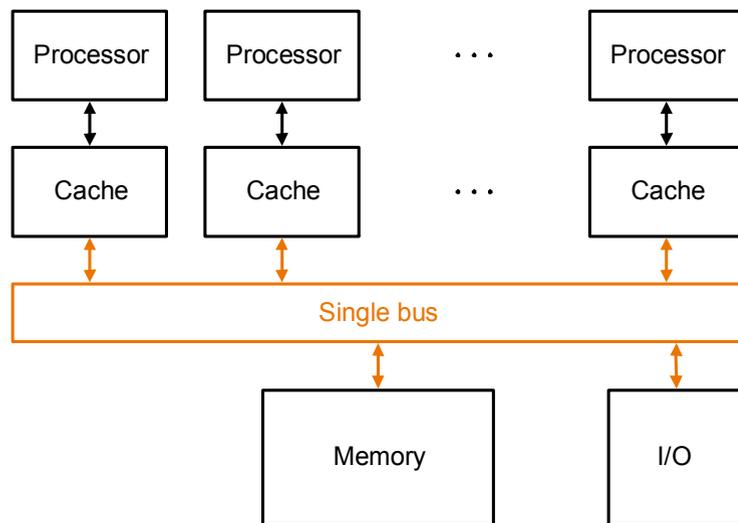
Yonah (centrino dual-core)



Computação Paralela

Memória partilhada centralizada (CC-UMA, CC-NUMA)

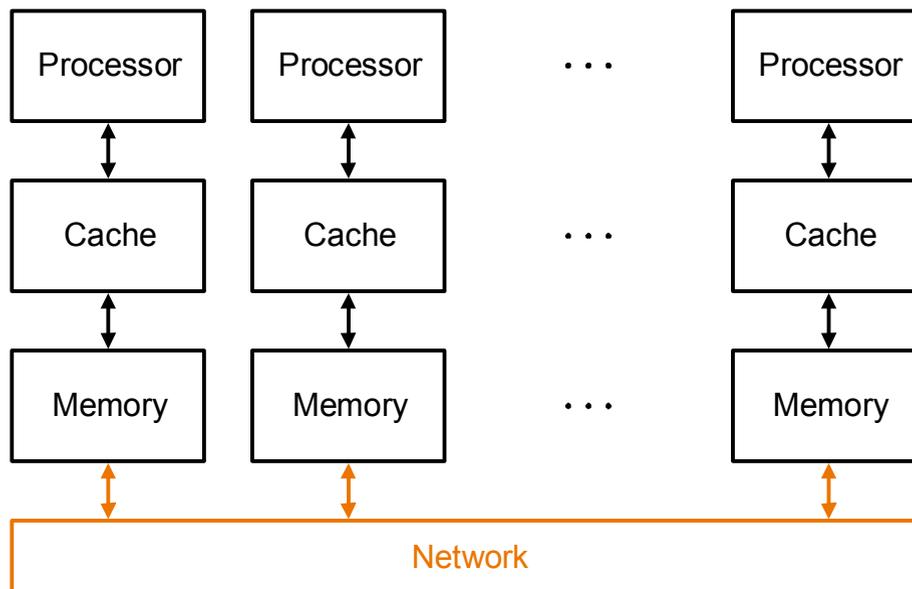
- ❑ Vários processadores partilham um barramento de acesso à memória
- ❑ As *caches* de cada processador contribuem para reduzir o tráfego no barramento e a latência dos acessos à memória
- ❑ um valor pode estar replicado em vários sítios => são necessários mecanismos para assegurar a coesão entre as *caches* dos vários processadores e a memória
- ❑ A largura de banda de acesso à memória é partilhada pelos vários processadores => limitação à escalabilidade deste tipo de arquitectura:



Computação Paralela

Memória distribuída – processadores conectados por uma rede de interligação (CC-NUMA, passagem de mensagens)

- Os sistemas interligados por um barramento tendem a limitar o número de processadores que efectivamente podem ser ligados
- A alternativa reside na utilização de uma rede dedicada à interligação dos vários processadores, possuindo cada processador a sua memória dedicada



Computação Paralela

Cluster de máquinas

- ❑ **Constituídos pode HW "normal", interligados por uma rede de alta velocidade (Gbit/s com baixa latência)**
- ❑ **Cada nodo de processamento pode ser uma máquina de memória partilhada com vários processadores**
- ❑ **Cada nodo possui uma cópia do SO**
- ❑ **Alta disponibilidade: quando falha uma máquina basta substituí-la por outra**
- ❑ **Custos de administração na ordem do número de máquinas**
- ❑ **Exemplo (Cluster investigação da Universidade do Minho - 2005):**
 - **8 nodos com SAN FC 3TB**
 - **8 nodos com NVIDIA 7800 GTX**

CPU	Processador	Freq. Relógio	Memória Actual/Máxima	Comunicação	Nodos	Máximo de nodos
8	AMD Athlon MP	1,5 GHz	4GB/16 GB	250 MB/s, 10us (2,0 Gbit/s)	2-way	4
14	Intel Xeon	3,2 GHz	10 GB / 112 GB	250 MB/s, 10us (2,0 Gbit/s)	2-way	7
96	Intel Xeon	3,0 GHz	96GB / 768GB	1 GB/s, 3us (10,0 Gbit/s)	2-way	48

Computação Paralela

Cluster front-end

plutao.di.uminho.pt/pe254 Celeron 466 MHz 256MB, 6.4 GB (Linux)

— Plutao (193.136.19.48)

Rede DI (100 Mbit)
193.136.19.xxx

8 x Pentium II, 350/400 MHz, 256 MB, 3.2GB (Linux/Windows 2000)

4 x Bi-Athlon MP 1800+ 512MB, 60 GB (Linux/Windows 2000)



Pe254 (10.0.10.254)

Pe1
(10.0.10.1)

Pe8 (10.0.10.8)

Pe9 (10.0.10.9)

Pe12 (10.0.10.12)



Switch 16 Portas
10/100 Mbit

Pm9
(192.168.3.9)

Pm12
(192.168.3.12)



Switch Myrinet 2.0 Gbit/s
8 portas Fibra óptica



Switch Myrinet 2.0 Gbit/s
8 portas SAN

Pm1 (192.168.3.1)

Rede Myrinet (2.0/1.2 Gbit)

192.168.3.xxx

Pe0 (10.0.10.253)

Cluster Server (Linux+DHCP+NFS+NIS)

Rede interna (100 Mbit)
10.0.10.xxx

Athlon XP 2000+
512MB, 80 GB

Pm0
(192.168.3.253)

Cluster do GEC

João Luís Sobral
Janeiro 2004

gecinv.di.uminho.pt

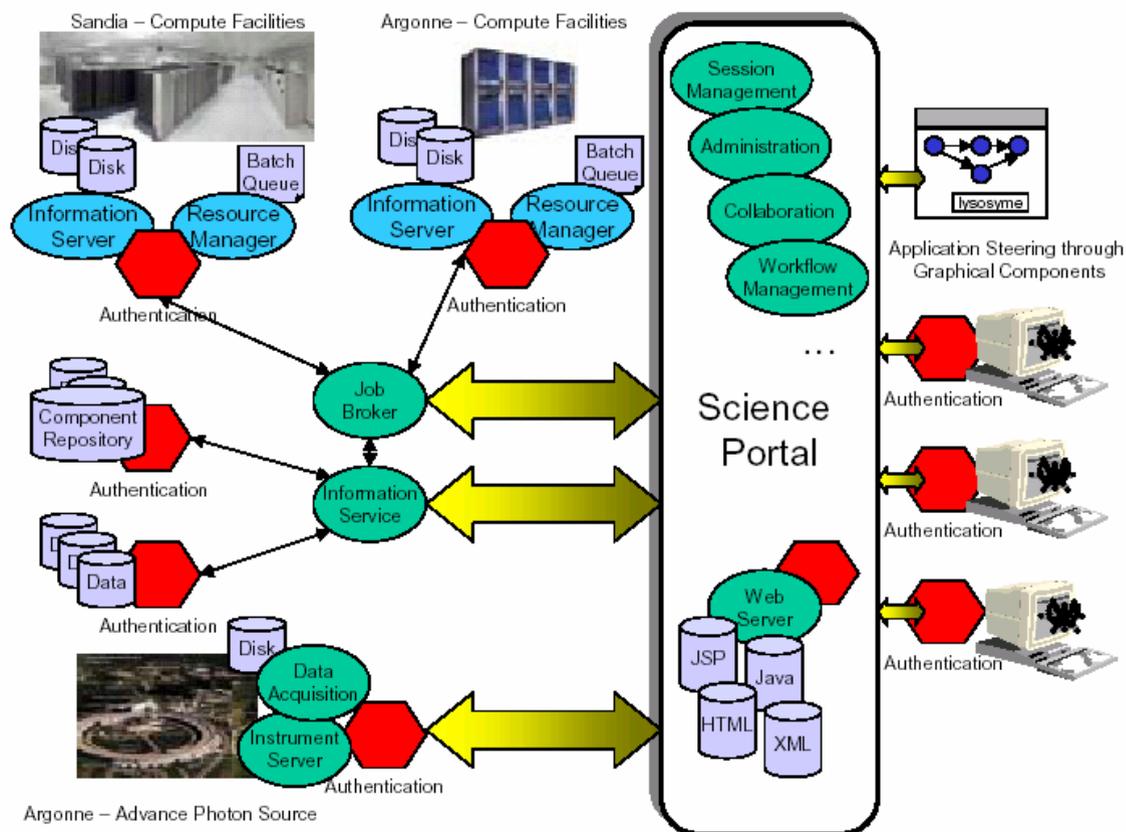
Computação Paralela

GRID

- ❑ **Pode ser a próxima geração da internet**
- ❑ **"A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities.", Foster 1998.**
- ❑ **"coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations.", Foster and AI, 2000.**
- ❑ **A partilha de recursos é altamente controlada, com os fornecedores e os utilizadores dos recursos a definirem claramente o que é partilhado, que está autorizada a utilizar cada recurso e em que condições a partilha ocorre.**
- ❑ **a coordenação de recursos não deve ser centralizada (os vários recursos encontram-se em domínios diferentes)**
- ❑ **deve utilizar protocolos standard e abertos**
- ❑ **deve fornecer vários níveis, não triviais, de serviço**
- ❑ **requer novos paradigmas de programação? (os recursos dinâmicos, altamente heterogéneos e sujeitos a grandes variações de desempenho)**
- ❑ **pode tornar, a longo prazo, os supercomputadores dispensáveis?**

Computação paralela

Aplicações de GRID



Computação paralela

Tecnologias para desenvolvimento de aplicações em GRID

