Grid Computing

L'informatica come public utility del calcolo: oltre il web, verso un moderno modello economico di servizio

dr. Antonio Candiello, PhD
Business Solutions & Responsabile Qualità Met Sogeda

Introduzione

La necessità di una nuova tecnologia informatica in grado di condividere consistenti risorse di calcolo indipendentemente dalla loro localizzazione geografica è conseguenza del processo di trasformazione del grande acceleratore di particelle europeo presente al CERN (Conseil Européen pour la Recherche Nucléaire). Il progetto che ha promosso un primo sviluppo in Europa di tale obiettivo è noto con il nome di *Data Grid*. Data Grid, finanziato con i fondi del V Framework Programme dalla Comunità Europea, attivo nel triennio 2001-2003, ha visto impegnati come principali partner il CERN, il CNRS francese, l'ESA, l'INFN italiano, il NIKHEF olandese, il PPARK inglese.
L'infrastruttura sviluppata da Data Grid è stata oggetto di dimostrazioni nel corso del 2003, che hanno evidenziato il grado di diffusione e di affidabilità raggiunta. In una di queste dimostrazioni il *middleware* EDG si è rivelato in grado di gestire un sistema composto di un migliaio di processori, 10-20 Tbytes di storage distribuito in 12 siti in cinque paesi diversi, utilizzato da 12 *Virtual Organizations* (con qualche VO composta di oltre 100 utenti attivi).
Il progetto Data Grid è ora concluso. La Comunità Europea ha deciso di continuare ad investire su queste architetture informatiche e di passare dallo sviluppo della tecnologia alla stabilizzazione dell'infrastruttura come strumento affidabile e disponibile per la fornitura di servizi al mondo scientifico europeo. Il relativo progetto è denominato Enabling Grid for E-science in Europe (EGEE): avrà il suo avvio ad aprile 2004 e durerà due anni.
Il progetto EGEE intende creare un'infrastruttura europea per i servizi computazionali in tecnologia Grid Computing che utilizzi risorse di università, istituti di ricerca ed imprese e le metta a disposizione delle nuove Virtual Organizations di utenti che necessitano di risorse computazionali e di memorizzazione di elevate prestazioni e capacità[1].
Ma non è solo l'Europa ad investire in questa tecnologia perché negli USA la National Science Foundation (NSF) ha finanziato la "Cyberinfrastructure", impegnando notevoli risorse finanziarie. I grandi produttori americani di informatica sono peraltro i principali sostenitori attivi del Grid Computing.

---

[1] Le necessità elaborative connesse all'analisi dei dati relativi agli urti di particelle che verranno generati da LHC dal 2007 sono impressionanti: sono previsti tra 6 e 8 PB/anno (1PB = 1.000.000 Gbytes) che rappresentano i dati dei 100.000.000 eventi/anno di collisioni di particelle; tali dati dovranno poter essere analizzati da migliaia di utenti batch ed interattivi contemporanei.

In questo breve approfondimento descriveremo come si è giunti alle nuove architetture di calcolo distribuito e quali sono le componenti tecnologiche delle Computational Grids. Una breve analisi dei modelli economici sottintesi e dei connessi vantaggi per le aziende e gli utenti ci porterà quindi alle conclusioni.

Dal calcolo parallelo al calcolo distribuito

Il calcolo parallelo è oggetto di studi da quando l'informatica ha superato le iniziali fasi pionieristiche. La possibilità di aggregare la potenza di calcolo è stata frequentemente utilizzata per superare le limitazioni tecnologiche legate alla potenza disponibile nei processori esistenti al momento del bisogno.
Tuttavia, tale aggregazione è sempre stata un'operazione complessa a causa di diversi fattori: innanzitutto per i costi diretti, dal momento che l'utilizzo di *n* processori si traduce in un fattore moltiplicativo *n* del costo unitario, in secondo luogo per le difficoltà connesse alle comunicazioni inter-processore, infine a causa delle attività di programmazione "su misura" che è dipendente dalla specifica architettura.
La convenienza del calcolo parallelo è inoltre stata messa in discussione dalla crescita continua della disponibilità di calcolo unitario garantita dalla legge di Moore[2], che ha sistematicamente ridotto il vantaggio in termini di capacità elaborativa offerta dal calcolo parallelo.
Tale situazione si è recentemente evoluta grazie a diversi fattori.

a) La continua crescita della memoria di massa e della larghezza di banda disponibile ha reso conveniente la condivisione di risorse computazionali non adiacenti. La dimensione media della memoria di massa raddoppia ogni 12 mesi, ad un ritmo superiore alla legge di Moore. In base alla *legge di Gilder* (George Gilder, 1992) la larghezza di banda raddoppia ogni 6 mesi ad un ritmo ancor più rapido.
Nasce quindi la convenienza di utilizzare un modello distribuito di computing che sfrutti questa maggiore convenienza della banda disponibile nei confronti della pura potenza di calcolo e dello storage disponibile. Più avanti si va nel tempo, maggiore è infatti la facilità e la convenienza economica di aggregare e allocare risorse computazionali da diversi settori per costruire una potenza di calcolo combinata virtualmente illimitata.
b) La maturazione dei sistemi software di intercomunicazione tra processi in esecuzione indipendenti dall'hardware utilizzato ha ampliato le possibilità di applicazione del calcolo parallelo. Citiamo ad esempio le tecnologie software PVM (Parallel Virtual Machine) e MPI (Message Passing Interface), con le quali è possibile processare richieste batch aggregando sistemi eterogenei e gestendone la comunicazione. Si tratta comunque di tecnologie non in grado di assicurare un supporto completo da parte del sistema di supervisione e che richiedono una consistente attività operativa all'implementatore.

---

[2] La potenza di calcolo espressa come numero di transistor disponibili nei processori cresce ad un tasso descritto dalla *Legge di Moore*. Gordon Moore ha pronosticato nel 1965 un sostanziale raddoppio del numero di transistor per processore ogni 18-24 mesi circa: tale raddoppio è stato *sistematicamente confermato* dall'evoluzione della tecnologia dei processori. Tale incremento è strettamente connesso alla capacità elaborativa unitaria dei calcolatori attuali.

c) La diffusione capillare e mondiale dell'interconnessione standard garantita dai protocolli tcp/ip di internet ha di fatto reso possibile per la prima volta definire architetture globalmente distribuite e utilizzabili a livello mondiale con protocolli comuni. Ad esempio, in concomitanza con lo sviluppo delle reti P2P (Peer To Peer) di condivisione / file sharing, il cui fenomeno è noto soprattutto per il successo di Napster e dell'interscambio di file musicali, sono stati sviluppati dei protocolli di condivisione della potenza di calcolo, di cui il miglior esempio è rappresentato da Seti@Home.
La traduzione commerciale di tale fenomeno è rappresentato dalle *P2P Grid*, ed è avvenuta ad opera di aziende specializzate che, distribuendo nei desktop agenti specializzati "dormienti", sono in grado di trasformare la potenza inutilizzata di migliaia di PC desktop in un aggregato con la potenza di calcolo di un costoso sistema dedicato. I nomi più noti sono United Devices, Entropia, Platform. Si tratta della "prima via" del *Grid Computing*, di cui segnaliamo i punti deboli: appesantimento dei desktop, frequente indisponibilità delle risorse hardware, difficoltà di applicazione del modello alle risorse mobili, una certa ritrosia degli utenti finali nel garantire accesso al proprio PC.
d) L'introduzione del sistema operativo Linux e la nascita della comunità mondiale Open Source ha fornito l'elemento fondante sul quale costruire l'impianto complessivo di un nuovo modello di *distributed computing*.
e) La recente massiva capacità elaborativa richiesta dalla comunità scientifica (per i rilevatori di particelle, la biogenetica, le osservazioni terrestri, le scienze mediche), ha giustificato i forti investimenti necessari per la definizione di tale modello di distributed computing nell'ambito della costruzione di una nuova infrastruttura di calcolo.

Il processo di maturazione delle tecnologie di calcolo parallelo è giunto così alle *Computational Grids* nel momento in cui la necessità di costruire una infrastruttura affidabile, altamente scalabile e sicura in grado di supportare un modello di servizio a consumo di tipo industriale, ha comportato la scelta di allocare le risorse di calcolo e di storage in *farm* dedicate.

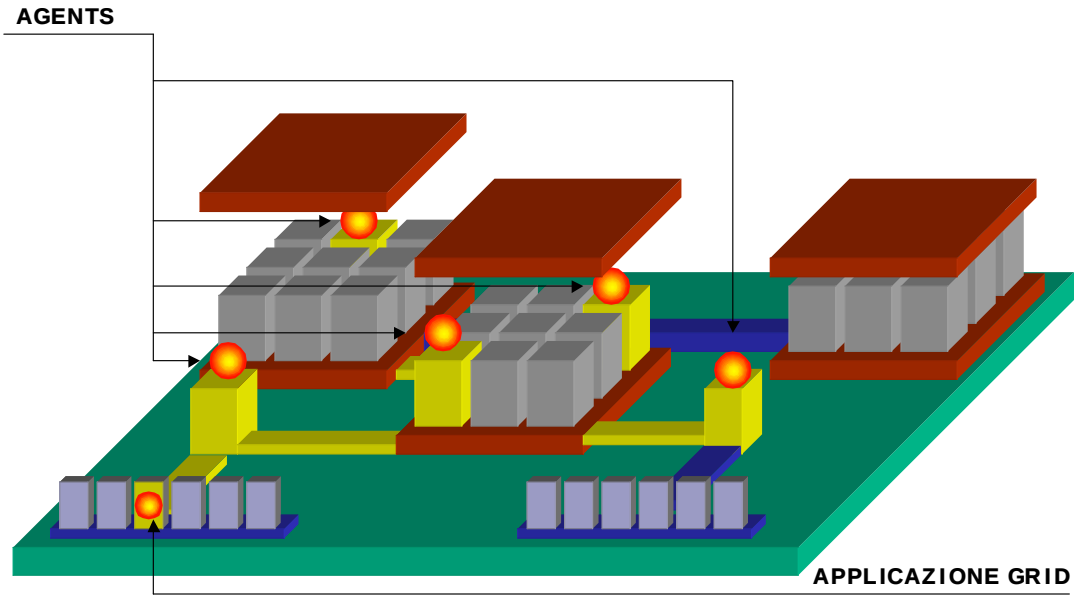

Figura 1: una rappresentazione grafica di utilizzo dell'infrastruttura Grid Computing

Dalle Power Grids alle Computational Grids

Il concetto di *computational grids* e di *data grids* è sorto nei primi anni '90, mutuando il concetto di *grid* da un settore industriale ad elevata maturità: quello dell'energia elettrica (*power grids)*. La rete di distribuzione dell'energia elettrica fornisce infatti la potenza in una modalità diffusa e standardizzata: è sufficiente un dispositivo di voltaggio compatibile e con la spina adatta per poter utilizzare le prese di corrente disponibili; al consumatore non interessa sapere dove è stata prodotta e come è stata distribuita l'energia elettrica che consuma, gli basta poterla utilizzare.

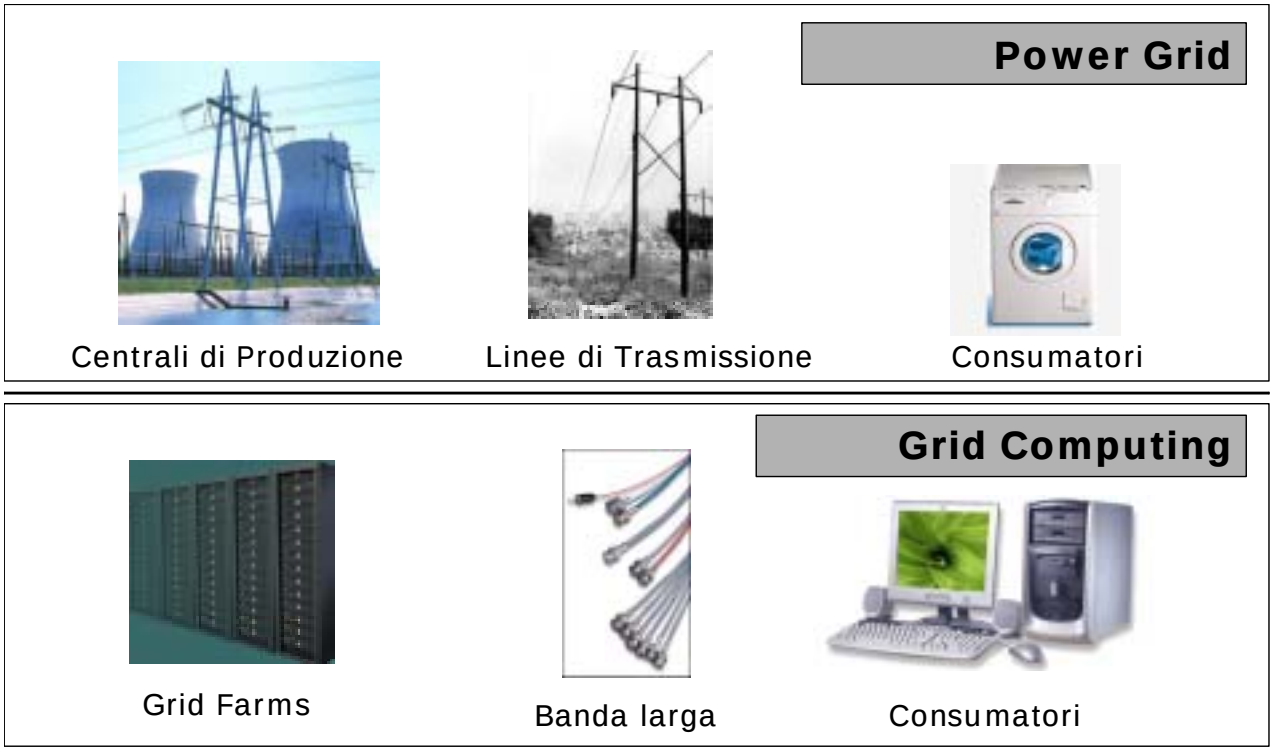

Figura 2: la metafora dell'energia elettrica applicata all'informatica

Come Ian Foster e Carl Kesselman per primi hanno illustrato, la situazione è oggi per l'informatica come nei primi anni del secolo scorso per l'elettricità: in assenza di una rete (*grid*) di distribuzione della potenza, nonostante si disponga della capacità di costruire dispositivi di consumo della stessa, è necessario dotarsi ogni volta di un generatore di potenza (un server, nella terminologia informatica). E, nonostante si disponga da alcuni anni di un ampio sistema di interconnessione (internet), al quale sono collegati milioni di dispositivi di calcolo e di memorizzazione, si sente ora la necessità di trasformare tale sistema in una vera e propria *infrastruttura* in grado di fornire una modalità di accesso standard a tali risorse.

Computational Grids

In completa analogia con il modello di produzione, distribuzione e consumo dell'energia elettrica, l'obiettivo delle computational grids è quello di permettere all'utente di effettuare la richiesta di elaborazione con specifici

requisiti di alto livello, tramite opportune interfacce, disinteressandosi di dove il sistema vada a reperire le effettive risorse necessarie. Il sistema provvederà a gestire la richiesta, a monitorare il processo in esecuzione, a notificare all'utente quando i risultati saranno disponibili, e quindi a fornirli.
Ecco la definizione di Foster e Kasselman di computational grid:

> "*una infrastruttura hardware e software in grado di fornire accesso differenziabile, consistente, pervasivo, ed economico a risorse computazionali di fascia elevata*".

Tale infrastruttura è composta sia di hardware che di software.
La componente *hardware* è rappresentata da sistemi dotati di potenza elaborativa (cpu), risorse di storage (dischi rigidi) e capacità di interconnessione. La capacità di gestire più sistemi come un unico insieme di risorse permette di utilizzare hardware di basso costo dotato di processori e periferiche standard (tipicamente processori x86 e dischi con interfaccia eide). Il sistema operativo che governa localmente i sistemi è generalmente Linux; in alternativa sono utilizzate le varianti proprietarie di Unix.
La componente *software* è quella dove sono stati concentrati i maggiori sforzi implementativi. Il consorzio internazionale Globus guidato da Ian Foster si è posto come obiettivo quello di costruire un *kit* di abilitazione di base delle Griglie Computazionali per la gestione delle risorse, dei dati e degli accessi al sistema, di attivare gli appropriati ambienti di sviluppo, di costruire i servizi informativi di supporto, e di consentire la gestione della sicurezza del sistema nel suo complesso. Il *Globus Toolkit* prodotto dal consorzio è il nucleo software di Grid Computing che va installato di sopra del sistema operativo di base su ogni macchina che farà parte della grid; tale nucleo abilita diverse funzionalità di calcolo distribuito nella forma dei seguenti servizi disponibili:

- Allocazione delle risorse e process management (il Globus Resource Allocation Manager – GRAM per supportare la sottomissione remota dei job, ed il Dynamically Updated Request Online Co-allocator - DUROC),
- servizi di comunicazione (Globus I/O, Nexus, Grid-Enabled MPI - MPICH-G),
- accesso distribuito informazioni di struttura e status (Monitoring and Discovery Service - MDS),
- servizi di sicurezza e di autenticazione (Grid Security Infrastructure - GSI, realizzato tramite crittografia a chiave pubblica, certificati X.509 ed il protocollo SSL),
- monitoraggio (Hearth Beat Monitor – HBM, per la rilevazione dei malfunzionamenti),
- accesso remoto ai dati (Globus Access to Secondary Storage - GASS, Grid File Transfer Protocol - GridFTP, una estensione del popolare protocollo FTP dotato di maggiore affidabilità, sicurezza e performance),
- gestione degli eseguibili (Globus Execution Management - GEM).

Nella sua ultima edizione il Globus Toolkit è appoggiato ad un framework comune costruito tramite i Web Services, per i quali è stato definito il nuovo standard OGSA (Open Grid Services Architecture).

Diverse collaborazioni internazionali hanno quindi lavorato per estendere il Globus Toolkit allo scopo di poter gestire sistemi di produzione in condizioni di funzionamento realistiche. Il middleware costituito da tali estensioni deve fornire i servizi essenziali sia per facilitare agli utenti la gestione delle applicazioni distribuite che per garantirne il funzionamento anche in caso di rotture hardware o malfunzionamenti degli elementi costituitivi la grid. Il middleware European Data Grid (EDG) sviluppato nell'ambito del progetto europeo presenta al riguardo un notevole grado di maturità.

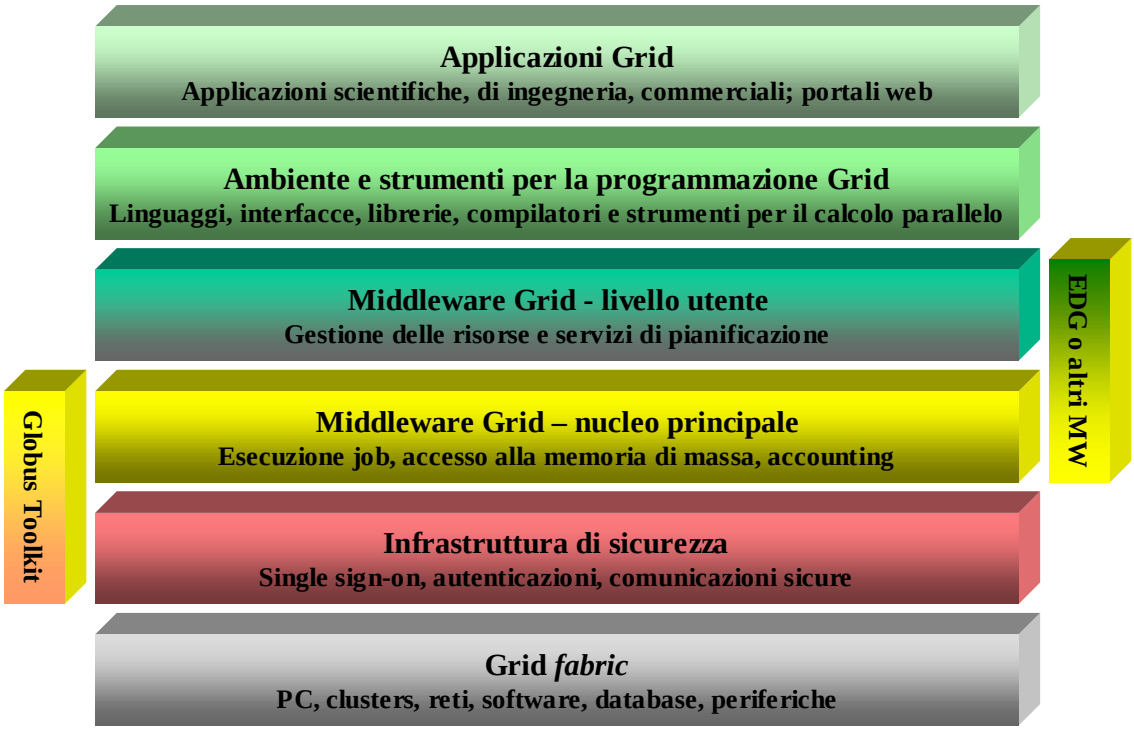

Figura 3: una visione a strati dell'architettura Grid Computing

Va sottolineato che sia il middleware EDG che il Globus Toolkit sono software di pubblico dominio, *open source*, come lo è il sistema operativo preferito nelle implementazioni, ovvero Linux. L'intento delle organizzazioni di ricerca è infatti di garantire la massima diffusione della tecnologia appoggiandosi su modelli standard privi dei tipici vincoli proprietari imposti dai fornitori di software. Liberandosi dal vincolo delle licenze software si garantisce una reale scalabilità sul numero di macchine che possono costituire la grid. Moltiplicando il numero di server di una grid costruita con sistemi proprietari cresce infatti linearmente l'eventuale costo di licenza (che è generalmente calcolato sul numero di processori utilizzati), costo che rischia di pregiudicare i vantaggi determinati dalla maggiore disponibilità delle risorse hardware di basso costo.

Le imprese di informatica come *Information Utilities*

Le Computational Grids sono lo strumento abilitante per la trasformazione dell'industria informatica verso il modello delle *Information Utilities*. Secondo questo modello industriale, le imprese informatiche forniranno servizi a consumo in completa analogia con le *Public Utilities* di servizi infrastrutturali (energia, gas, acqua).
I vantaggi sono sia tecnologici che di gestione economica.

a) Dal punto di vista tecnologico, la diffusione del modello *utility* garantisce che gli applicativi nei quali si sono investite risorse siano rapidamente e facilmente trasferibili da un'infrastruttura ad un'altra, che sia proprietaria o fornita da terze parti esterne all'azienda. Le attività di sviluppo e personalizzazione software sono infatti molto onerose, e sono frequenti le attività di trasferimento di applicativi da una piattaforma all'altra, da un sistema operativo all'altro, rendendo vane molte promesse sul riutilizzo del software prodotto. Il trend sempre più rapido di trasformazione dei sistemi operativi e delle piattaforme hardware ha di fatto aggravato il problema, aumentando i costi di gestione degli applicativi, invece di facilitarne la riduzione.
La standardizzazione dell'utilizzo delle risorse elaborative garantito dalle Computational Grids consente invece di produrre applicativi indipendenti dalla specifica tecnologia hardware/software allo stesso modo in cui il protocollo internet tcp/ip ha permesso la standardizzazione delle risorse di interconnessione. Gli investimenti in sviluppo software e personalizzazioni vengono quindi protetti, sono azzerati i costi di trasformazione degli applicativi dovuti a revisioni tecnologiche dell'infrastruttura informatica e ne viene garantita la scalabilità in termini di risorse elaborative impegnate.
b) Dal punto di vista economico, si ha la trasformazione dei costi fissi in costi variabili, con la promessa di una gestione contabile maggiormente semplificata, in linea con quanto avviene per la fornitura dei servizi essenziali. La standardizzazione dell'utilizzo delle risorse di calcolo rende inoltre possibile il raggiungimento di un obiettivo da sempre molto difficile per l'informatica, la saturazione delle risorse disponibili. Grazie alla standardizzazione l'economia di scala conseguente può essere facilmente perseguita da chi fornisce grandi quantità di calcolo. Comunità di aziende, istituzioni ed individui che lavorano per conseguire degli obiettivi condivisi potranno poi utilizzare le Computational Grids per condividere le risorse, i dati e gli strumenti con cui affrontare gli specifici problemi comuni.[3] Gli utilizzatori saltuari – aziende che magari utilizzano in maniera consistente le loro risorse elaborative solo in determinati giorni del mese o della settimana – possono invece trovare convenienza in contratti di servizio nei confronti di fornitori esterni, ottenendo significative economie e nel contempo migliorando i livelli di servizio cui erano abituati.

Il paradigma dell'utility computing quindi ha i suoi termini di riferimento: *standard*, *scalabilità*, *saturazione delle risorse*, *livelli di servizio*, *costi a consumo*. Si tratta di un linguaggio tipico del modello di business delle imprese di servizio, meno noto alle imprese attive nello sviluppo software che sono più abituate a ragionare in termini di commessa e di progetto.
Met Sogeda Spa è avvantaggiata in questo contesto grazie alla recente convergenza che ha visto l'anima "applicativa" (Met), dove il modello prevalente è il progetto, unirsi all'anima "tecnologica" (Sogeda), dove il modello prevalente è il servizio: per questo motivo in azienda sono oggetto di analisi e sperimentazione da un certo tempo le tecnologie abilitanti alle Computational Grids nell'intento di completare l'offerta attuale e di porsi come

[3] Tale condivisione, unita alle necessarie regole che la governano, dà origine a quelle che vengono definite in termini moderni come *Virtual Organizations* (VO).

Information Utility in grado di anticipare il mercato e di soddisfare al meglio le mutate esigenze dei clienti.

Considerazioni conclusive

Il processo di maturazione ed industrializzazione dell'informatica è già in fase avanzata. L'architettura Grid Computing rappresenta lo strumento necessario per catalizzarne la trasformazione finale da industria organizzata secondo i modelli tradizionali di produzione verso il modello delle Information Utilities.
I grandi *market maker* mondiali dell'informatica promuovono con grandi energie e notevoli investimenti questo nuovo modello emergente, fornendo a supporto numerose tecnologie ed adattando il Grid Computing alla propria filosofia aziendale. Tra i produttori di hardware emblematico è il caso di IBM a sostegno dell'*on demand computing*, mentre HP utilizza la metafora dell'*utility computing*; per quanto riguarda i produttori di software, notevole è l'impegno di Sun nel *Grid Engine*, forte il segnale di Oracle con la rev.10g ("g" sta per *grid*) del popolare database e importante l'abilitazione fornita da Sas ai propri sistemi di *business intelligence* al funzionamento coordinato, solo per citare i riferimenti più evidenti.
La tecnologia a supporto delle Computational Grids è ancora in fase di crescita, e comincia solo ora ad essere utilizzabile nel mondo produttivo, ma siamo nella situazione in cui l'infrastruttura di erogazione del servizio sta acquisendo sempre maggiore affidabilità. Non siamo forse così distanti da una situazione nella quale vi sarà totale accessibilità a risorse informatiche virtualmente illimitate e facili da usare.

Riferimenti bibliografici

I. Foster, C. Kesselman (ed), *The Grid: Blueprint for a Future Computing Infrastructure*, Morgan Kaufmann Publishers, USA, 1999;
Fran Barman, Anthony J.G. Hey, Geoffrey C. Fox (ed) *Grid Computing – making the Global Infrastructure a Reality*, Wiley Series in Communications Networking & Distributed Systems, 2003;
I. Foster, C. Kesselman, *The Grid*, 2nd ed. Morgan Kaufmann, 2004;
F. Berman, G. Fox, T. Hey, *Grid Computing*, Wiley, 2003;
I. Foster, C. Kesselman, S. Tuecke, *The Anatomy of the Grid: enabling Scalable Virtual Organizations*, Int'l J. Supercomputer Applications, vol. 15, no. 3, 2001;
Introduction to Grid Computing with Globus, IBM RedBooks, 2002;
Globus Toolkit 3.0 Quick Start, IBM RedBooks, 2003;
R. Buyya, D. Abramson, J. Giddy, *An Economy Grid Architecture for Service-Oriented Grid Computing*, 10th IEEE international Heterogeneus Computing Workshop (HCW), 2001.