



CONFERENCE '06

Sun N1 Grid Engine

l'alternativa open al calcolo distribuito

Gian Luca Farina Perseu
Responsabile Sviluppo Software
adMin Srl



Agenda:

- **Il mondo HPC**
- **Sun N1 Grid Engine**
- **DRMAA**
- **Demo GECO (Open Source)**

adMin System Solutions Srl

Chi è adMin System Solutions Srl:

- Società di servizi informatici nata nel gennaio 2000
- Progetti relativi all'organizzazione informatica in ambito Automotive
- Consulenze sistemistiche in ambito CAE/CAD
- Progetti in collaborazione con fornitori Software/Hardware
- Formazione sistemistica Unix/Linux

Chi è adMin System Solutions Srl:

- Attività relative al supporto sistemistico:
 - Gestione Posti di Lavoro Tecnici
 - Gestione Storage/Backup
 - Gestione Network
 - Gestione Sicurezza
 - Utilizzo privilegiato di software Open Source

Chi è adMin System Solutions Srl:

- Attività relative all'HPC
 - Esperienza 15 anni
 - Installazione e configurazione cluster HPC
 - Gestione e ottimizzazione di server di calcolo SMP (Symmetric Multi Processor)
 - Metodologia raffinata nel tempo

Chi è adMin System Solutions Srl:

- Partner Sun per N1 Grid Engine
- Partner MSC Software per SimManager (prodotto PLM per il CAE e la simulazione)
- Partner per attività di consulenza e installazioni/gestione di cluster basati su processori a 32bit e 64bit Linux o Unix:



Chi è adMin System Solutions Srl:

- adMin e Sun N1 Grid Engine:
 - 4 anni di esperienza
 - gestisce in modo continuativo per 6 clienti circa 250 CPU
 - gestisce presso un cliente, per conto di Sun Microsystem, un cluster di +1000 CPU

Chi è adMin System Solutions Srl:

- Principali clienti:



Chi è adMin System Solutions Srl:

- Dal 2005 sviluppa progetti software nelle aree applicative tipiche dei suoi clienti:
 - CAE
 - CAD
 - HPC
- **Monlic:** Software per il monitoraggio dei Licence Server (FlexLM, Lum, altri proprietari)

Il mondo HPC

(High Performance Computing)

Un po' di terminologia:



- Cluster: insieme di nodi interconnessi
- Nodi: ogni computer facente parte del cluster
- Risorse di Sistema: i vari componenti Hw/Sw che compongono un nodo
- Job: esecuzione di un programma assegnato a uno o più nodi
- Scheduling dei Job: il criterio con il quale un dato job viene assegnato ad un certo nodo/nodi

Cos'è l'HPC ?

- Per High Performance Computing si intende una tipologia di clustering
- Permette l'esecuzione distribuita di processi sui nodi del cluster
- I processi dialogano fra di loro implementando anche il parallelismo
- Realizza nel modo migliore la legge di Amdahl

Altre tipologie di cluster.

Oltre ai cluster per l'HPC si possono identificare altre tipologie di cluster per :

- High Availability:
- Load Balancing:
- Grid Computing:

Altre tipologie di cluster

- High Availability:
 - Garantisce una continuità dell'infrastruttura tramite una disponibilità dei servizi e dei sistemi
 - Implementazione basata sia su software che su hardware
 - Tende a garantire i servizi sempre di più al 100%
 - Bastano minimo due nodi

Altre tipologie di cluster

- Load Balancing:
 - Usato per bilanciare il carico su vari server
 - Il bilanciamento avviene dinamicamente
 - Include caratteristiche di High Availability

Cos'è l'HPC ?

Cosa differenzia un cluster HPC da uno Load Balancing ?

- In un cluster HPC i nodi eseguono in maniera coordinata programmi paralleli che fanno un uso intenso della CPU
- In un cluster Load Balancing i processi non dialogano fra di loro

Altre tipologie di cluster

- **Grid Computing:**
 - Usa le risorse di molti computer separati e collegati in rete fra di loro
 - La rete può essere anche geografica (WAN) o Internet
 - Uno degli esempi più famosi di Grid Computing è il progetto SETI@home

Cos'è l'HPC ?

Cosa differenzia un cluster HPC da uno Grid ?

- La differenza concettuale è minima
- La differenza maggiore è nella localizzazione dei sistemi:
 - HPC: tipicamente i molteplici nodi sono interconnessi attraverso canali ad altissima velocità e bassissima latenza (es. InfiniBand, Myrinet, Quadrics)
 - Grid: i nodi sono distanti fra loro e collegati attraverso Internet. Interconnessione lenta.

Cos'è l'HPC ?

- Quello che rende particolare un cluster HPC è di possedere tutte le caratteristiche atte a permettere un'elaborazione nel più breve tempo possibile.
- Per tale obbiettivo si sono approntati accorgimenti atti a sfruttare il parallelismo di alcuni algoritmi, ovvero la possibilità di applicare contemporaneamente lo stesso algoritmo su dati differenti

Cos'è l'HPC ?

Come si realizza il parallelismo in un cluster HPC ?

- Insieme di nodi Single System Image
- Il singolo job viene eseguito contemporaneamente su più nodi
- Librerie per il calcolo parallelo:
 - MPI (Message Passing Interface)
 - PVM (Parallel Virtual Machine)

Cos'è l'HPC ?

- MPI: Usata dai maggior software vendor
 - Due implementazioni principali:
 - MPICH: <http://www-unix.mcs.anl.gov/mpi/mpich1/>
 - LAM: <http://www.lam-mpi.org/>
- PVM: implementazione poco usata
<http://www.csm.ornl.gov/pvm/>

Cos'è l'HPC ?

Meglio un cluster o un sistema multiCPU SSI ?

Un cluster garantisce:

- Maggior rapporto prestazioni/prezzo
- Scalabilità
- Obsolescenza dell'Hardware
- Disponibilità delle risorse
- Minor Down-Time per manutenzione

Sun N1 Grid Engine

Sun N1 Grid Engine

- Implementa un completo sistema DRM (Distributed Resource Management)
- Completamente gratuito (con supporto a pagamento da parte di Sun)
- Disponibile su <http://gridengine.sunsource.net>
- Binari per i maggiori SO del mercato
- Utilizzato in più di 10.000 installazioni

Sun N1 Grid Engine

- Funzionalità DRM:
 - Ottimizza l'allocazione delle risorse richieste dai job
 - Gestisce un pool di risorse, garantendo un unico punto di accesso alle risorse distribuite del cluster
 - Accetta la sottomissione dei job degli utenti e li schedula sui nodi del cluster

Sun N1 Grid Engine

- Come funziona:

Access Tier



Management Tier



Computer Tier

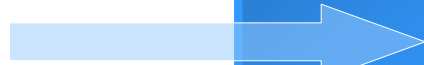


Server Farm

Sun N1 Grid Engine

- Come funziona:

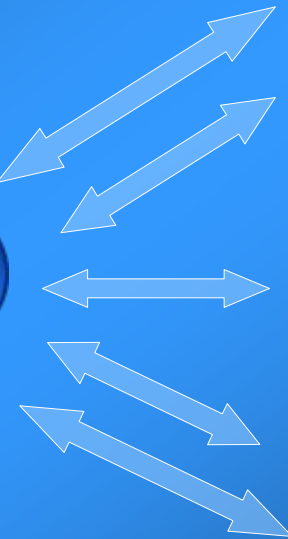
**Sottomissione
(Submit)**



**Distribuzione
(Scheduling)**



**Elaborazione
(Execution)**



Server Farm

Sun N1 Grid Engine

- Come funziona:
 - Accetta le richieste dell'utente (submit job)
 - Tiene in sospeso i job finchè le risorse necessarie all'esecuzione sono disponibili (pending)
 - Invia il job al nodo per l'esecuzione (dispatching)
 - Gestisce l'intero ciclo di vita del job (in run)
 - Informa dello stato dei job e delle informazioni di esecuzione (accounting)

Sun N1 Grid Engine

- L'interazione con N1GE avviene attraverso linea di comando (shell)
 - Comandi principali:
 - `qstat` (lista job e dettaglio)
 - `qsub` (submit di un job)
 - `qalter` (modifica job)
 - `qhost` (stato dei nodi del cluster)

Sun N1 Grid Engine

- Come funziona – qstat:

job-ID	prior	name	user	state	submit/start at	queue	slots	ja-task-ID
647	1.09816	ppp.csh	bianchi	qw	06/16/2006 13:15:34		4	
650	1.09328	ppp.csh	bianchi	qw	06/16/2006 15:21:08		4	
651	1.02661	ppp.csh	bianchi	qw	06/16/2006 15:35:45		2	
658	0.50500	Running	rossi	Eqw	06/16/2006 16:36:15		1	
663	0.50500	N_input_20	rossi	qw	06/16/2006 18:36:17		1	
664	0.50500	N_input_22	rossi	qw	06/16/2006 18:36:35		1	

Sun N1 Grid Engine

- Come funziona – `qstat -j`:

```
job_number:          664
exec_file:           job_scripts/664
submission_time:    Fri Jun 16 18:36:35 2006
owner:              rossi
uid:                5000
group:              fem
gid:                1000
sge_o_home:         /home/col
sge_o_log_name:     col
sge_o_path:         /usr/X11R6/bin:/root/bin
sge_o_shell:        /bin/csh
sge_o_workdir:      /home/spawnarea/f11/513
sge_o_host:         admin2
account:            sge
stderr_path_list:   input.SGE_22133_err
hard_resource_list: naslic=1
mail_list:          col@admin2
notify:             FALSE
job_name:           N_input_22133.job
stdout_path_list:   input.SGE_22133_out
```

Sun N1 Grid Engine

- Come funziona – qhost:

HOSTNAME	ARCH	NCPU	LOAD	MEMTOT	MEMUSE	SWAPTO	SWAPUS
global	-	-	-	-	-	-	-
admin1	1x24-amd64	1	0.07	1001.6M	289.5M	2.0G	92.2M
admin2	1x24-amd64	1	0.02	1001.6M	661.7M	2.0G	92.6M
node1	1x24-x86	1	-	273.8M	-	737.3M	-
node2	1x24-x86	1	-	273.8M	-	737.3M	-

Sun N1 Grid Engine

- Feature precedentemente a pagamento, ora gratuite:
 - DB per accounting e analisi dettagliate
 - ARCO, applicazione webBased per la visione dei dati di accounting e analisi
 - Supporto ai nodi Windows

Sun N1 Grid Engine DEMO

DRMAA

Distributed Resource Management Application API

DRMAA

- Librerie in C o Java per la gestione dei job
- Working Group su www.drmaa.org
- Standard riconosciuto dal Global Grid Forum (www.ggf.org)
- Promosso da numerosi vendor, quali Sun, Intel, IBM, Altair, Condor
- Sun rilascia l'implementazione DRMAA per N1GE gratuitamente

DRMAA

- Queste librerie permettono lo sviluppo di applicazioni *grid-aware*
- Utili soprattutto in contesti applicativi *misti*
- Facile realizzare Web-Service che permettono un utilizzo, in modalità ASP, di servizi specifici (ad esempio Decision Support)

DRMAA

Cosa permettono di fare ?

- Il set di operazioni al momento è relativamente limitato
- Vengono supportate le maggiori operazioni:
 - Sottomissione Job
 - Monitoraggio e Controllo Job
 - Recupero dello stato di un Job terminato

DRMAA

Pro:

- Astrazione dei comandi da shell
- Ogni vendor rilascia la sua implementazione
- API veramente semplici da usare

Contro:

- API solo *Job oriented*

DRMAA

Come Funzionano ? Le due classi principali:

- **Session:**
 - Interfaccia unica per tutte le operazioni sul sistema DRM
- **JobTemplate:**
 - Classe per la creazione di un job con tutte le sue proprietà

DRMAA

- Come Funzionano ? Esempio.

```
public static void main (String[] args) {
    SessionFactory factory = SessionFactory.getFactory ();
    Session session = factory.getSession ();

    try {
        session.init (null);

        JobTemplate jt = session.createJobTemplate ();
        jt.setRemoteCommand ("sleeper.sh");
        jt.setWorkingDirectory (HOME_DIRECTORY + "/jobs");
        jt.setArgs (new String[] {"5"});

        String id = session.runJob (jt);
    }
}
```

(continua)

DRMAA

- Come Funzionano ? Esempio (continua).

```
        session.deleteJobTemplate (jt);  
        session.exit ();  
    } catch (DrmaaException e) {  
        System.out.println ("Error: " + e.getMessage ());  
    }  
}
```

DRMAA

Come Funzionano ?

- Un'altra importante classe:
 - **JobInfo:**
 - Informazioni sullo stato e l'uso di un job terminato.

DRMAA

Come Funzionano ? Esempio:

```
JobInfo info = session.wait (id, Session.TIMEOUT_WAIT_FOREVER);

if (info.wasAborted ()) {
    System.out.println("Job "+info.getJobId()+"never
                                                                ran");
} else if (info.hasExited ()) {
    System.out.println("Job "+info.getJobId()+"finished
                                                                regularly with exit
                                                                status"+info.getExitStatus());
} else if (.....
    .....
    .....
}
```



Software per la GESTione delle COde



Cos'è GECO ?

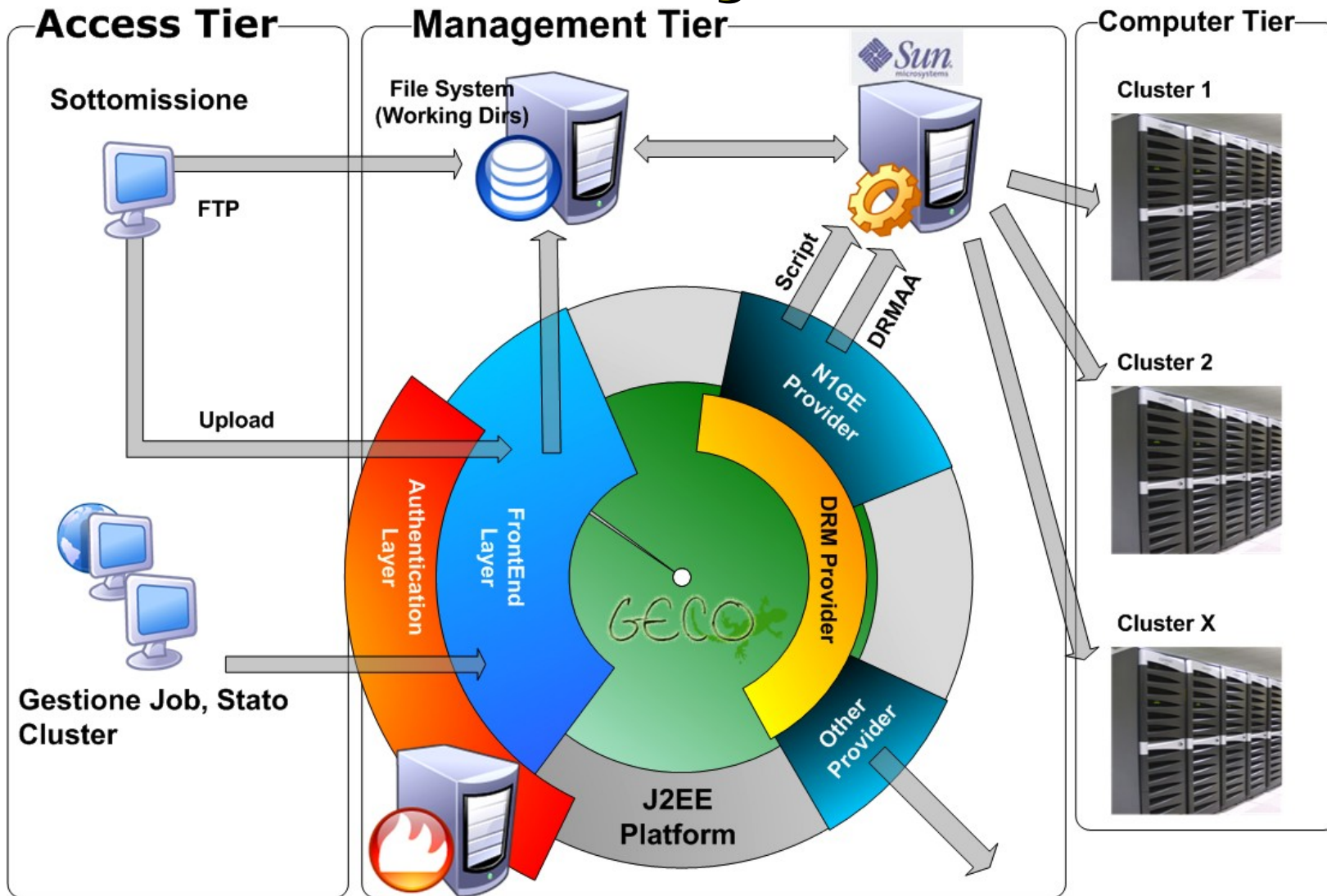
- Interfaccia Web-Based per il monitoraggio e la gestione dei Job e dei cluster.
- Versione Open Edition per il monitoraggio delle code di un sistema DRM
- Versione Enterprise Edition per la gestione dei progetti e del ciclo di vita dei Job (sottomissione, controllo, cambio stato, gestione working dir) e il monitoraggio dei cluster



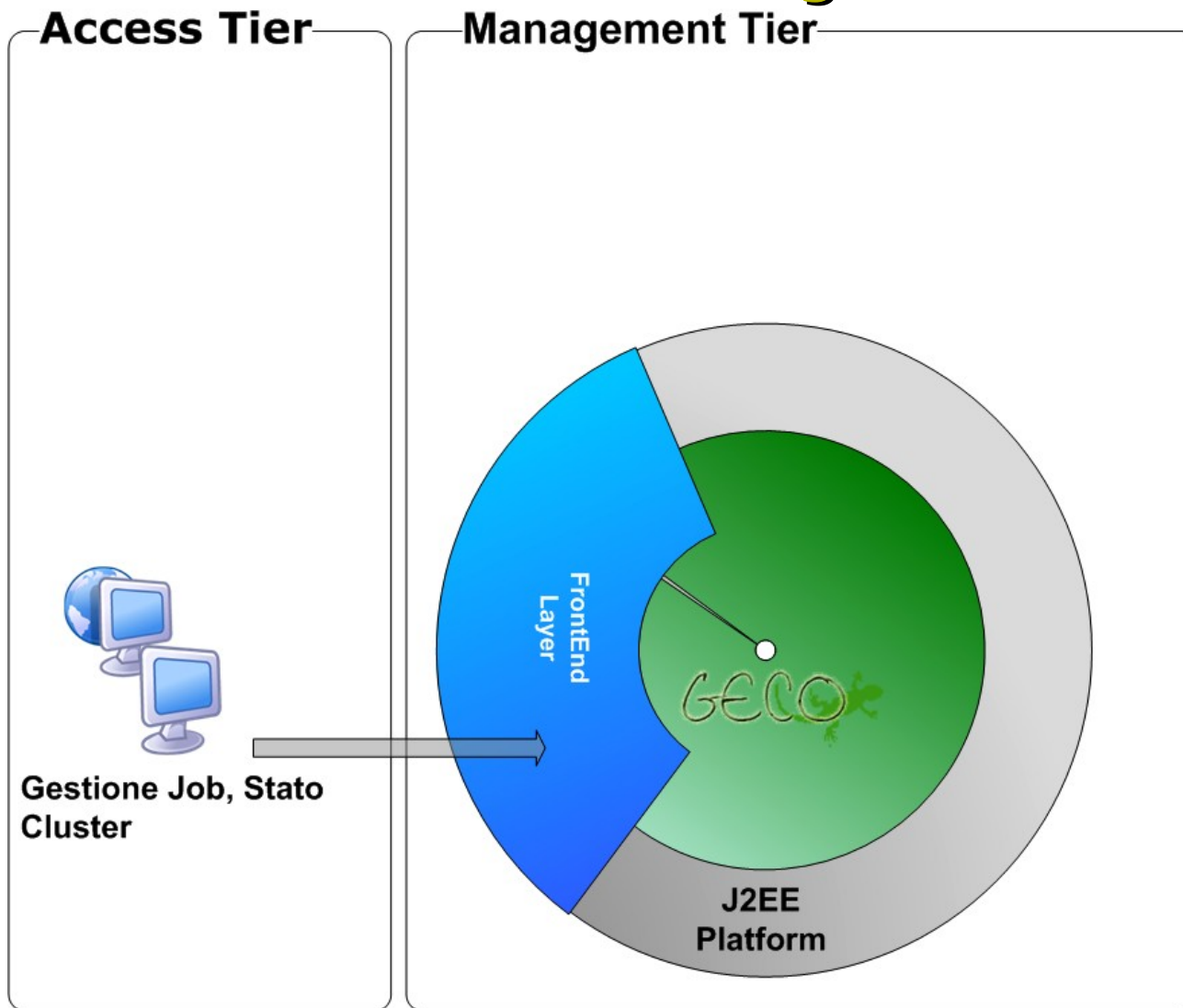
Architettura

- Sistema completamente J2EE compatibile
- Interfaccia Web-Based Mozilla/FireFox e Internet Explorer compatibile
- Interfaccia grafica dinamica (AJAX powered)
- Sistema pluggabile per compatibilità con i maggiori sistemi DRM (N1GE, LSF, PBS)

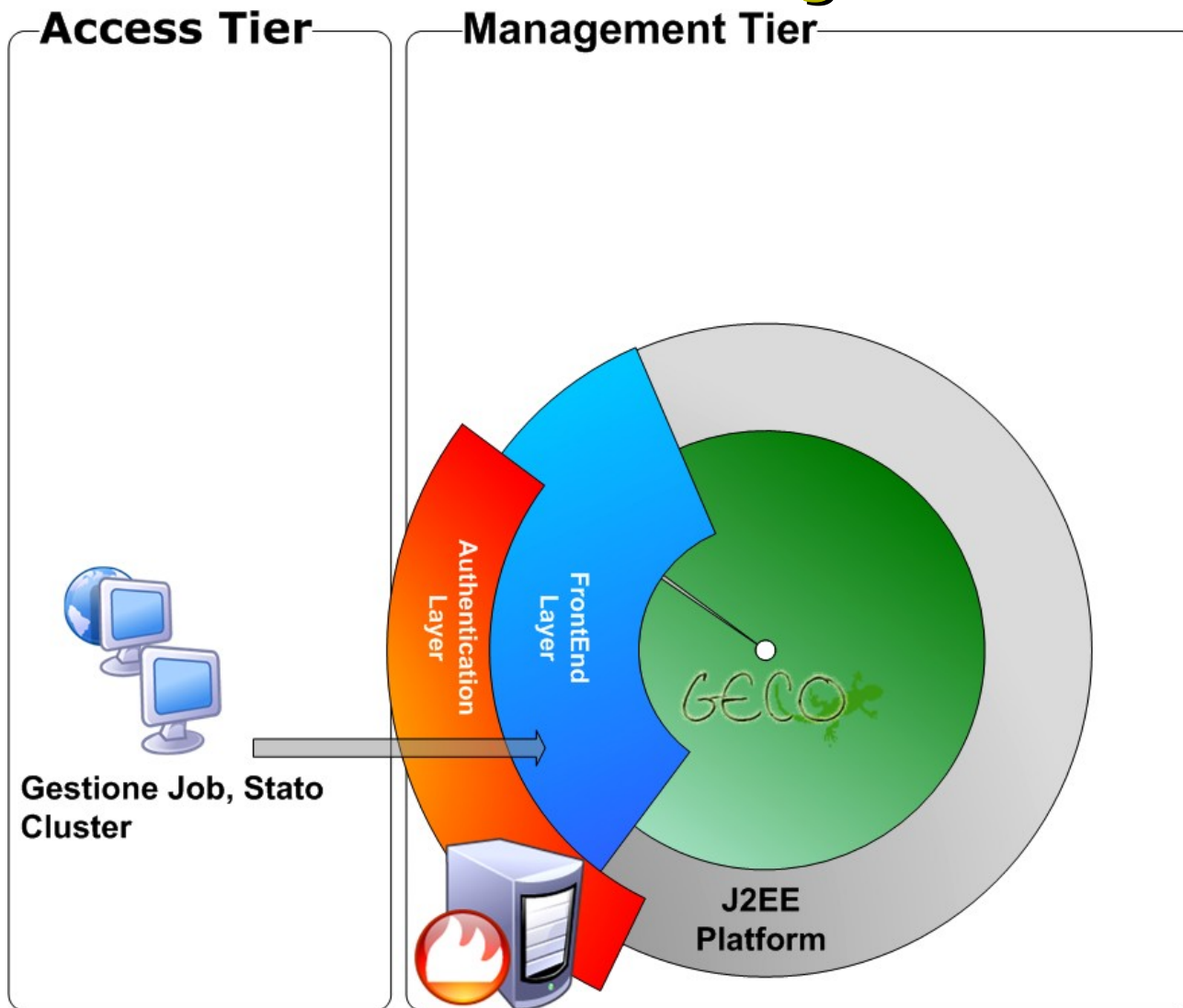
GECO - Architettura Logica



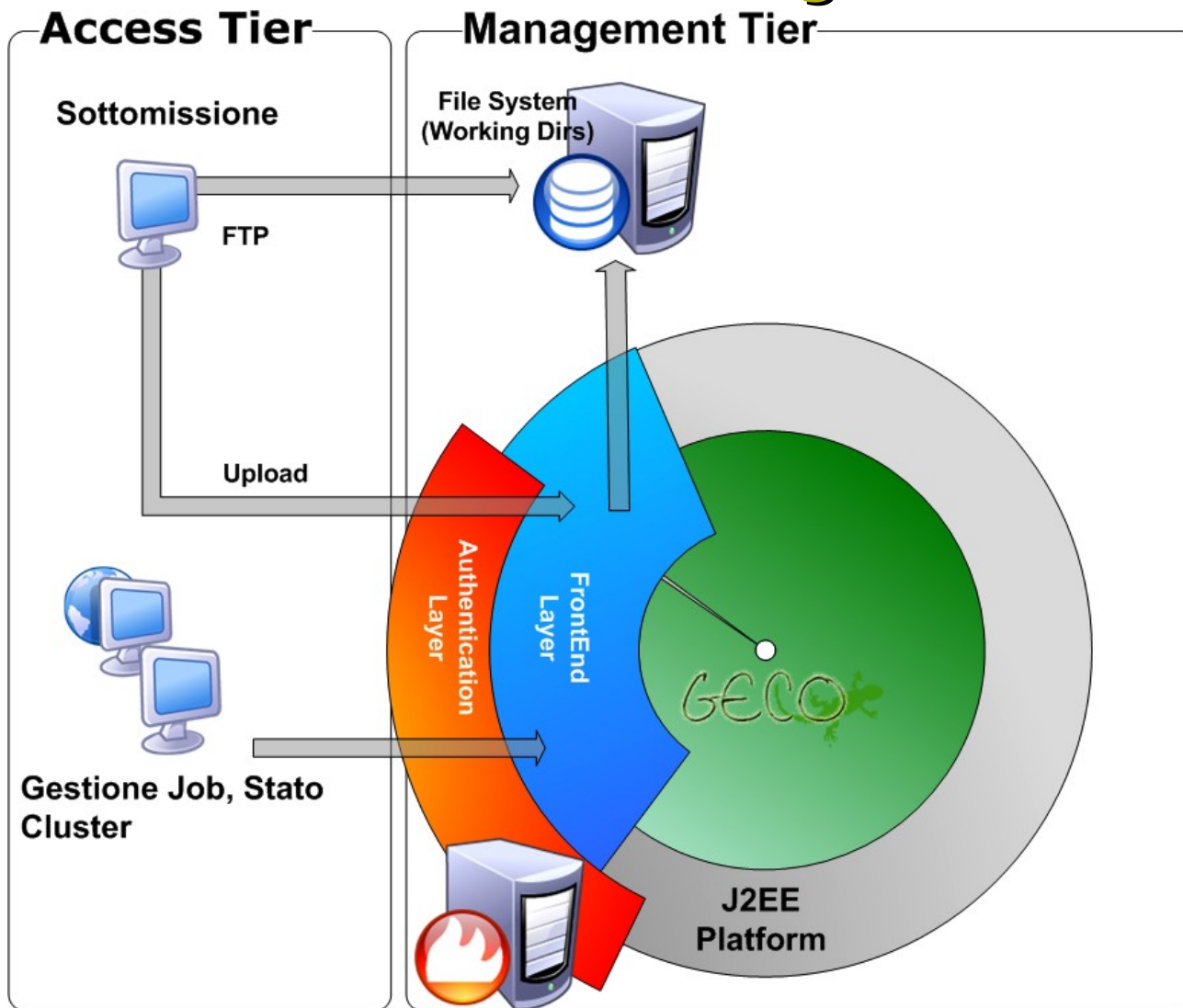
GECO - Architettura Logica



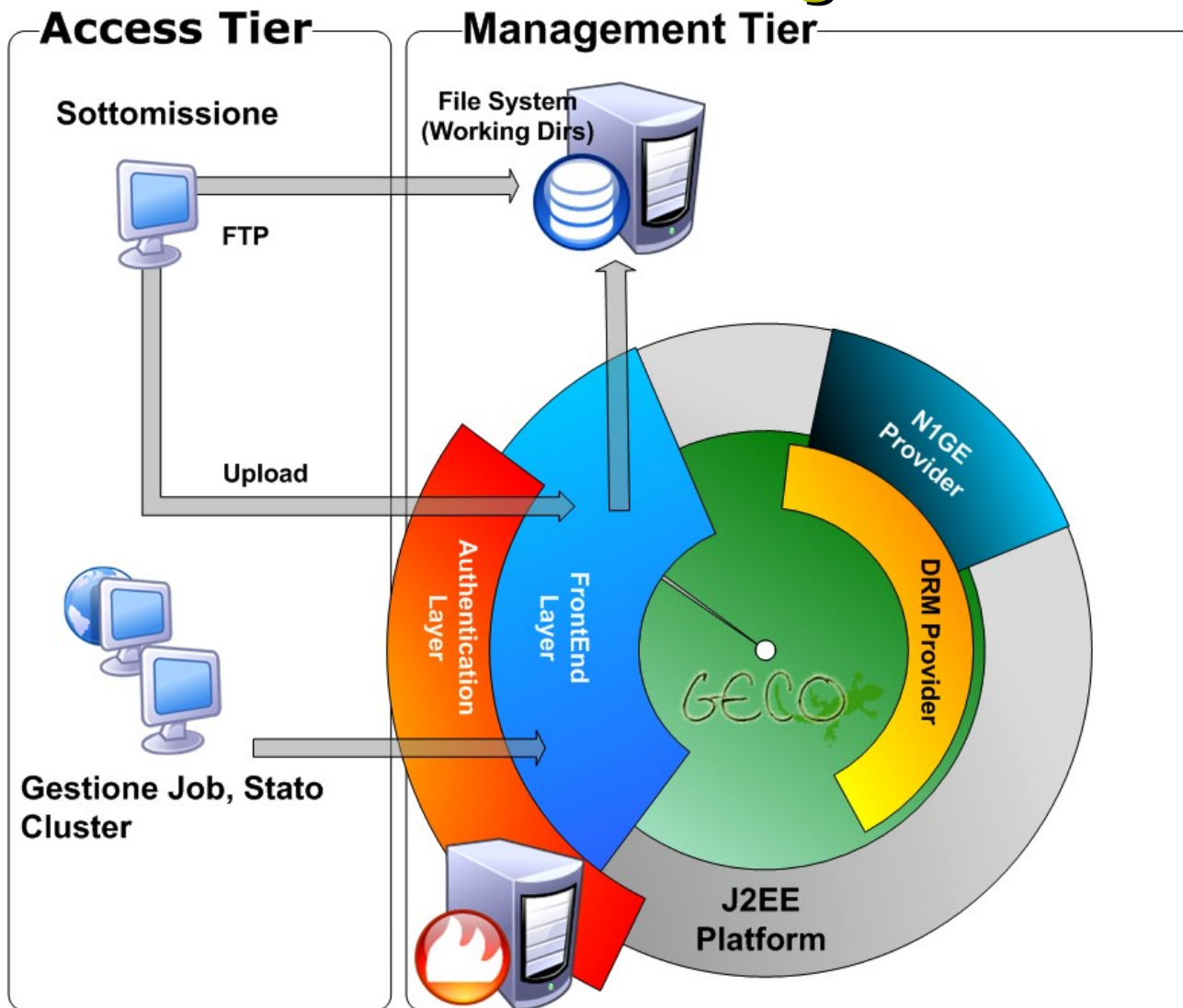
GECO - Architettura Logica



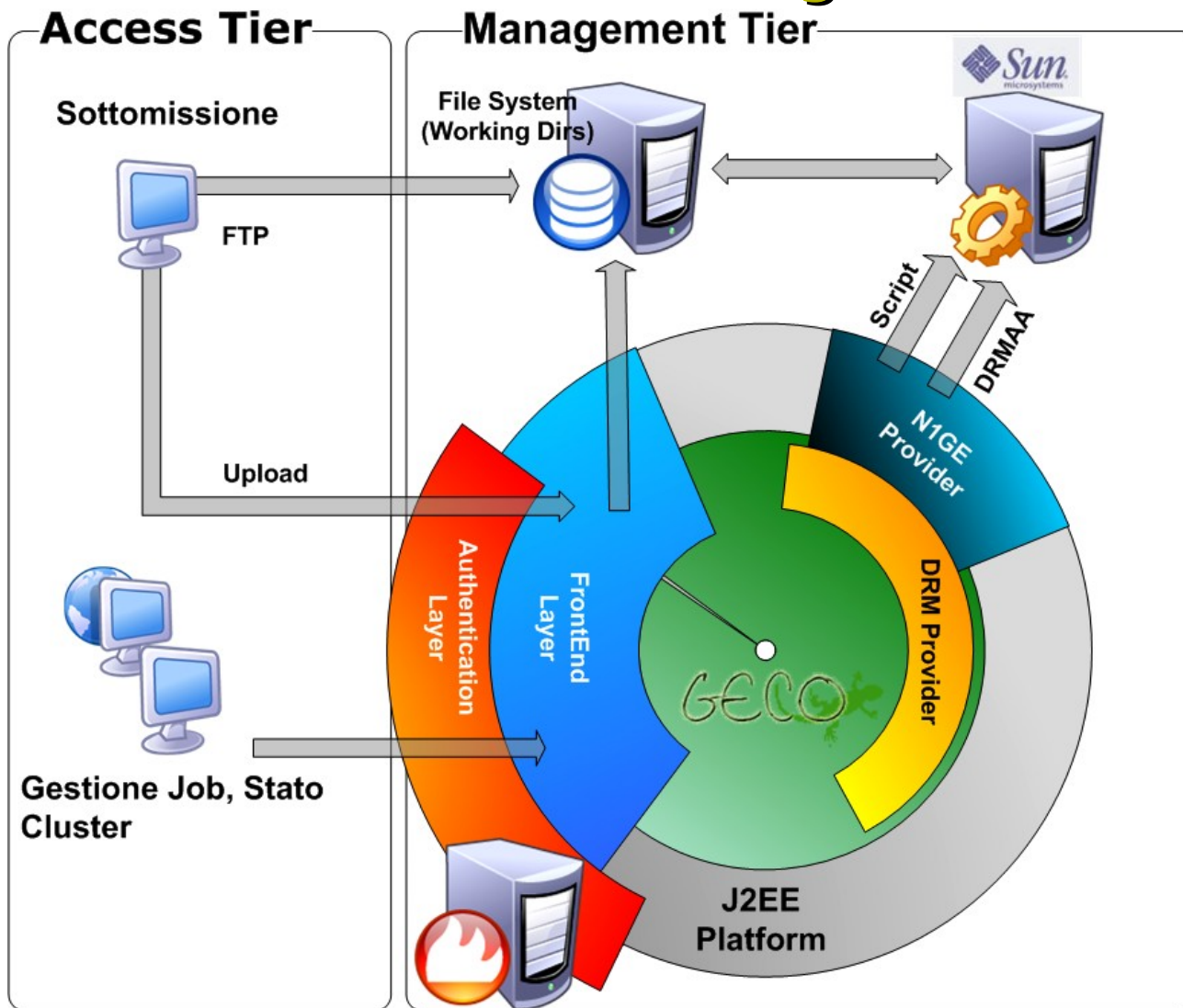
GECO - Architettura Logica



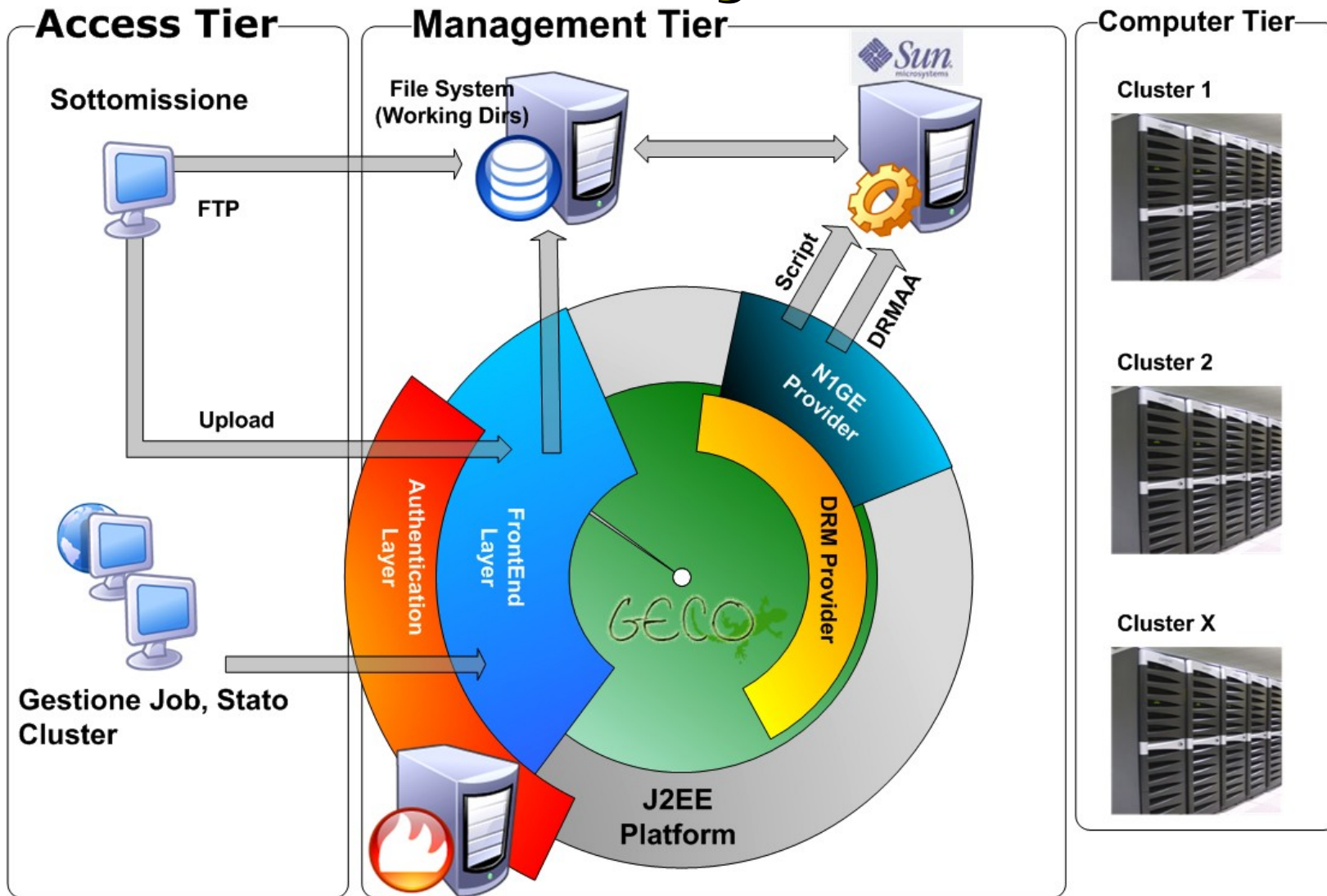
GECO - Architettura Logica



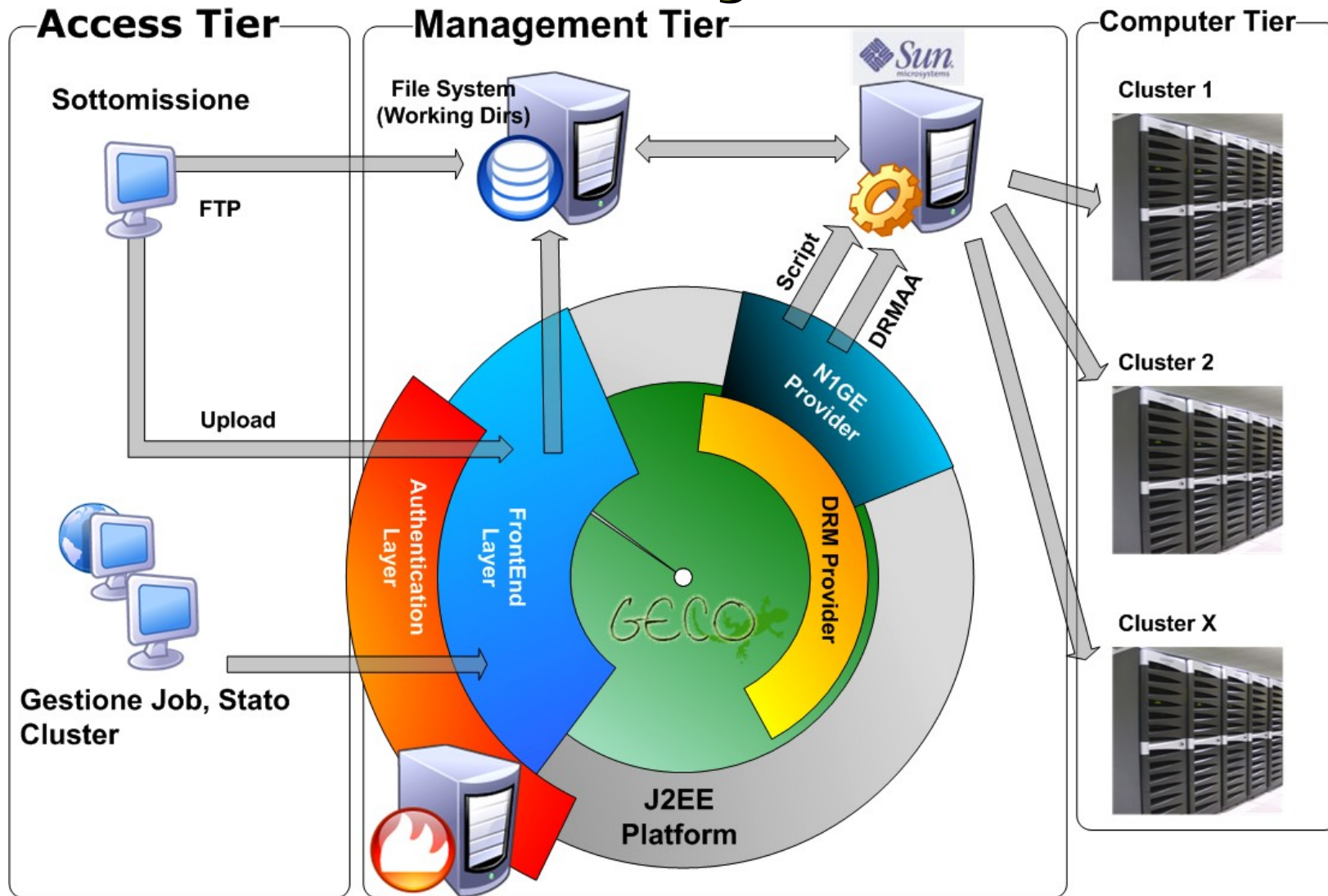
GECO - Architettura Logica



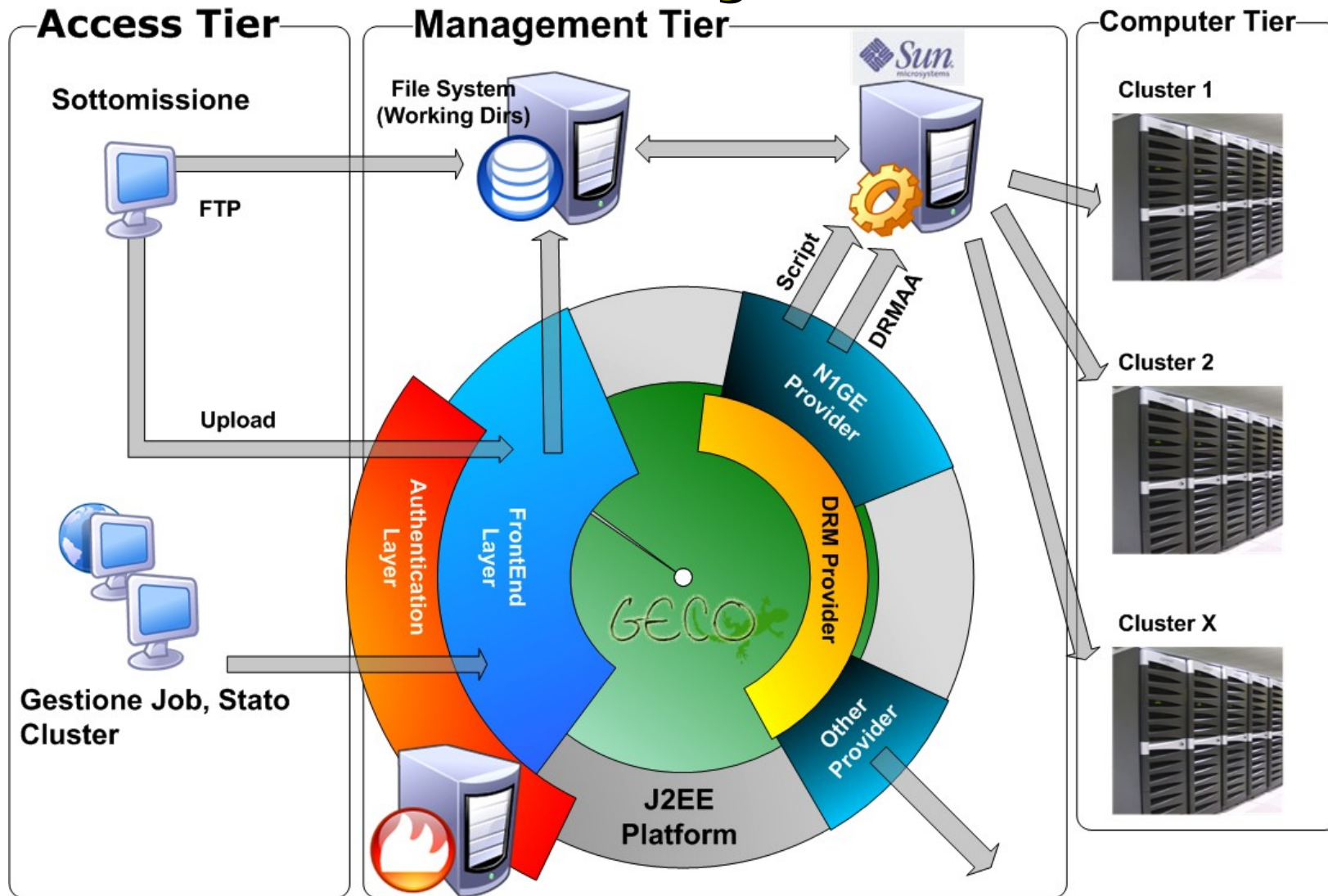
GECO - Architettura Logica



GECO - Architettura Logica



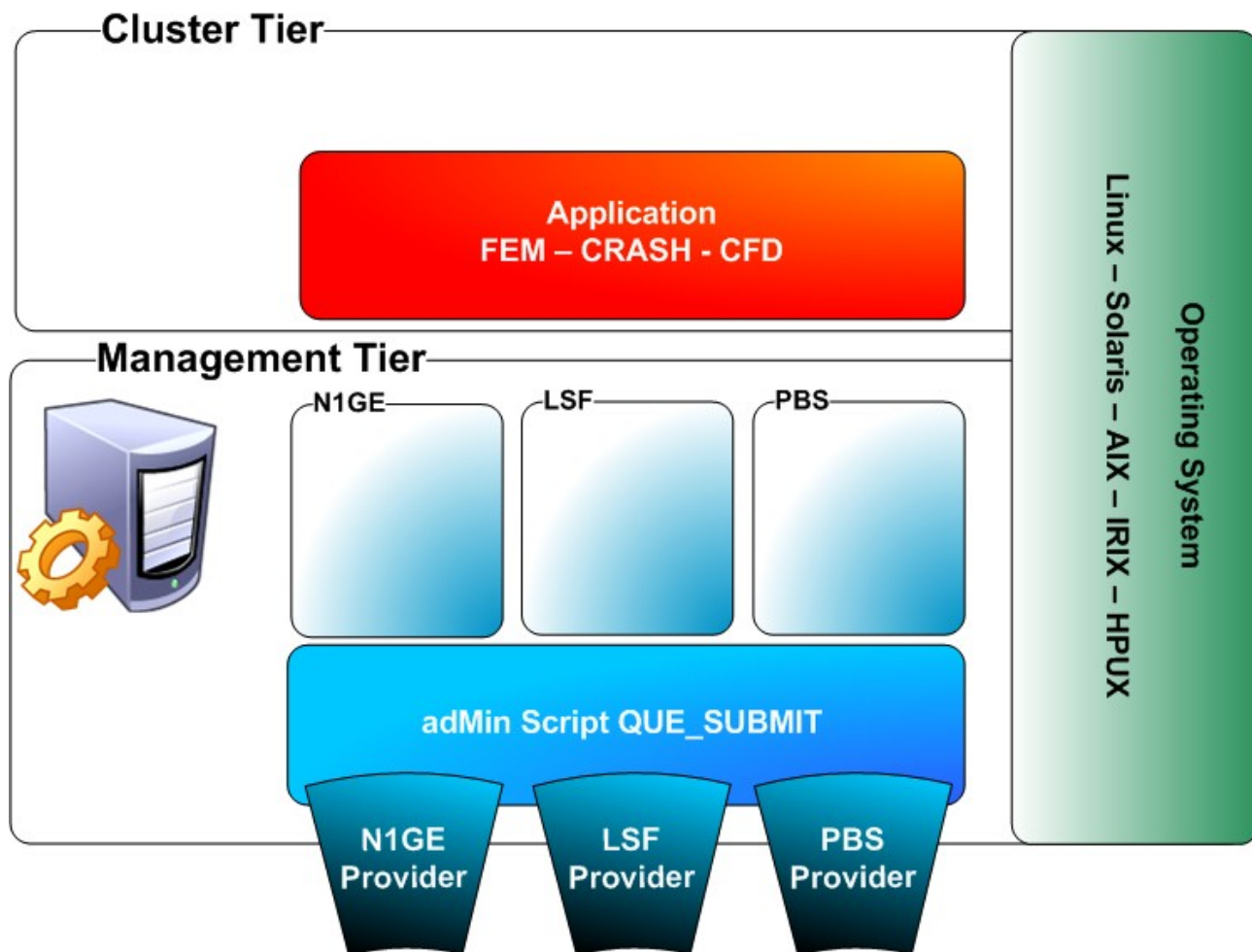
GECO - Architettura Logica





- Il cuore del sistema di integrazione con i DRM

Architettura di Scripting



GECCO Architettura di Scripting (que_submit)

- Gestisce il trasferimento degli input sul nodo di calcolo e la restituzione dei risultati
- Attualmente integrato e testato con le seguenti applicazioni:
 - CFX – Fluent – Poweflow - Star-CD
 - MSC Nastran, Marc, Adams
 - Abaqus – LS-Dyna – Pam-Crash - Radioss

GECCO 

DEMO



CONFERENCE '06

Grazie !

Per info:

minelli@adminsrl.it

farinaperseu@adminsrl.it

<http://www.adminsrl.it>

