

Esercitazione su Globus Toolkit 2:  
LDAP, MDS

Paolo Trunfio  
trunfio@deis.unical.it

1

## LDAP

Il Monitoring and Discovery Service (MDS) di Globus Toolkit 2 è basato sullo standard *Lightweight Directory Access Protocol (LDAP)*.

LDAP definisce un metodo standard per l'accesso alle informazioni di una directory.

Una directory è un elenco di informazioni su oggetti sistemati in un certo ordine, che fornisce dettagli su ciascun oggetto.

Una directory è un database specializzato (data repository), che memorizza informazioni *tipate* ed *ordinate* su *oggetti*.

Un esempio comune è una directory dei numeri di telefono di una città: gli oggetti elencati sono le persone, i cui nomi sono inseriti in ordine alfabetico, ed i dettagli forniti su ogni persona sono l'indirizzo ed il numero telefonico.

2

## LDAP (cont.)

Una caratteristica delle directory è che sono accedute in lettura molto più spesso che in scrittura. Ad es., migliaia di persone possono cercare in una directory il numero di telefono di un abbonato, ma il numero di telefono cambierà raramente.

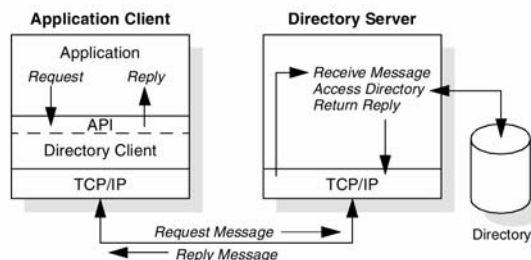
L'accesso in scrittura è in genere limitato agli amministratori del sistema o al proprietario della porzione di informazione da aggiornare. Poiché le directory sono state pensate per memorizzare informazioni relativamente statiche e sono ottimizzate per questo scopo, non sono appropriate per memorizzare informazioni che cambiano rapidamente.

Mentre i database relazionali supportano query ed operazioni di update mediante SQL, le directory LDAP usano un protocollo di accesso semplificato ed ottimizzato che può essere usato in applicazioni relativamente semplici.

3

## LDAP (cont.)

Le directory sono accedute secondo un modello client-server:



La richiesta è effettuata dal *directory client*, ed il processo che cerca le informazioni nella directory è chiamato *directory server*. Talvolta un server può divenire client di altri server per ottenere le informazioni necessarie per processare una richiesta.

Il client non è dipendente da una particolare implementazione del server, ed il server può implementare la directory come preferisce.

4

## LDAP (cont.)

LDAP definisce il contenuto dei messaggi scambiati tra i directory client ed i directory server. I messaggi specificano le operazioni richieste dal client (ricerca, modifica, cancellazione, inserimento ed altro), le risposte del server, ed il formato dei dati trasportati nei messaggi.

I messaggi LDAP sono trasportati su TCP/IP: ci sono quindi operazioni per la connessione e la disconnessione di una sessione fra il client ed il server.

Poiché LDAP fu inteso originariamente come un'alternativa Lightweight al protocollo DAP per accedere directory X.500, esso segue il modello di informazione X.500.

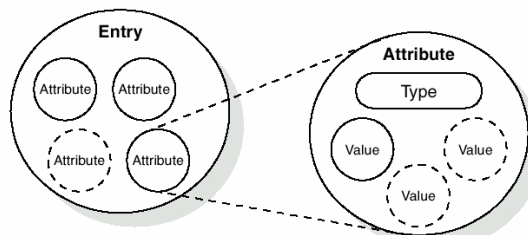
5

## LDAP: modello d'informazione

La directory memorizza ed organizza strutture dati conosciute come *entry*. Ogni entry ha un nome chiamato *Distinguished Name (DN)* che lo identifica univocamente.

Il DN consiste di una sequenza di parti chiamate *Relative Distinguished Name (RDN)*, così come il nome di un file consiste di un percorso di nomi di directory. Le entry sono organizzate in una struttura gerarchica ad albero basata sui loro DN. Questo albero di entry è chiamato *Directory Information Tree (DIT)*.

Ogni entry contiene uno o più *attributi* che descrivono l'entry e ciascun attributo ha un tipo ed uno o più *valori*:



6

## LDAP: modello d'informazione (cont.)

Il tipo dell'attributo è associato con una sintassi, che specifica che genere di valori può essere memorizzato, e definisce le regole utili, ad esempio, per l'ordinamento lessicografico o la comparazione in fase di ricerca (case sensitive o insensitive, etc.).

Le entry possono rappresentare oggetti di interesse di vario genere, quali persone, organizzazioni, schede di rete, software; gli attributi contengono informazioni su tali oggetti.

Ogni entry ha un attributo speciale chiamato *objectclass*: i valori di tale attributo determinano quali sono gli altri attributi che l'entry deve o può contenere. Ad esempio se un'entry contiene un attributo *objectclass* con valore **MdsMemoryRam**, deve contenere gli attributi **Mds-Memory-Ram-sizeMB** e **Mds-Memory-Ram-freeMB**.

7

## LDAP: modello d'informazione (cont.)

### Esempio di entry nella directory LDAP di un GRIS:

```
dn: Mds-Device-name=physical memory,  
    Mds-Device-Group-name=memory,  
    Mds-Host-hn=thebe.deis.unical.it,  
    Mds-Vo-name=local,o=grid  
objectClass: Mds  
objectClass: MdsDevice  
objectClass: MdsMemoryRam  
Mds-Device-name: physical memory      } MdsDevice  
Mds-Memory-Ram-sizeMB: 1006           } MdsMemoryRam  
Mds-Memory-Ram-freeMB: 880           }   
Mds-validfrom: 20040127091330Z        }   
Mds-validto: 20040127101330Z         } Mds  
Mds-keepsto: 20040130040650Z
```

8

## LDAP: modello d'informazione (cont.)

La definizione di una objectclass è inclusa in un cosiddetto *schema*. Uno schema contiene la definizione degli attributi implicati da una objectclass, e se tali attributi sono richiesti od opzionali.

La conoscenza da parte dell'LDAP server di uno schema, assicura che tutti gli attributi richiesti per un'entry che include l'objectclass corrispondente allo schema, siano presenti prima che l'entry stessa sia memorizzata.

Lo schema-checking assicura anche che attributi non presenti nello schema non possano essere memorizzati nell'entry. Gli schemi definiscono anche l'ereditarietà ed il subclassing degli oggetti e dove nella struttura DIT tali oggetti possono apparire.

Lo schema LDAP usato dai server MDS si trova nel file:

**\$GLOBUS\_LOCATION/etc/grid-info-resource.schema**

9

## LDAP: modello d'informazione (cont.)

### Un estratto dello schema MDS:

```
attributetype ( 1.3.6.1.4.1.3536.2.6.2.5.1.1.0.1
  NAME 'Mds-Memory-Ram-sizeMB'
  DESC 'Installed space (MB)'
  EQUALITY integerMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
)

attributetype ( 1.3.6.1.4.1.3536.2.6.2.5.1.1.0.2
  NAME 'Mds-Memory-Ram-freeMB'
  DESC 'Unallocated space (MB)'
  EQUALITY integerMatch
  ORDERING caseIgnoreOrderingMatch
  SUBSTR caseIgnoreSubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
)

objectclass ( 1.3.6.1.4.1.3536.2.6.2.5.1.1
  NAME 'MdsMemoryRam'
  SUP 'Mds'
  AUXILIARY
  MUST ( Mds-Memory-Ram-sizeMB $ Mds-Memory-Ram-freeMB )
)
```

10

## Uso di LDAP nel MDS

Il Directory Information Tree (DIT) di LDAP è progettato come un servizio distribuito: i sottoalberi, in numero arbitrariamente grande, possono essere memorizzati su altrettanti server.

Il servizio locale richiesto per supportare il MDS è esattamente un LDAP server, più le utility usate per popolare questo server con informazioni sulle risorse del sito. Il servizio MDS globale è semplicemente l'insieme di tutti questi server.

Ogni servizio Globus (per es. il GRAM) è responsabile di produrre le informazioni che gli utenti di questo servizio possono trovare utili, e di usare l'informazione per aumentare la sua flessibilità e le sue prestazioni.

11

## GRIS

Il servizio locale installato su ogni host è detto *Grid Resource Information Service (GRIS)*.

Le informazioni fornite da un GRIS includono tipo e versione del sistema operativo, memoria disponibile e tipo di CPU di uno specifico host.

Il servizio GRIS risponde alle interrogazioni mediante un LDAP server attivo sulla porta standard 2135.

12

## GIIS

Un Grid Index Information Service (GIIS) è un LDAP server centralizzato che fornisce informazioni su un insieme di risorse. Il GIIS può essere descritto come una cache per le informazioni fornite dai GRIS di una determinata organizzazione.

Le informazioni sui diversi host, fornite dai rispettivi GRIS, sono pubblicate dal GIIS in modo gerarchico, sulla base dello spazio dei nomi (DN) delle risorse.

Un host sul quale risiede un GRIS, esegue uno script che invia periodicamente al proprio GIIS le informazioni aggiornate sulle proprie risorse.

Il GIIS è un componente opzionale del servizio di informazione di Globus: anche senza un GIIS un nodo Globus è completamente funzionante, pur non disponendo di un server centrale contenente informazioni sulle sue risorse. Ogni nodo Globus, infatti, rende disponibili alle applicazioni remote le proprie informazioni di sistema, mediante il proprio GRIS.

13

## grid-info-search

Consente all'utente di interrogare un server MDS sulla base di un filtro di ricerca.

Sintassi di base:

**grid-info-search** OPTIONS [searchFilter] [attributes]

**searchFilter** Definisce i criteri con i quali si stabilisce se un'entry deve essere restituita da una operazione di ricerca; se non viene indicato un filtro di ricerca, vengono restituite tutte le entry.

**attributes** Lista degli attributi che devono essere restituiti; se non viene indicato alcun attributo, vengono restituiti tutti gli attributi delle entry selezionate dal filtro di ricerca.

14

## grid-info-search (cont.)

### OPTIONS

- h host**            Il nome dell'host su cui il server MDS è in esecuzione. Il default è localhost.
- p port**            Il numero di porta su cui il server MDS è in ascolto. La porta di default è 2135.
- b basedn**        Punto del DIT da cui deve iniziare la ricerca. Il default è "Mds-Vo-name=local, o=Grid"
- s base | one | sub**    Specifica la profondità della ricerca: con *base* la ricerca è limitata ad una particolare entry; con *one* la ricerca viene estesa ad un livello sotto *basedn*; con *sub* la ricerca viene estesa a tutto il sottoalbero sotto *basedn* (default).
- T seconds**      Timeout (in secondi). Il valore di default è 30
- x**                Effettua un bind anonimo.

15

## grid-info-search (cont.)

Un filtro di ricerca di base è un'asserzione di valore d'attributo della forma:

`attributo operatore valore`

I filtri di ricerca possono essere combinati con operatori booleani per formare filtri più complessi. La sintassi per combinare i filtri di ricerca è:

```
("&" or "|" (filtro1) (filtro2) (filtro3) ...)  
("!" (filtro))
```

Una ricerca effettuata con il filtro "`(&(filtro1) (filtro2))`" restituisce le entry che rispettano sia i requisiti del `filtro1` che del `filtro2`, mentre l'espressione "`(|(filtro1) (filtro2))`" causa la restituzione di tutte le entry che rispettano i requisiti del `filtro1` oppure del `filtro2`. Il filtro "`(!(filtro1))`" restituisce le entry per le quali l'espressione di `filtro1` non è verificata.

E' possibile innestare gli operatori booleani su più livelli, al fine di costruire filtri più specializzati. L'uso del carattere "\*" è ammesso con il consueto significato.

16



## grid-info-search (cont.)

### Esempi:

```
grid-info-search -x -h jupiter -b "Mds-Vo-name=kgrid, o=Grid"
```

Restituisce tutto il DIT mantenuto dal GIIS dell'organizzazione "kgrid"

```
grid-info-search -x -h jupiter "objectclass=MdsCpu"
```

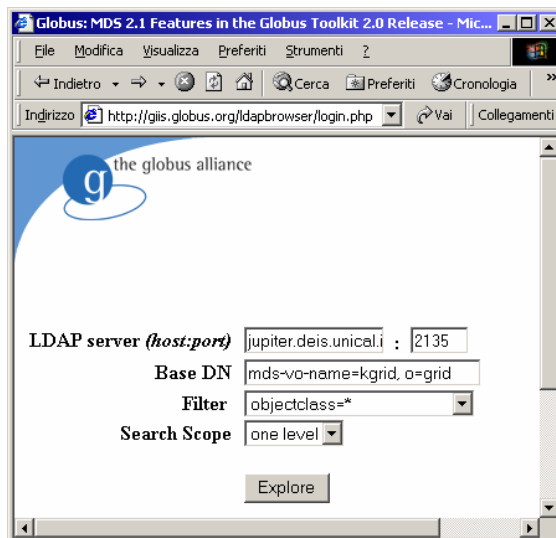
Restituisce tutte le entry memorizzate nel GRIS di `jupiter` che possiedono un attributo `objectclass` con valore uguale a `MdsCpu`, stampando Distinguished Name e attributi di ciascuna entry.

```
grid-info-search -x -h jupiter -b "Mds-Vo-name=kgrid, o=Grid" \  
"(&(objectclass=MdsCpu)(Mds-Cpu-speedMHz>=2000))" dn
```

Restituisce i `dn` di tutte le risorse di calcolo dell'organizzazione "kgrid", avente una `cpu` con `speed > 2000` MHz.

17

## Esempio di browser LDAP



The screenshot shows a web browser window titled "Globus: MDS 2.1 Features in the Globus Toolkit 2.0 Release - Mic...". The address bar shows the URL `http://giis.globus.org/ldapbrowser/login.php`. The page content includes the Globus Alliance logo and a search configuration form with the following fields:

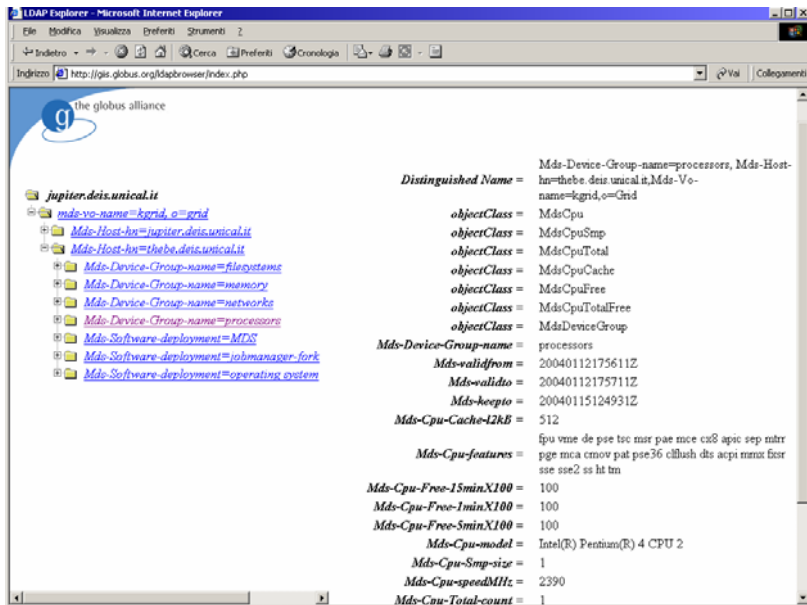
LDAP server ( <i>host:port</i> )	<input type="text" value="jupiter.deis.unical.i : 2135"/>
Base DN	<input type="text" value="mds-vo-name=kgrid, o=grid"/>
Filter	<input "="" type="text" value="objectclass="/>
Search Scope	<input type="text" value="one level"/>

Below the form is an "Explore" button.

<http://giis.globus.org/ldapbrowser/login.php>

18

## Esempio di browser LDAP (cont.)



The screenshot shows the LDAP Explorer interface in Microsoft Internet Explorer. The left pane displays a tree view of LDAP entries under the domain `jupiter.deis.unical.it`. The right pane shows the details for the entry `Mds-Device-Group-name=processors`.

**LDAP Explorer - Microsoft Internet Explorer**

Indirizzo: <http://gis.globus.org/ldapbrowser/index.php>

**Tree View:**

- `jupiter.deis.unical.it`
  - `mds-vo-name=kgrid,o=grid`
  - `Mds-Host-ho=jupiter.deis.unical.it`
  - `Mds-Host-ho=thebe.deis.unical.it`
  - `Mds-Device-Group-name=filesystems`
  - `Mds-Device-Group-name=memory`
  - `Mds-Device-Group-name=networks`
  - `Mds-Device-Group-name=processors`
  - `Mds-Software-deployment=MDS`
  - `Mds-Software-deployment=jobmanager.fork`
  - `Mds-Software-deployment=operating system`

**Entry Details:**

**Distinguished Name =** Mds-Device-Group-name=processors, Mds-Host-ho=thebe.deis.unical.it, Mds-Vo-name=kgrid, o=Grid

**objectClass =** MdsCpu

**objectClass =** MdsCpuSmp

**objectClass =** MdsCpuTotal

**objectClass =** MdsCpuCache

**objectClass =** MdsCpuFree

**objectClass =** MdsCpuTotalFree

**objectClass =** MdsDeviceGroup

**Mds-Device-Group-name =** processors

**Mds-validfrom =** 20040112175611Z

**Mds-validto =** 20040112175711Z

**Mds-keopto =** 20040115124931Z

**Mds-Cpu-Cache-4kB =** 512

**Mds-Cpu-features =** fpu vme de ppe tsc mtr pae mce c68 apic sep nhr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm

**Mds-Cpu-Free-15minX100 =** 100

**Mds-Cpu-Free-1minX100 =** 100

**Mds-Cpu-Free-5minX100 =** 100

**Mds-Cpu-model =** Intel(R) Pentium(R) 4 CPU 2

**Mds-Cpu-Smp-size =** 1

**Mds-Cpu-speedMHz =** 2390

**Mds-Cpu-Total-count =** 1

19

## Riferimenti

### LDAP:

H. Johner et al., "Understanding LDAP", IBM Redbooks  
<http://www.redbooks.ibm.com/redbooks/pdfs/sg244986.pdf>

### MDS di Globus Toolkit 2:

<http://www.globus.org/mds/mds2>

20