Introduzione agli URI

Fabio Vitali

Introduzione

Qui esaminiamo:

- Gli Universal Resource Identifier (URI)

URI

- Gli URI (Universal Resource Identifier) sono una sintassi usata in WWW per definire i nomi e gli indirizzi di oggetti (risorse) su Internet.
- Questi oggetti sono considerati accessibili tramite l'utilizzo di protocolli esistenti, inventati appositamente, o ancora da inventare.
- Gli URI si orientano a risolvere il problema di creare un meccanismo ed una ***sintassi di accesso unificata*** alle risorse di dati disponibili via rete.
- Tutte le istruzioni d'accesso ai vari specifici oggetti disponibili secondo un dato protocollo sono codificate come una stringa di indirizzo

L’esigenza di identificatori (1)

Gli URI sono stati verosimilmente il fattore determinante per il successo del WWW.

Attraverso gli URI, il WWW è stato in grado di identificare risorse accessibili tramite il proprio protocollo, HTTP, e tramite tutti gli altri protocolli esistenti (FTP, Telnet, Gopher, WAIS, ecc.).

Il punto principale a cui gli altri sistemi non erano arrivati era una sintassi universale, indipendente dal protocollo e facilmente memorizzabile o scambiabile con cui identificare le risorse di rete.

L’esigenza di identificatori (2)

Il WWW utilizza gli identificatori in una varietà di modi:

- Link ipertestuali disponibili nel documenti HTML
- Immagini ed altri oggetti inclusi nel documento HTML (che è un formato solo testo)
- Connessioni e relazioni globali tra documenti (ad esempio, script e link possono essere messi esternamente al documento HTML e da esso riferiti globalmente.

In tutti questi casi lo stesso identificatore può essere usato dal protocollo di comunicazione, espresso nella sintassi HTML, o digitato direttamente dall’utente.

Criteri di design degli URI (1)

La sintassi degli URI é progettata per essere

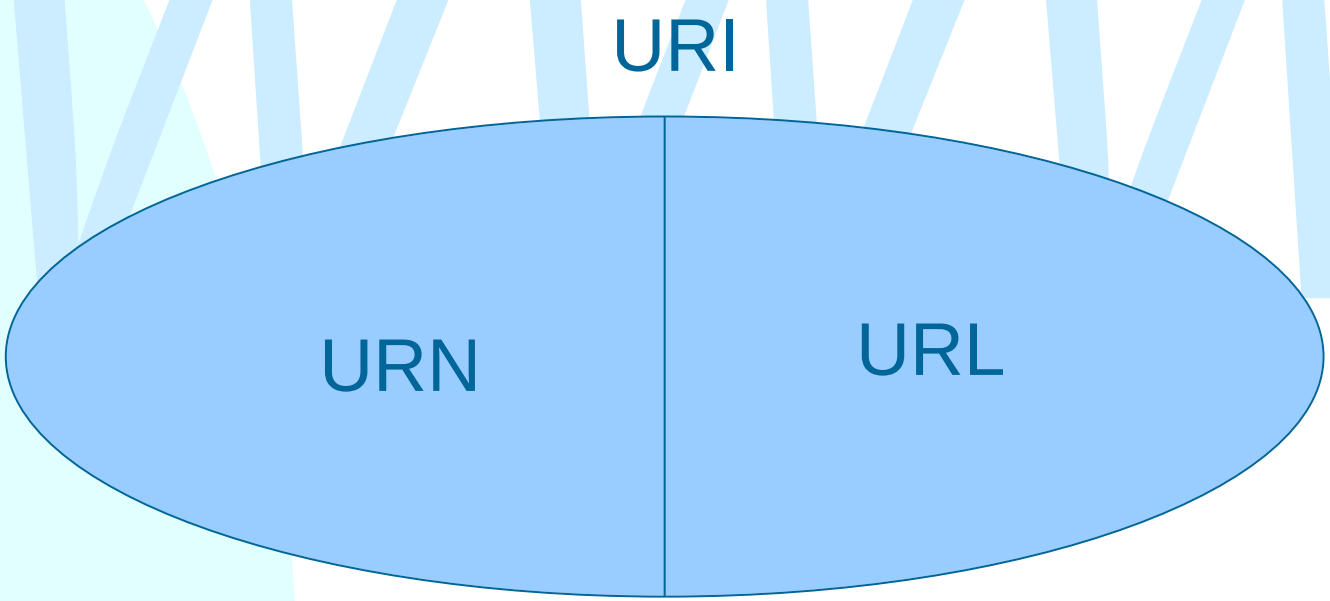
- **Estensibile**: si possono aggiungere nuovi schemi, al fine di mantenere l'accessibilità delle risorse anche se nuovi protocolli vengono inventati
- **Completa**: tutti i nomi esistenti sono codificabili e nuovi protocolli sono comunque esprimibili tramite URI
- **Stampabile**: é possibile esprimere URI con caratteri ASCII a 7-bit, così da permettere scambi lungo qualunque canale, per quanto limitato o inefficiente, inclusi carta e penna.

Lo standard URI definisce alcune regole per la generazione di schemi di naming (insiemi di nomi caratterizzati dalla dipendenza da un protocollo di accesso comune), per la definizione dei caratteri accettabili e del carattere di escape.

Criteri di design degli URI (2)

Gli ***Universal Resource Identifier*** (URI) sono, per definizione:

- ***Universal Resource Names*** (URN)
- ***Universal Resource Locator*** (URL).



Criteri di design degli URI (3)

- Gli URL sono un indirizzo della risorsa che possa essere immediatamente utilizzato da un programma per accedere alla risorsa.
- Gli URL contengono tutte le informazioni necessarie per accedere all'informazione, ma sono fragili a modifiche non sostanziali del meccanismo di accesso (es. cambio del nome di una directory).
- Gli URN sono un nome stabile e definitivo di una risorsa, che possa fornire un informazione certa ed affidabile sulla sua esistenza ed accessibilità.
- Gli URN debbono essere trasformati da un apposito servizio, negli URL attualmente associati alla risorsa. Inoltre la mappa deve essere aggiornata ogni volta che la risorsa viene spostata.

Il concetto di risorsa

Gli URI sono pensati per essere indipendenti dal meccanismo di memorizzazione effettiva sottostante.

Anche se la maggior parte degli URI fa riferimento a file memorizzati in un file system gerarchico, questo non è né necessario, né universale:

- Potrebbe essere in un file system relazionale (VM di IBM)
- Potrebbe essere in un database, e l'URI essere la chiave di ricerca
- Potrebbe essere il risultato dell'elaborazione di un'applicazione, e l'URI essere i parametri di elaborazione.

Per questo si usa il termine Risorsa, invece che File, e si fornisce una sintassi indipendente dal sistema effettivo di memrozzazione.

Mai assumere che si stia lavorando con un file system!

La sintassi degli URI

Un URI è diviso in due parti:

- `uri = schema ":" parte-specifica`

Lo schema di naming (in pratica, il protocollo) é identificato da una stringa arbitraria (ma registrata) usata come prefisso. Il carattere di due punti separa il prefisso dal resto. La decodifica del resto dell'URI é funzione del prefisso.

Ogni schema ha una sua sintassi, ma esistono delle regole che tutti gli schemi debbono rispettare.

Caratteri riservati negli URI (1)

% Il carattere “%” é il codice di escape, e serve per l’utilizzo di caratteri particolari nell’URI, precedendone il codice esadecimale. Ad esempio, per utilizzare un carattere “%” nel’URI bisogna usare la stringa “%25”

/ Il carattere “/” é utilizzato unicamente per l’identificatzione di sottoparti di uno schema gerarchico, e non può essere usato per altri scopi.

. Il punto singolo “.” o il punto punto “..” hanno anch’essi un significato gerarchico riservato, per indicare ovviamente risorse allo stesso livello o al livello superiore.

Caratteri riservati negli URI (2)

- # Il carattere di hash “#” serve per delimitare l’URI di un oggetto da un identificatore di un frammento interno alla risorsa considerata. Questo permette ad un URI di far riferimento non soltanto ad una risorsa (oggetto di interesse del server), ma anche a frammenti interni alla risorsa (che verranno identificati dal client).
- ? Il punto interrogativo “?” serve per separare l’URI di un oggetto su cui é possibile fare una query (un database, per esempio), dalla stringa usata per specificare la query.
- + All’interno della query, il segno più “+” é usato al posto dello spazio (che non é mai usato per nessuna ragione).

Caratteri riservati negli URI (3)

* L’asterisco “*” ha un significato speciale all’interno di schemi specifici.

! Analogamente il punto esclamativo “!” ha un significato all’interno di uno schema.

%XX Caratteri speciali o riservati o in generale non sicuri (es. quelli superiori al codice ASCII 127) possono essere specificati tramite codifica esadecimale introdotta dal carattere di escape.

Caratteri riservati negli URI (4)

Esempio: i due URI

- `http://www.alpha.edu/a/b/c/d`
- `http://www.alpha.edu/a/b/c%2Fd`

non sono uguali, perché, benché il codice esadecimale corrisponda al carattere “/”, nel primo caso esso ha significato gerarchico, e nel secondo fa parte del nome dell’ultima sottoparte della gerarchia, “c/d”.

URN (1)

Gli URN non hanno ancora molto successo. Non esiste ancora nessun meccanismo di URN sufficientemente affermato.

Gli scopi degli URN sono:

- **Ambito globale**: non viene indicata una locazione, ed ha lo stesso significato da ovunque lo si usi
- **Unicità globale**: non è possibile assegnare lo stesso URN a risorse diverse
- **Persistenza**: Non esiste ragione per la sua cessata esistenza a parte la cancellazione della risorsa a cui fa riferimento.
- **Scalabilità**: ogni risorsa sulla rete deve poter possedere per lungo tempo un URN

URN (2)

- **Estensibilità**: nuove funzionalità emergeranno. E' necessario che lo schema di URN permetta estensioni per coprire le esigenze delle nuove funzionalità.
- **Supporto per i meccanismi esistenti**: esistono già dei meccanismi di naming globali: numeri ISBN per i libri, identificatori pubblici ISO per gli standard, codici UPC per i prodotti fisici. Lo schema di naming deve inglobare trasparentemente questi schemi di naming.
- **Risoluzione**: deve esistere un meccanismo semplice per la mappatura di un URN nell'URL più appropriato
- **Indipendenza**: ogni suddivisione gerarchica dell'autorità dei nomi deve essere autonoma (cioè gestisce in autonomia i nomi ad essa soggetti).

URL

Lo schema, in un URL, corrisponde al protocollo di accesso da utilizzare per accedere alla risorsa. La parte specifica dello schema dipende dal protocollo specifico.

Vediamo brevemente i seguenti schemi:

- HTTP e HTTPS
- FTP
- NNTP
- SMTP
- Telnet

HTTP e HTTPS

La sintassi della parte specifica è:

`http://`*`host`*`[:`*`port`*`]/`*`path`*`[#`*`fragment`*`][?`*`query`*`]`

`https://`*`host`*`[:`*`port`*`]/`*`path`*`[#`*`fragment`*`][?`*`query`*`]`

dove:

- **host** é l'indirizzo TCP-IP o DNS, dell'host su cui si trova la risorsa
- **port** é la porta a cui il server é in ascolto per le connessioni. In mancanza di specificazione, la porta é quella di default, 80 per HTTP e 443 per HTTPS.
- **path** é un pathname gerarchico (per esempio, un filename parziale) per l'identificazione della risorsa
- **fragment** é un identificativo di una sottoparte dell'oggetto. La definizione e il ritrovamento di queste sottoparti é a carico del client, e quindi la parte di fragment viene ignorata dal server, che restituisce l'intero oggetto.
- **query** é una frase che costituisce l'oggetto di una ricerca sulla risorsa specificata.

FTP

La sintassi della parte specifica è:

```
ftp://[user[:password]@]host[:port]/path [type]
```

dove:

- User e password sono utente e password per l'accesso ad un server FTP. La loro mancanza fa partire automaticamente una connessione anonima
- Host, port e path sono l'indirizzo del server, la porta di connessione ed il nome del file dell'oggetto ricercato, come per HTTP. La porta di default è 21.
- type regola i parametri di connessione FTP, come il tipo di trasferimento (ASCII o binario).

SMTP e Telnet

SMTP

La sintassi della parte specifica è:

```
mailto:user@host
```

dove

- non esiste il prefisso “//” perché lo schema non è gerarchico
- User e host sono i componenti dell’indirizzo di e-mail del destinatario

Telnet

La sintassi della parte specifica è:

```
telnet:host
```


NNTP

La sintassi della parte specifica è:

```
news:group
news:articleID@host
nntp:host/group/digit
```

dove

- l'accesso viene fatto usualmente al news server locale (specificato in varie preferenze).
- La specifica del solo gruppo restituisce l'elenco dei messaggi presenti nel gruppo.
- La specifica nella forma articleID@host permette di specificare l'articolo secondo l'identificativo interno locale al news server identificato.
- La terza sintassi, con specifica esplicita del protocollo nntp, viene usata scarsamente e solo per news server limitati privi di meccanismo di identificazione dei messaggi per articleID.

Conclusioni

Qui abbiamo parlato di

- La sintassi degli URI e degli URL

Riferimenti

Wilde's WWW, capitolo 2

Altri testi:

- K. Sollins, L. Masinter, ***Functional Requirements for Uniform Resource Names***, RFC 2276, Jan. 1998
- T. Berners-Lee, L. Masinter, M. McCahill, ***Uniform Resource Locator***, RFC 1738, Dec. 1994
- R. Fielding, ***Relative Uniform Resource Locator***, RFC 1808, Jun 1995.