

# URI

## Introduzione

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. Tutte le istruzioni d'accesso ai vari specifici oggetti disponibili secondo un dato protocollo sono codificate come una stringa di indirizzo.

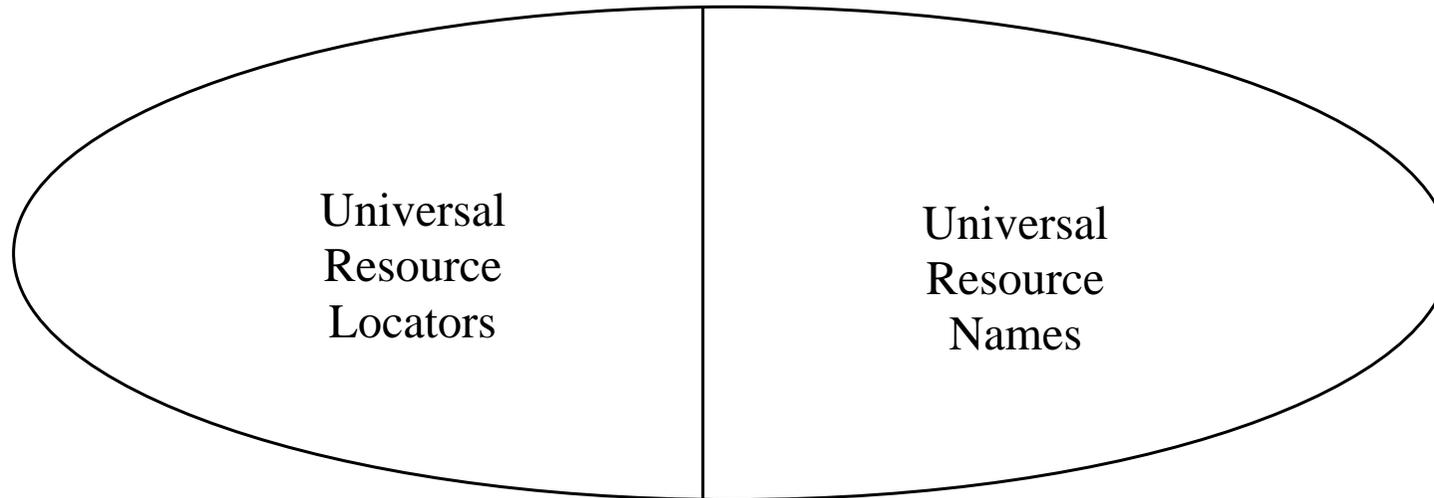
Questa stringa viene poi utilizzata dal meccanismo di accesso per la corretta identificazione dell'oggetto e per l'attivazione delle operazioni necessarie per il recupero e il trasferimento dell'oggetto stesso.

Gli URI si indirizzano a risolvere il problema di creare un meccanismo ed una sintassi di accesso unificata alle risorse di dati disponibili via rete. WWW mette a disposizione un name-space che permette di registrare ogni nome specifico di una risorsa al suo interno. Ogni membro di questo insieme é un URI.

Gli URL (Universal Resource Locator) sono URI che si riferiscono ad oggetti accessibili tramite protocolli esistenti standard, come HTTP, SMTP, WAIS, GOPHER, etc.

Gli URN (Universal Resource Names) sono una evoluzione (ancora in fieri) degli URI, rivolti alla definizione di un nome più persistente degli URL (che dipendono dal nome dell'host acceduto, dal nome del file richiesto, etc.)

### Universal Resource Identifier



# Criteri di progettazione degli URI

La sintassi degli URI é progettata per essere

- **estensibile** (si possono aggiungere nuovi schemi), completa (tutti i nomi esistenti sono codificabili)
- **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 name caratterizzati dalla dipendenza da un protocollo di accesso comune), per la definizione dei caratteri accettabili e del carattere di escape.

# La sintassi degli URI

Lo schema di naming (ovvero il protocollo) é identificato da una stringa arbitraria (ma registrata) usata come prefisso. Il carattere di due punti separa il prefisso dal resto dell'indirizzo. Il resto dell'indirizzo é chiamato path. La decodifica del path é funzione del prefisso, ma l'esistenza del carattere “/” implica l'esistenza di una struttura gerarchica.

- Il carattere “%” é un carattere riservato, e serve per produrre il codice di escape per l'utilizzo di caratteri particolari nell'URI.
- Il carattere “/” é utilizzato unicamente per l'identificazione di sottoparti di uno schema gerarchico, e non può essere usato per altri scopi.
- Il punto singolo “.” o i due punti “..” hanno un significato riservato relativo alla strutturazione gerarchica delle risorse, e fanno riferimento alla posizione attualmente identificata, e alla posizione gerarchicamente superiore a quella attualmente identificata.
- Il carattere di hash “#” serve per delimitare l'URI di un oggetto da un identificatore di un frammento interno alla risorsa considerata.

## La sintassi degli URI (segue)

- 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).
- Asterisco “\*” e punto esclamativo “!” hanno significati speciali all’interno di schemi specifici.
- 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.

### **Nota bene:**

`http://host.edu/a/b/c/d`

e

`http://host.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”.

# HTTP

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

dove:

- `host` é l'indirizzo TCP-IP, numerico o DNS, dell'host su cui é memorizzata la risorsa
- `port` é la porta a cui il server HTTP é in ascolto per le connessioni. In mancanza di specificazione, la porta é quella di default, 80.
- `path` é un pathname gerarchico (tipicamente 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.

Ad esempio:

`http://www.cs.unibo.it:8088/pub/vitali/index.html#sezione2`

# FTP

```
ftp://login/path [ftptype]
```

dove:

- `login` é la stringa raccolta di utente, password e hostname per l'accesso ad un server FTP. Essa ha la forma  
`user:password@host`
- La mancanza di `user` e `password` fa partire automaticamente una connessione anonima
- `path` é il filename dell'oggetto ricercato
- `ftptype` regola i parametri di connessione ftp, come il tipo di trasferimento (ASCII o binario).

# NNTP

`news:group`

`news:articleID@host`

`nntp: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 protocollo `nntp`, viene usata scarsamente e solo per news server limitati privi di meccanismo di identificazione dei messaggi per `articleID`.