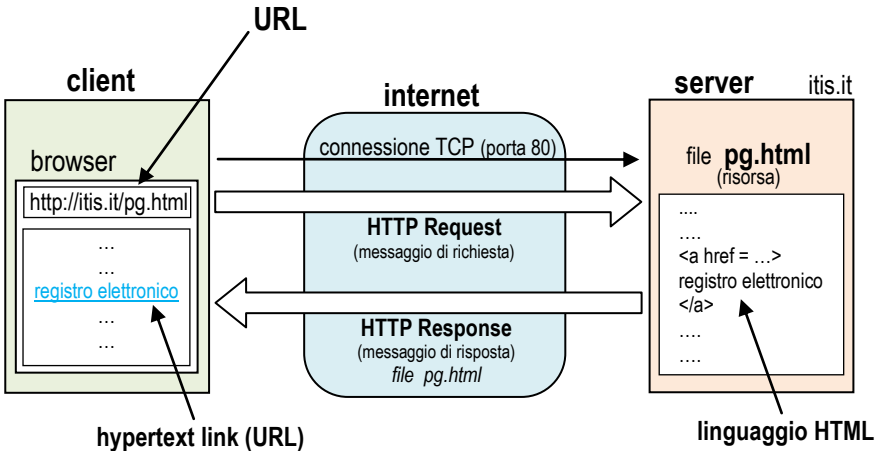


Protocollo HTTP: WWW, URL, Request e Response

<p style="text-align: center;">WWW (World Wide Web) e Navigazione Web Statica</p>	 <p>The diagram shows a client (browser) on the left and a server (pg.html) on the right, connected via the internet. The client sends an HTTP Request (messaggio di richiesta) to the server, and the server responds with an HTTP Response (messaggio di risposta) containing the file pg.html. The browser displays the content as a hypertext link (URL) and the server uses HTML language (linguaggio HTML).</p>
	<p style="text-align: center;"><u>WWW è basato su 3 elementi ...</u></p> <p>Protocollo HTTP (HyperText Transfer Protocol): protocollo, di livello applicativo, che regola lo scambio di dati fra il client (tipicamente un browser) e il server web che ospita il sito sul quale si naviga.</p> <p>Schema URL (Uniform Resource Locator): sequenza di caratteri che individua univocamente una risorsa (pagina web, immagine, video, documento, ecc.) su una rete (tipicamente, internet).</p> <p>Linguaggio HTML (HyperText Markup Language): linguaggio, basato su marcatori, che descrive la struttura di una pagina web: esso viene interpretato dal browser per comporre e visualizzare la pagina sullo schermo.</p>
<p style="text-align: center;">Schema URL (semplificato)</p>	<p style="text-align: center; color: red;"><protocollo> :// <dominio> / <risorsa></p> <p>Esempi:</p> <p>http://www.negozio.it/offerte.html</p> <p>http://www.negozio.it/archivio/schede/galaxyA50.html ... con percorso</p> <p>http://80.210.94.71/contatti.html ... con ind. IP</p> <p>http://www.negozio.it ... home page</p> <p><protocollo> Indica il protocollo da usare nella comunicazione con il server: nella navigazione web il protocollo è sempre http, ma lo schema URL è usato anche con altri protocolli.</p> <p><dominio> Indica il nome di dominio che individua univocamente il sito web da contattare. Tramite il protocollo DNS, il browser può tradurre il nome di dominio nell'indirizzo IP del server web che ospita il sito desiderato e procedere quindi alla connessione. E' possibile specificare anche direttamente l'indirizzo IP, anziché il nome di dominio.</p> <p><risorsa> Specifica il nome della risorsa che si desidera ottenere (pagina web, immagine, ecc.). Se il sito è strutturato in cartelle, è possibile specificare il "percorso" della risorsa. Se la risorsa non viene specificata, il server web invierà la Home Page del sito.</p>

<p>HTTP Request (messaggio di richiesta)</p>	<p>Il Client apre una connessione TCP con il Server Web sulla porta 80 e invia il Messaggio di Richiesta (HTTP Request) ...</p> <p><u>Struttura del Messaggio di Richiesta</u></p> <p>E' una sequenza di caratteri organizzata per righe di testo e strutturata in 4 parti:</p> <p>Riga di Richiesta Sezione Header (<i>informazioni aggiuntive</i>) Riga vuota Corpo del Messaggio (<i>body</i>)</p>	
<p>HTTP Request: Riga di Richiesta e Metodi HTTP</p>	<p><i><metodo> <risorsa (con eventuale percorso)> <versione del protocollo></i></p> <p>Esempi: GET /archivio/schede/galaxyA50.html HTTP/1.1 POST /elaboraDati.php HTTP/1.1</p>	
		<p><metodo> Specifica il tipo di operazione richiesta. I metodi previsti dal protocollo HTTP sono GET, POST, HEAD, PUT, DELETE, TRACE, OPTIONS, CONNECT ma i più utilizzati sono senz'altro i seguenti:</p> <p>metodo GET: si utilizza per richiedere l'invio di una risorsa, tipicamente una pagina web. (può essere usato anche per inviare dati al server, ma con limitazioni che vedremo in seguito).</p> <p>metodo POST: si utilizza per inviare dati dal client al server: ad esempio, i dati di un "form" compilato dall'utente con il browser. (la risorsa specificata in un POST deve essere una "pagina dinamica" ossia una pagina che prevede l'esecuzione di codice sul server per poter processare i dati ricevuti. Si parla, in questi casi, di programmazione web dinamica o programmazione lato server)</p> <p>metodo HEAD: si utilizza ottenere informazioni su una risorsa. Il server, anziché rispondere inviando la risorsa, ne invia solo l'header, ossia solo le informazioni aggiuntive.</p> <p><risorsa> Indica la risorsa che si desidera richiamare, completa di eventuale percorso. Specificando sola una "barra" (/) si richiede la Home Page del sito.</p> <p><versione del protocollo> Specifica la versione del protocollo HTTP da usare nella comunicazione. Oggi è spesso utilizzata la versione 1.1 per cui questo elemento, tipicamente, è: HTTP/1.1</p>
<p>HTTP Request: Sezione Header</p>	<p>Contiene Informazioni Aggiuntive (Header), che aiutano il server a soddisfare la richiesta. Ogni Header è disposto su una riga, secondo la forma: <i><header>: <valore></i></p> <p>host (obbligatorio): Nome di Dominio del sito desiderato. Necessario perché a partire dalla versione 1.1, HTTP permette che un singolo server web ospiti più di un dominio. I possibili Header sono tanti e saranno esaminati progressivamente in seguito.</p>	
<p>HTTP Request: Riga Vuota e Corpo</p>	<p>Una Riga Vuota fa da separatore fra la <i>Sezione Header</i> e il <i>Corpo</i> della Request.</p> <p>Il Corpo contiene i <i>dati da inviare</i> al server:</p> <p><i>Riga Vuota e Corpo</i> sono utilizzati solo nel metodo POST che serve appunto per l'invio di dati al server. Request con i <i>metodi GET e HEAD</i> non hanno Riga Vuota e Corpo.</p>	
<p>Esempi di HTTP Request con metodo GET</p>	<p>GET / HTTP/1.1 host: danielesirangelo.altervista.org</p>	<p>GET /archivio/schede/galaxyA50.html HTTP/1.1 host: www.samsung.it</p>

HTTP Response (messaggio di risposta)	<p>... il Server riceve il messaggio di richiesta e risponde inviando al Client un Messaggio di Risposta (HTTP Response) contenente la risorsa richiesta.</p> <p><u>Struttura del Messaggio di Risposta</u></p> <p>E' una sequenza di caratteri organizzata per righe di testo e strutturata in 4 parti:</p> <p>Riga di Stato Sezione Header (<i>informazioni aggiuntive</i>) Riga vuota Corpo del Messaggio (<i>body</i>)</p>										
HTTP Response: Riga di Stato e Codici di Stato HTTP	<p><i><versione del protocollo> <codice di stato HTTP> <descrizione></i></p> <p>Esempi:</p> <table><tr><td>HTTP/1.1</td><td>200</td><td>OK</td></tr><tr><td>HTTP/1.1</td><td>400</td><td>Bad Request</td></tr><tr><td>HTTP/1.1</td><td>404</td><td>Not Found</td></tr></table>		HTTP/1.1	200	OK	HTTP/1.1	400	Bad Request	HTTP/1.1	404	Not Found
	HTTP/1.1	200	OK								
HTTP/1.1	400	Bad Request									
HTTP/1.1	404	Not Found									
	<p><versione del protocollo> specifica la versione del protocollo HTTP usata nella comunicazione.</p> <p><codice di stato HTTP> codice numerico che riporta l'esito della richiesta. Il protocollo HTTP prevede una serie di codici standard a 3 cifre in cui la prima cifra indica il <i>tipo di esito</i>:</p> <p>1xx: Informational (messaggi informativi) 2xx: Successful (la richiesta è stata soddisfatta) 3xx: Redirection (non c'è risposta immediata, ma "reindirizza" su un'altra risorsa) 4xx: Client error (la richiesta non può essere soddisfatta perché sbagliata) 5xx: Server error (la richiesta non può essere soddisfatta per un problema del server)</p> <p>Esiti particolarmente ricorrenti sono: 200 (OK - risorsa inviata correttamente); 400 (Bad Request - la richiesta non è comprensibile); 404 (Not Found - risorsa non trovata); ecc.</p> <p><descrizione>: breve descrizione del codice di stato (OK, Bad Request, Not Found, ecc.)</p>										
HTTP Response: Sezione Header	<p>Contiene Informazioni Aggiuntive (Header), con informazioni sulla risposta inviata. Ogni Header è disposto su una riga, secondo la forma: <i><header>: <valore></i></p> <p>server: indica il software per Web Server utilizzato (Apache, Microsoft IIS, ecc.) content-length: lunghezza, in byte, dei dati inviati nel corpo della risposta content-type: tipo dei dati inviati, secondo lo standard MIME (Multimedia Internet Mail Extensions). esempi: text/html (pag.web), text/plain (testo), text/xml (dati in xml), image/jpeg, ecc.</p> <p>I possibili Header sono tanti e saranno esaminati progressivamente in seguito.</p>										
HTTP Response: Riga Vuota e Corpo	<p>Una Riga Vuota fa da separatore fra la <i>Sezione Header</i> e il <i>Corpo</i> della Request. Il Corpo contiene i <i>dati inviati</i> dal server al client in risposta alla Request inoltrata (ad esempio, il codice HTML della pagina web richiesta).</p>										
Esempi di HTTP Response	<p>HTTP/1.1 200 OK server: Apache/2.4.41 content-length: 325 content-type: text/html</p> <p><html> ... codice html della <i>pagina richiesta</i> ... </html></p>	<p>HTTP/1.1 400 Bad Request server: Microsoft IIS/7.0 content-length: 220 content-type: text/html</p> <p><html> ... codice html della <i>pagina di errore</i> ... </html></p>									