

Progettazione Web



Alessandra Degan Di Dieco

Analyst @icubedsrl

alessandra.degan@icubed.it



Summary

- ❖ Cenni di reti di calcolatori
- ❖ Cenni sul funzionamento del browser
- ❖ Cosa sono i linguaggi di markup: HTML, XHTML, XML
- ❖ Struttura di una pagina HTML e Ispezione dell'output HTML
- ❖ Sviluppo web moderno
 - ❖ Organizzazione grafica con CSS3
 - ❖ Progressive Enhancement
 - ❖ Tecniche di responsive design
- ❖ Nozioni di sviluppo in Javascript
- ❖ Framework

Introduzione a infrastrutture di Rete



Alessandra Degan Di Dieco
Analyst @icubedsrl
alessandra.degan@icubed.it

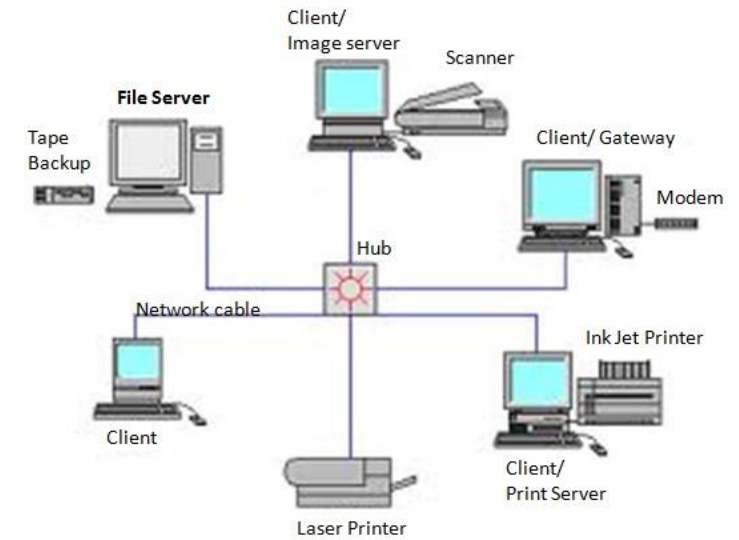
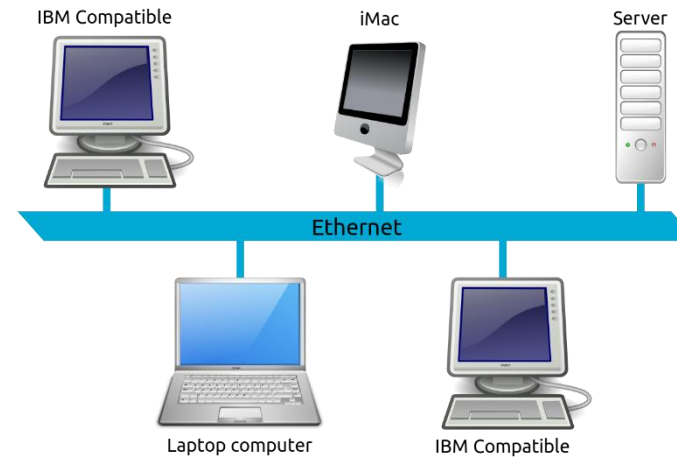


Reti di computer

Infrastrutture che permettono di collegare dispositivi permettendo scambio di dati e informazioni

Host: dispositivo connesso alla rete

Trasmissione visibile a tutti gli host collegati



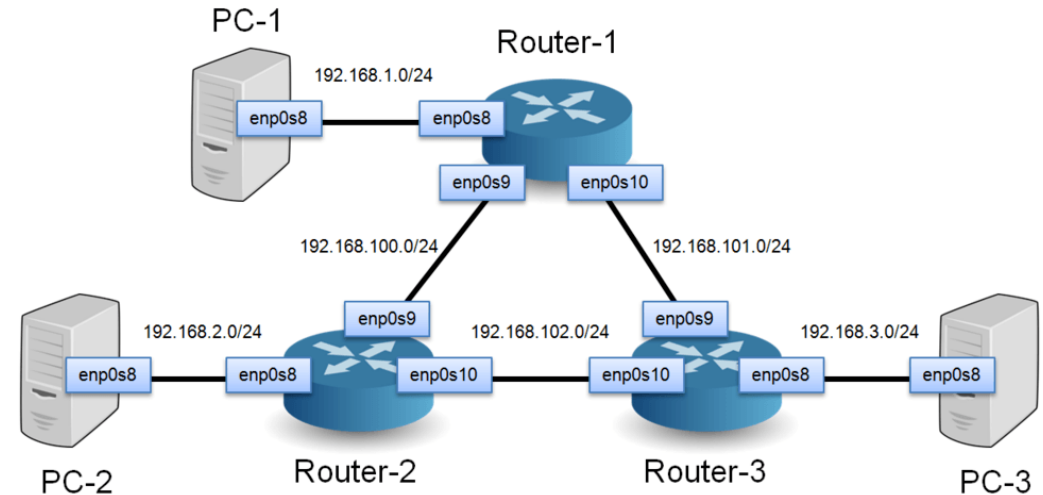
Reti di computer

Router: instrada i messaggi

Una rete di computer può essere collegata a un'altra tramite un Router

↑ dimensione e complessità

→ Classificazione delle reti per estensione geografica



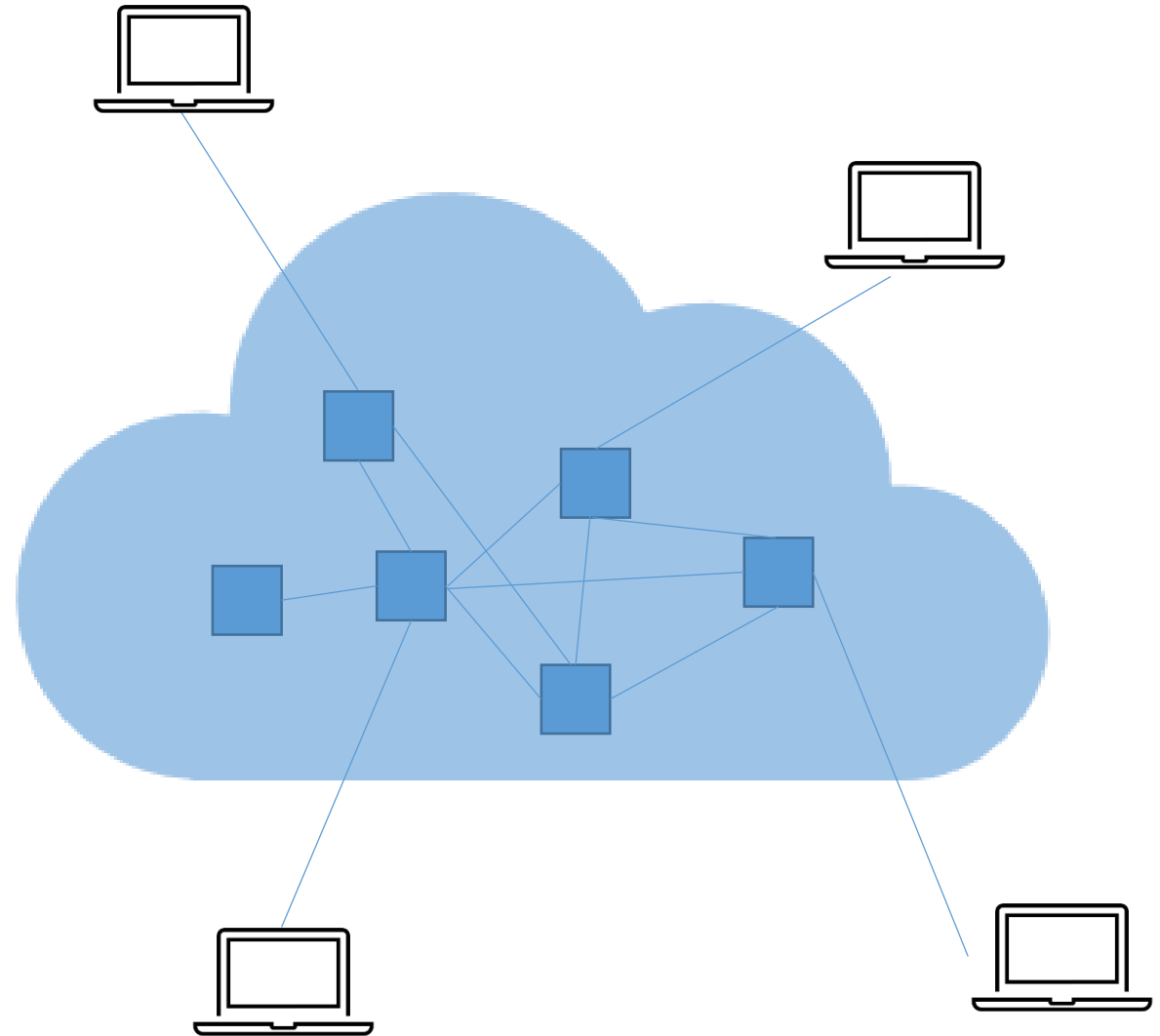
Internet

‘ Rete di reti ‘

interconnessione globale tra reti di natura ed estensione diversa, resa possibile da protocolli di rete (‘TCP/IP’)



Milioni di host, collegati da svariati mezzi trasmissivi, attraverso nodi di commutazione (es. router)

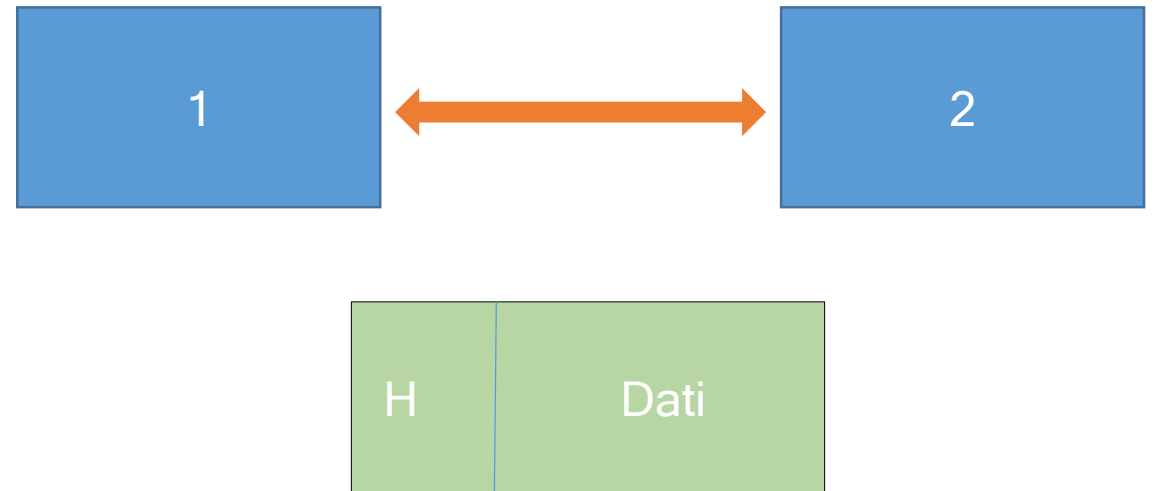


Protocolli di comunicazione

Un insieme di regole per lo scambio di messaggi tra due entità

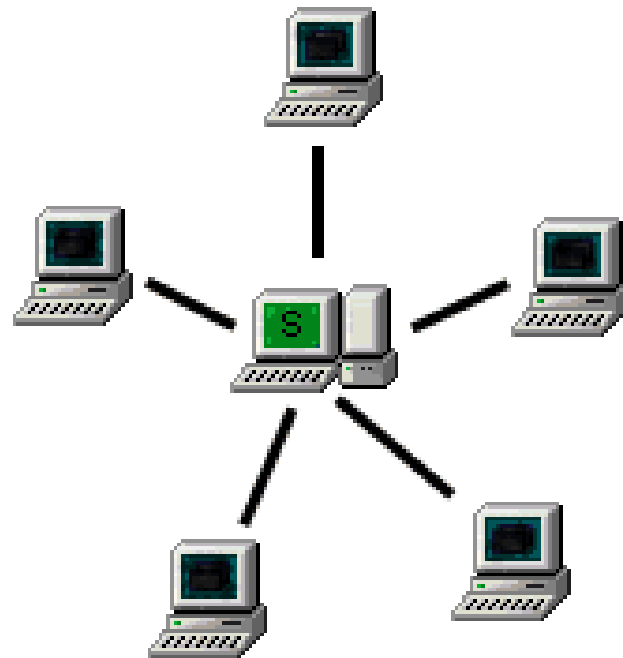
Nel caso che le entità siano computer, il messaggio sarà diviso in:

- Header
- Dati

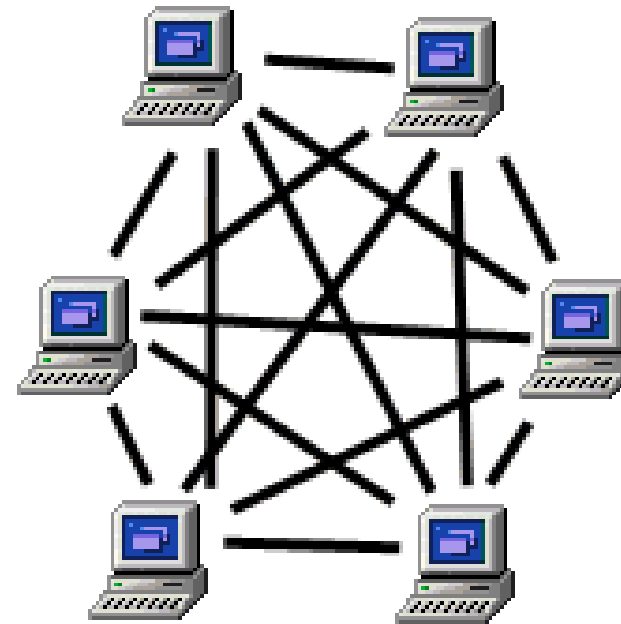


Differenza tra rete Server Based e rete Peer to Peer (P2P)

Server Based Network



Peer to Peer Network



Tipi di Reti

Commutazione di circuito

- Risorse riservate
- Connessione tra interlocutori dedicata per tutta la durata della comunicazione

Es: chiamata



Commutazione di pacchetto

- Risorse condivise
- Messaggio suddiviso in pacchetti → instradati dal router su link di comunicazione diversi → messaggio riassembleato dal destinatario



Tipi di Reti

Commutazione di circuito

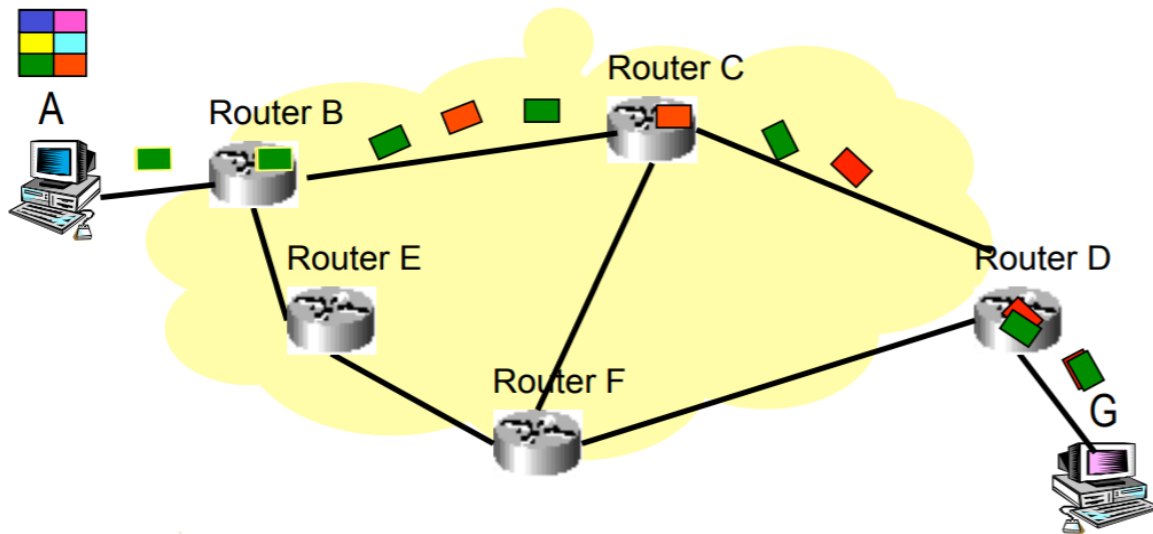


Commutazione di pacchetto

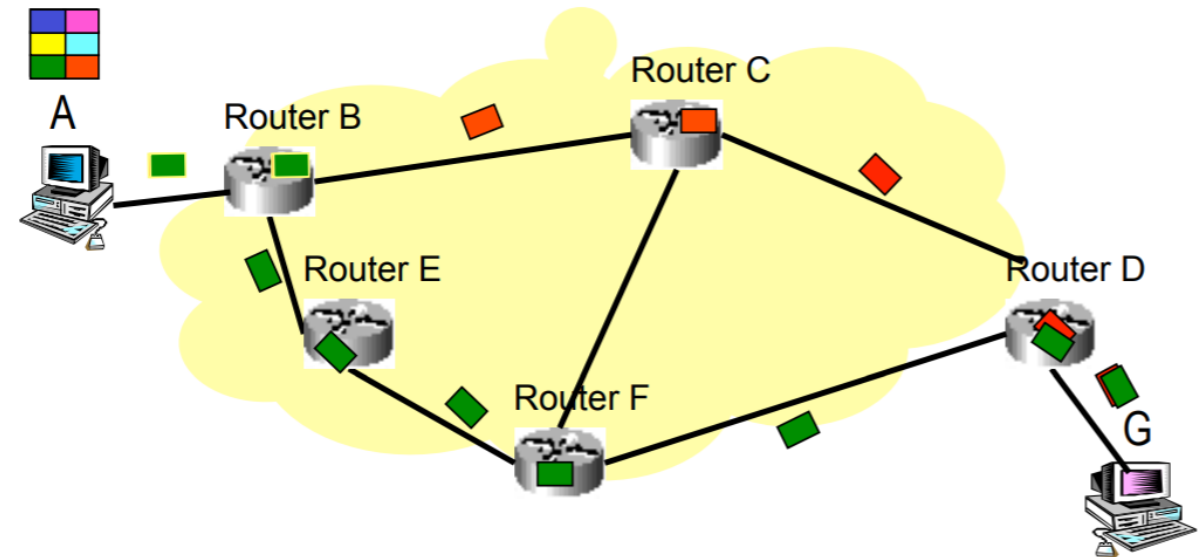


Servizi delle reti a commutazione di pacchetto

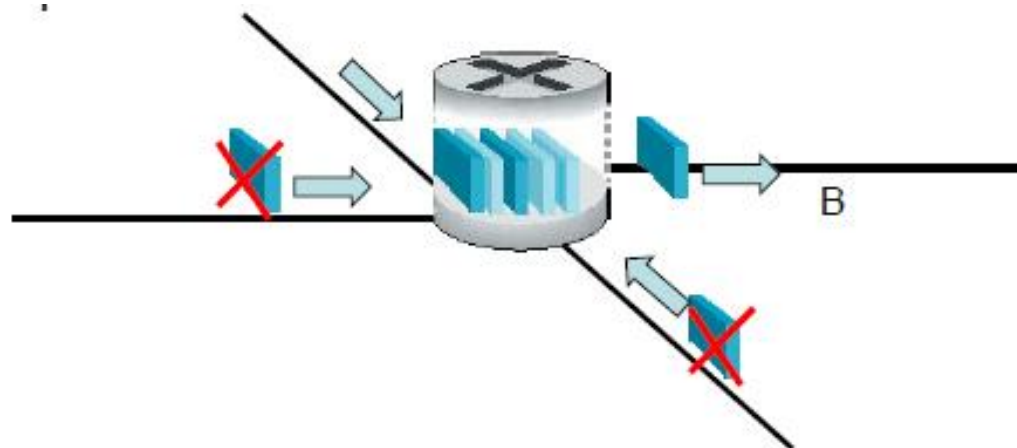
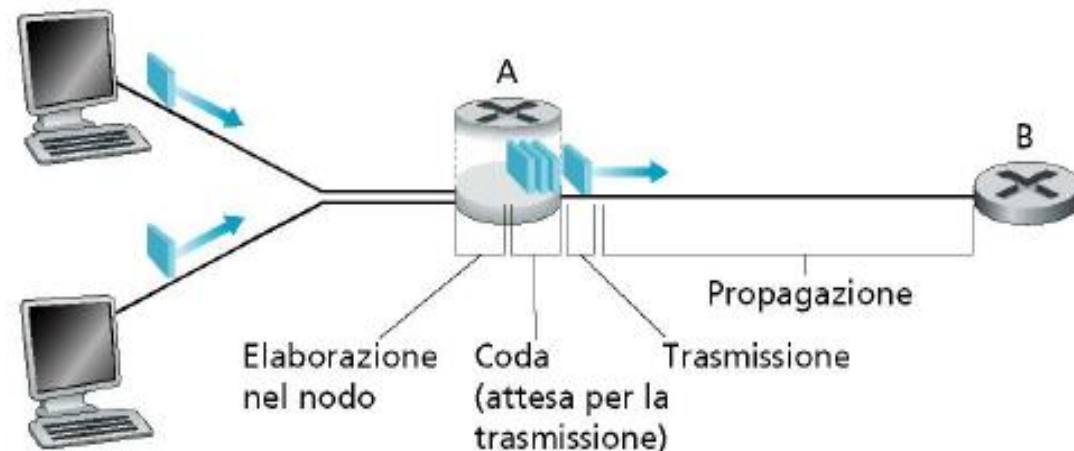
Circuito virtuale



Datagram



Qualità del Servizio - Ritardi e perdite



Ritardo da host a host

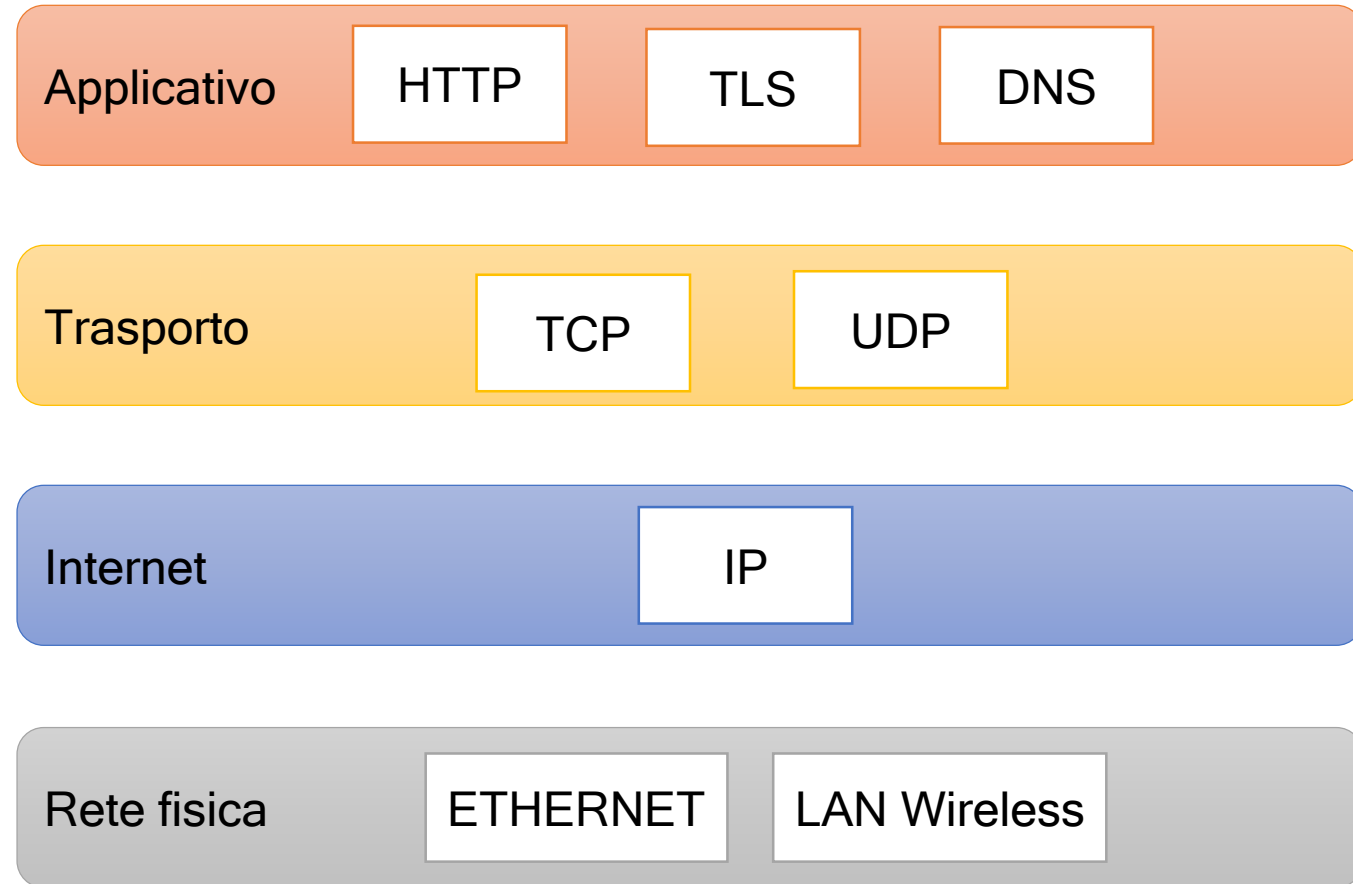
=

Somma ritardi dei singoli
nodi

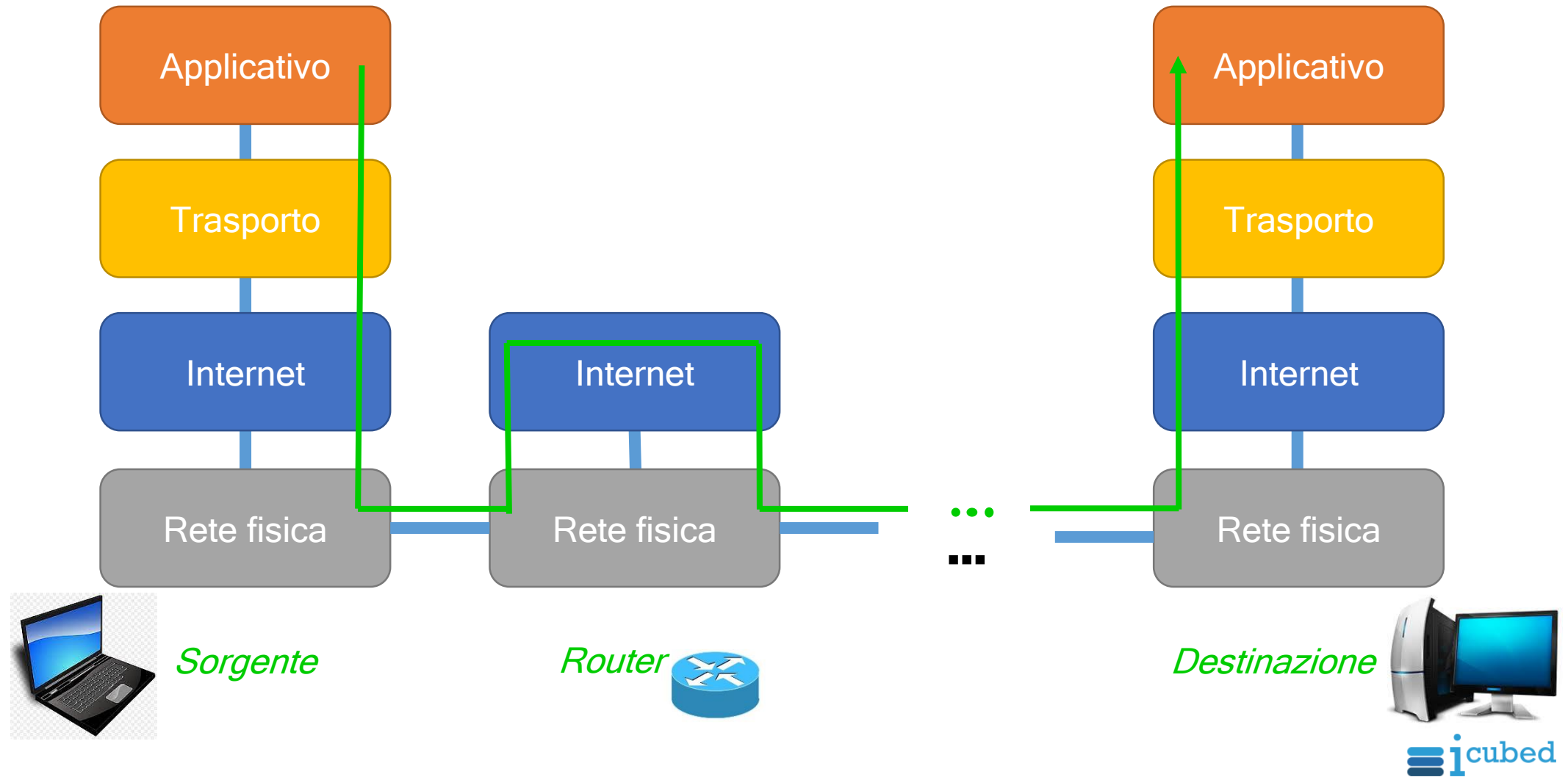
Coda piena → pacchetto
non memorizzato →
perdita del pacchetto

MA può essere ritrasmesso
(?)

Pila protocollare di Internet



Protocolli e Trasmissione



Internet Protocol

IP: sequenza che identifica univocamente i dispositivi connessi alla rete

Assegnazione dell'IP:

- *Statica*
- *Dinamica*

IPv4

32 bit: 4 interi tra 0-255

$2^{32} \sim 4$ miliardi di combinazioni

Parte di rete			Parte host
IPv4:	192.168.178.31		
8 Bit	8 Bit	8 Bit	8 Bit

IPv6

128 bit: 8 interi fra 0 e 2^{16}

2^{128}

Prefisso di rete				Interface Identifier			
IPv6:	0000:0000:0000:0000:0000:ffff:c0a8:b21f						
16 Bit	16 Bit	16 Bit	16 Bit	16 Bit	16 Bit	16 Bit	16 Bit

TCP vs UDP

- trasferimento dei messaggi *affidabile*
- controllo della congestione
- frammenta i messaggi e li riassembla nella destinazione

TCP

- trasferimento dei messaggi *non affidabile*
- NO controllo della congestione
- Messaggi frammentati a livello applicazione
- Es. per applicazioni telefonia internet

UDP

Domain Name System (DNS)

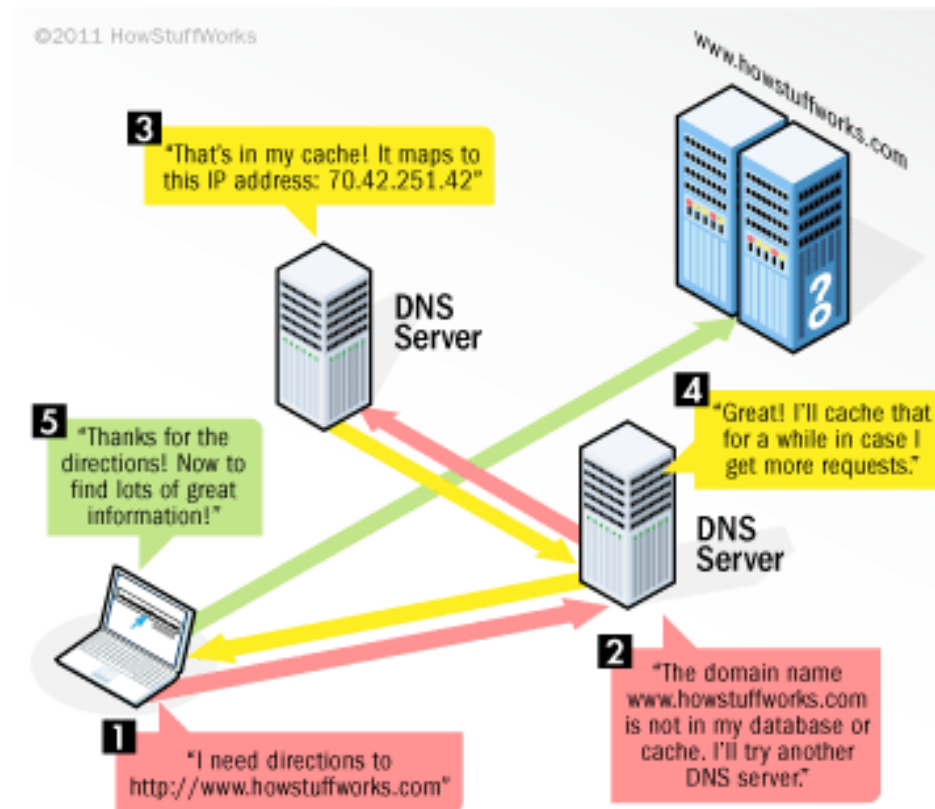
Solitamente non utilizziamo gli indirizzi IP per accedere a un particolare servizio → All'indirizzo IP viene associato un nome simbolico.

www.google.com

La corrispondenza viene conservata dal DNS server.

DNS

DNS è un Sistema composto da un database distribuito che mappa il nome simbolico all'IP



DNS

www.support.microsoft.com.
www.informatica.uniroma2.it.



Struttura gerarchica

Livello 1: Top Level Domain

Gestito dall'organizzazione IANA

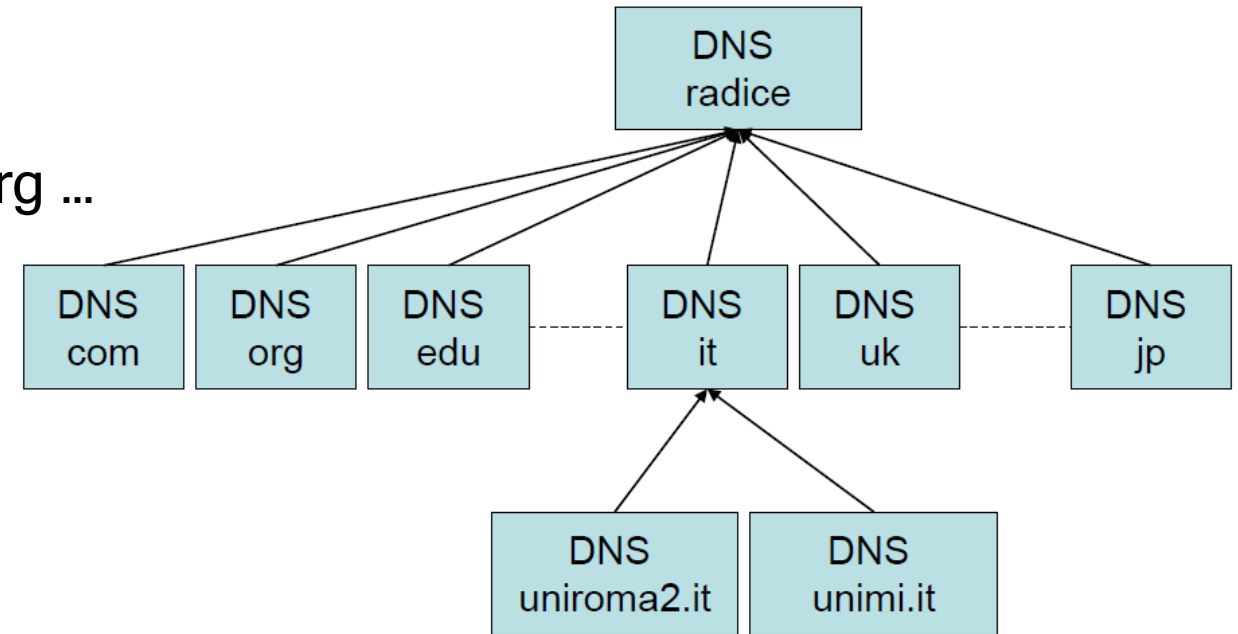
Definizione dei suffissi: .it, .com, .gov, .org ...

Livello 2: Second level Domain

Nome (Es. Microsoft)

Livello 3: Host

Da qui è organizzato internamente



Introduzione al Web



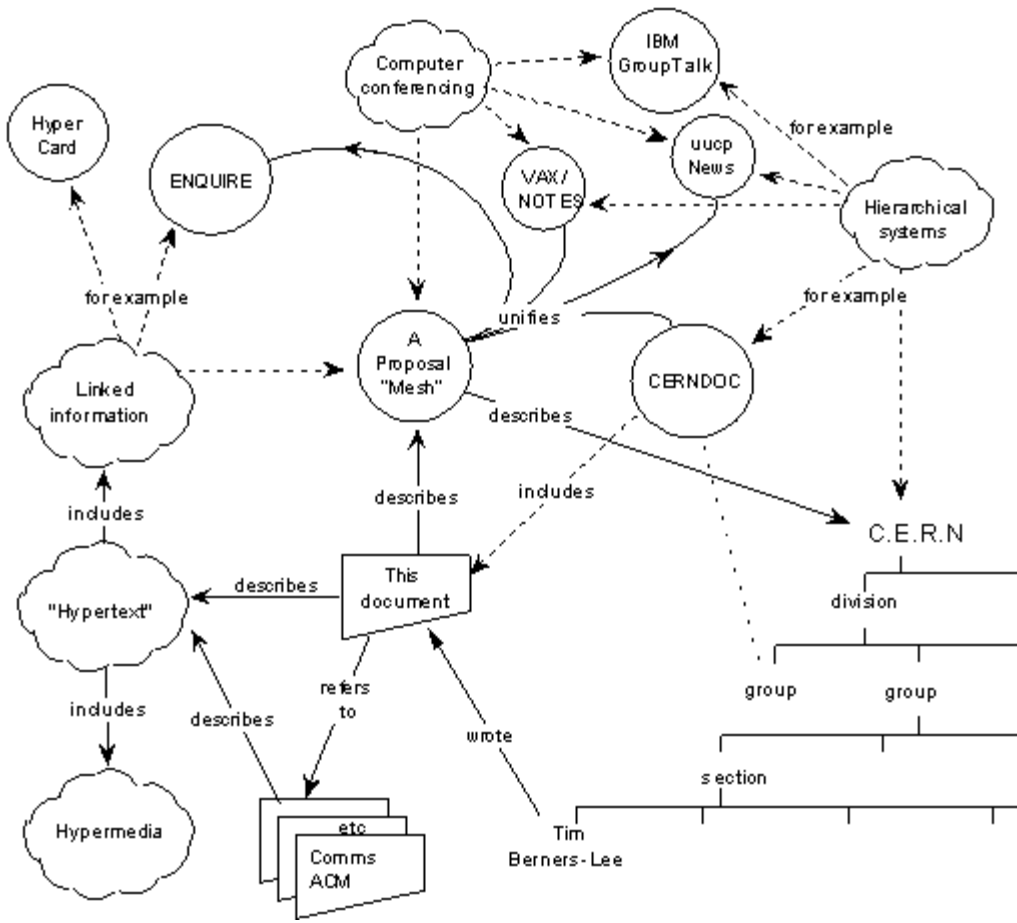
Alessandra Degan Di Dieco

Analyst @icubedsrl

alessandra.degan@icubed.it



World Wide Web



1989 - Proposta

Tim Berners-Lee, CERN

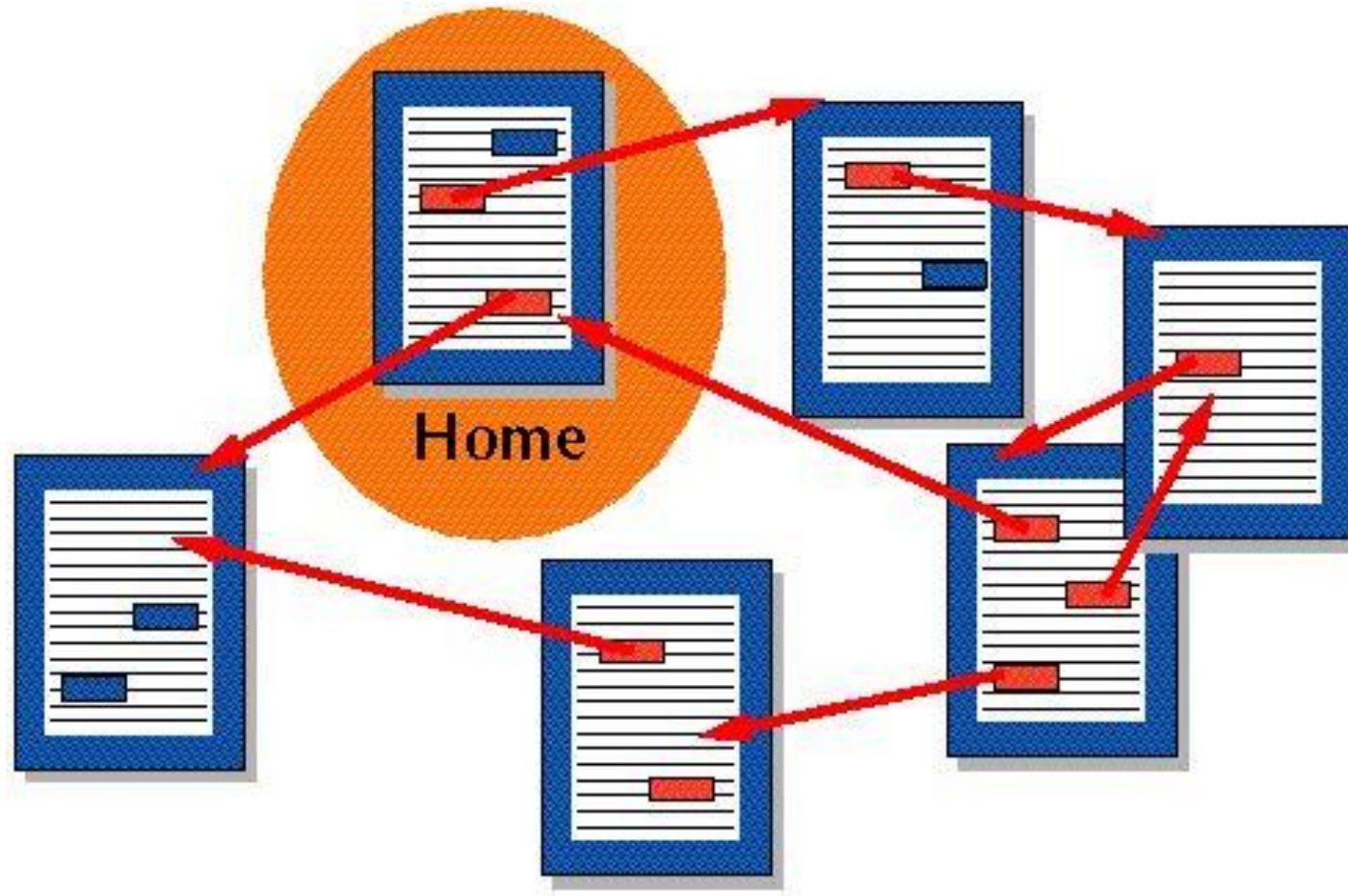
- Condivisione della documentazione scientifica tra ricercatori dell'istituto

1991 - Primo sito web

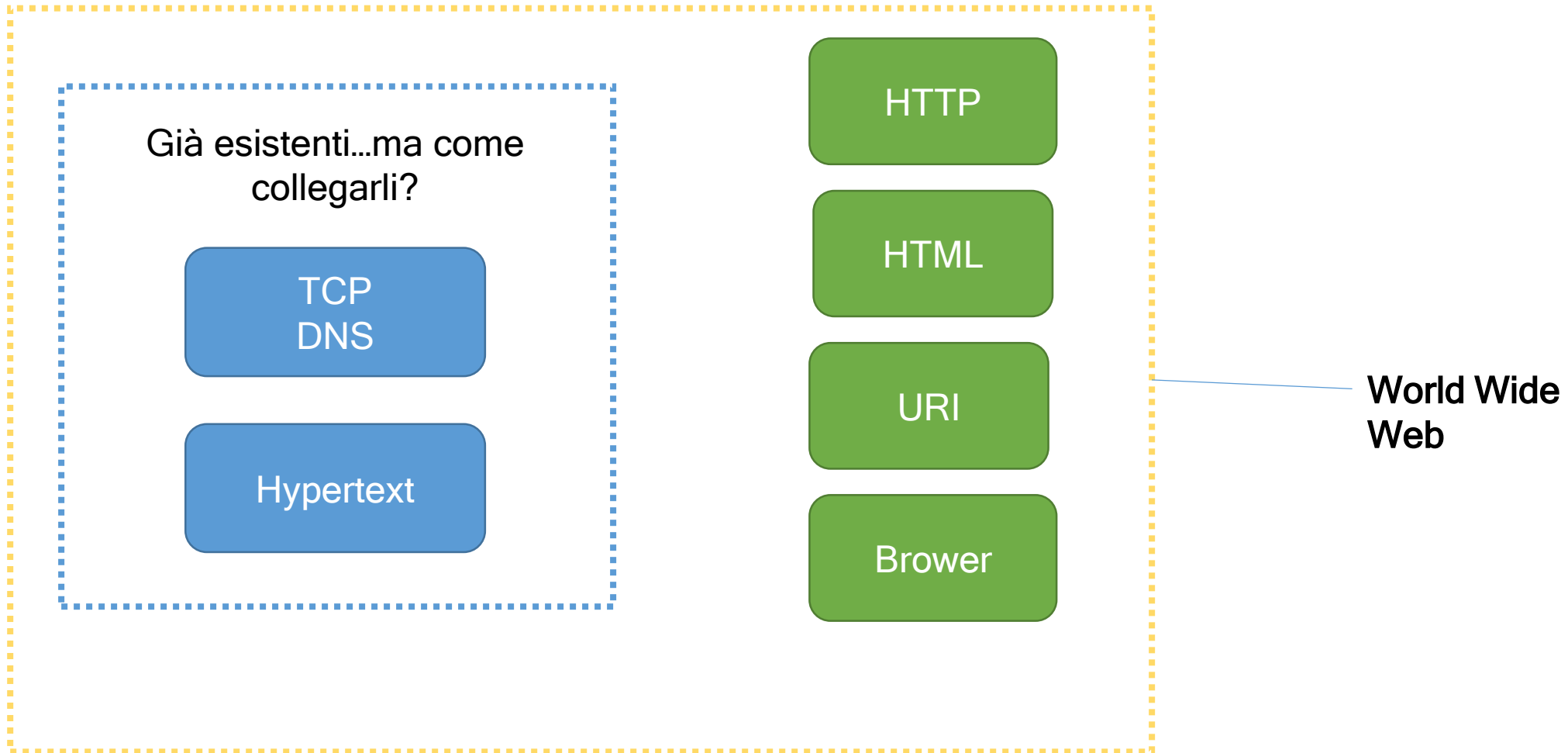
Tim Berners-Lee, CERN

1993 - CERN rilascia codice sorgente

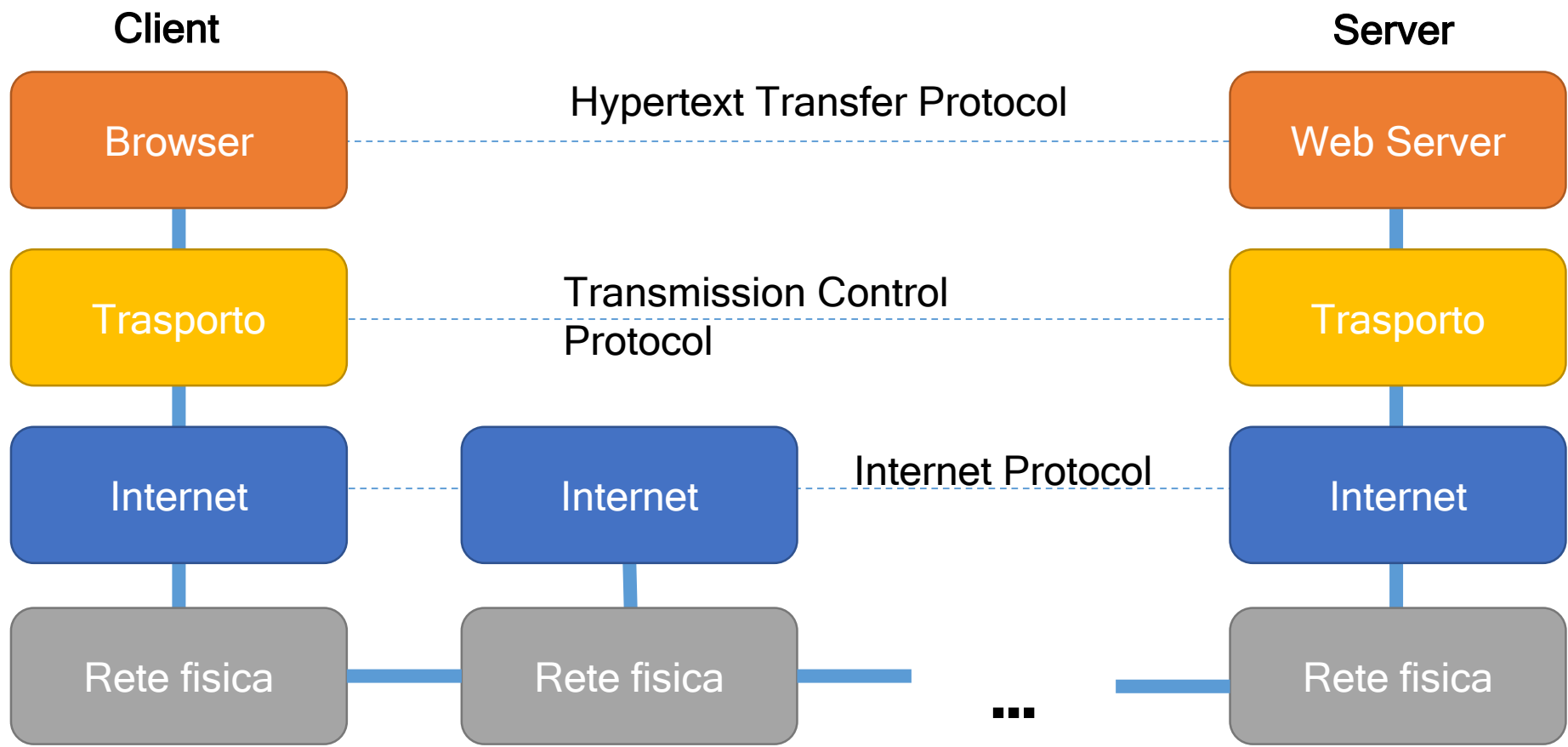
Hypertext



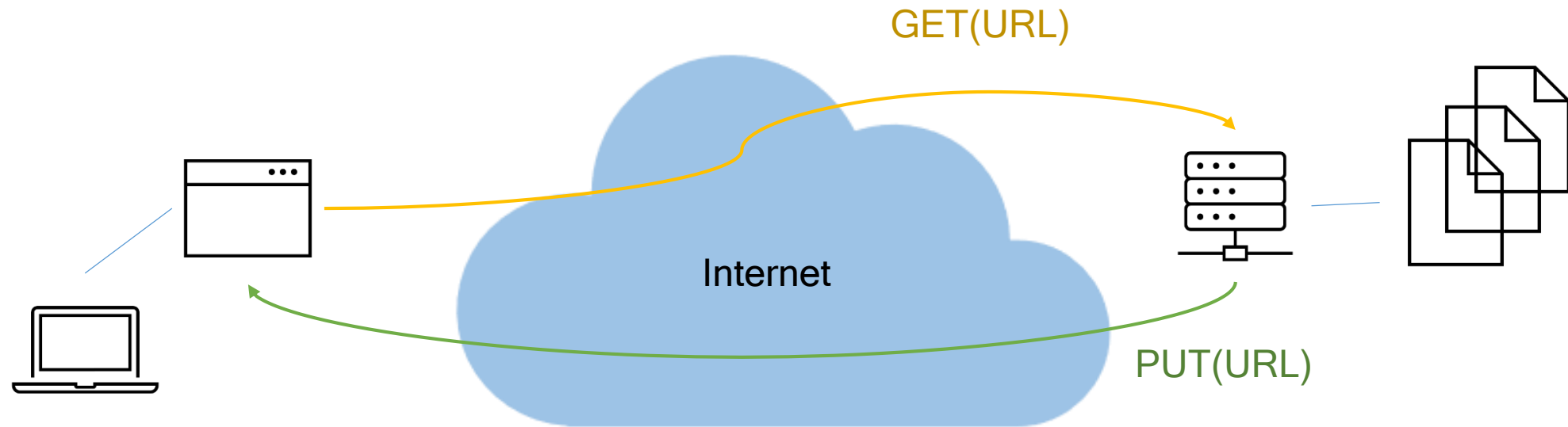
World Wide Web



Protocollo HTTP



Funzionamento HTTP



URI, URL, URN

URI: Uniform Resource Identifier

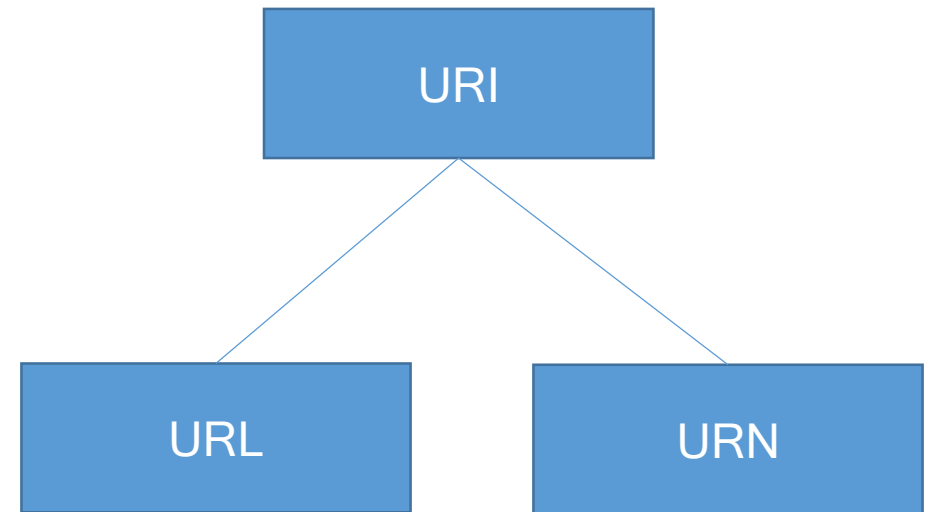
Concetto generale per identificazione sia in web che per altro.

URL: Uniform Resource Locator

Definizione dell'indirizzo della risorsa

URN: Uniform Resource Name

Definizione del nome della risorsa



URL

<http://www.miosito.it/A/index.html>

http://
schema

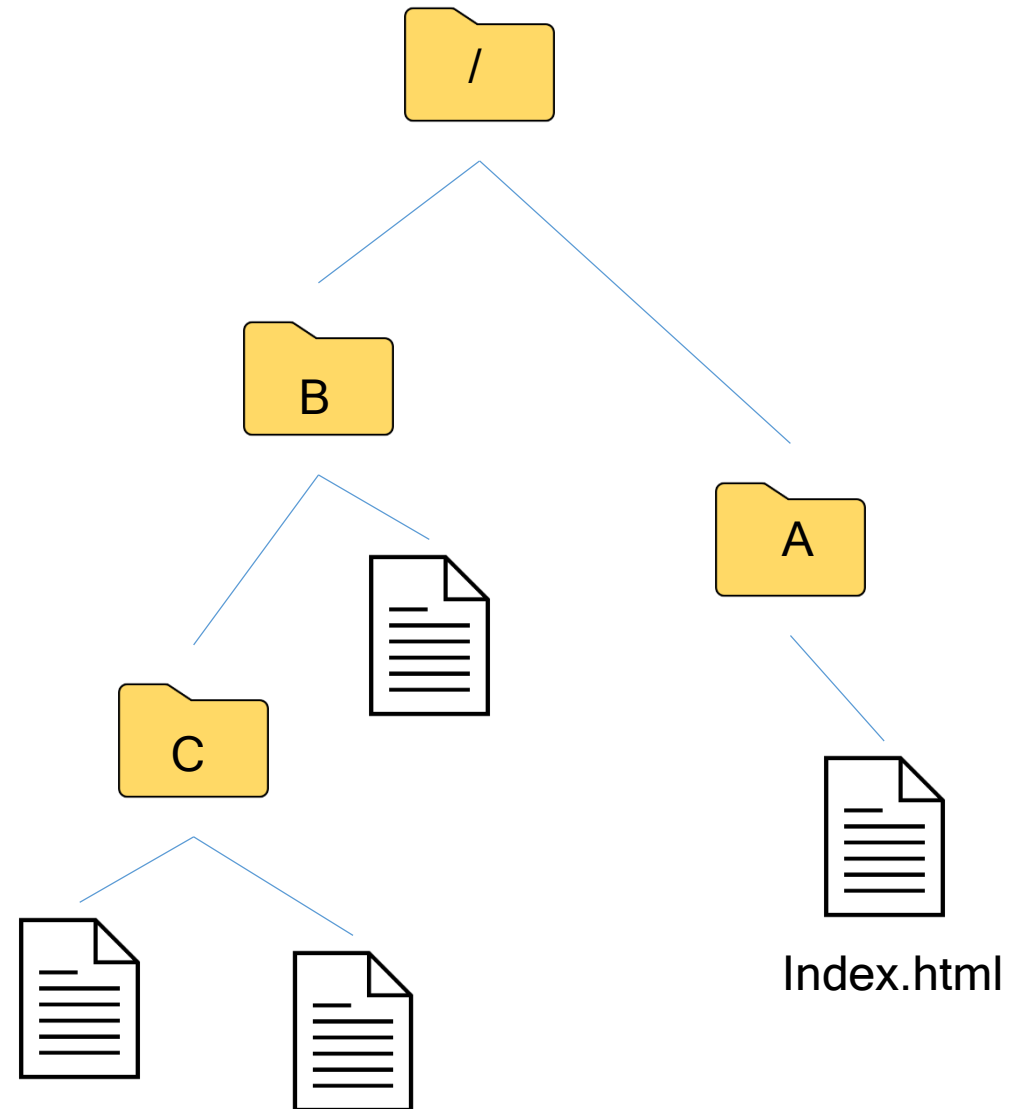
Nome dello

www.miosito.it

Nome del dominio

A/index.html

Nome file



HTML5, CSS & Javascript



Alessandra Degan Di
Dieco

Analyst @icubedsrl

alessandra.degan@icubed.it



Linguaggio HTML (HyperText Markup Language)

Si basa sulla distinzione tra contenuto, struttura logica e modalità di presentazione delle pagine.

Contenuto della pagina è dato da :

1. Informazioni
2. Meta-informazioni

Il rendering della pagina è gestito dal browser

Cos'è HTML5

HyperText Markup Language
(linguaggio di marcatura di ipertesti)

Linguaggio utilizzato per creare pagine web e
altri tipi di documenti visualizzabili in un
browser

Standard internazionale le cui specifiche sono
manutenute dal World Wide Web Consortium
(W3C)



Programmazione e Markup

Linguaggio di programmazione:

Istruisce una macchina di calcolo

Linguaggio di Markup:

Annota un document in modo che la logica della pagina sia divisa dal contenuto

World Wide Web Consortium (W3C)



Organizzazione internazionale che ha lo scopo di definire gli standard web e di sviluppare al massimo le potenzialità del World Wide Web

Guidato dall'inventore del Web Tim Berners-Lee



Sito italiano <http://www.w3c.it>

Un po' di storia

L'html è uno standard che è stato sviluppato e che si va a perfezionare nel tempo:

1990: HTML tags

1993: HTML 2.0

1995: HTML 3.0

1997: HTML 4.0

1999: HTML 4.01

- 2000: XHTML 1.0

Un gruppo del W3C (WHATWG) si separa e dà vita a delle nuove specifiche che definiscono l'HTML5

- 28 ottobre 2014: Il consorzio riconosce ufficialmente che HTML5 è a tutti gli effetti uno standard Web

Riferimenti

W3C Recommendation HTML5

- www.w3.org/TR/html5

W3C Recommendation HTML5: Edition for Web Authors

- www.w3.org/TR/html5-author

WHATWG community: HTML Living Standard

- <https://html.spec.whatwg.org/multipage>

Mozilla Developer Network (*in italiano!*)

- <https://developer.mozilla.org/it/docs/Web/HTML>

Si può usare HTML5 oggi?

Certo! Tramite apposito DocType

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```

HTML
4.01

→

```
<!DOCTYPE html>
```

HTML 5
(Semplificato!)

I browser ignorano i tag che non riconoscono

Struttura generale

Un document HTML è una **struttura ad albero** composta da elementi e testo.

Elemento: composto da una coppia di tag.

Tag: definisce un'informazione di natura strutturale con significato semantico particolare.

Il tag può contenere sia testo che altri elementi.

L'insieme di elementi di un testo viene detto markup.

Head e Body

<head></head>

Utilizzato per aggregare i metadati relativi alla pagina.

Riguarda soprattutto informazioni rispetto alle impostazioni della pagina

<body></body>

Rappresenta il corpo della pagina: comprende il contenuto di cui il browser deve fare rendering

Attributi

Il tag di apertura di un document può essere dotato di attributi particolari.

Gli attribute sono proprietà in coppia chiave-valore che danno informazioni aggiuntive sulle specifiche del tag.

```
<canvas id="html5_canvas" width="200" height="100">
```

Document Object Model


Il browser interpreta l'intera pagina html. Esegue:

1. Il parsing del markup
2. La rappresentazione ad albero della struttura del documento html (DOM)

Il DOM si compone di tutti gli elementi presenti nel markup, a partire dal tag `<html>`.

Come faccio a sapere cosa posso usare oggi?

In fase di progettazione

Download attribute  Global 58.7%

When used on an anchor, this attribute signifies that the browser should download the resource the anchor points to rather than navigate to it.

[current signed](#) [Linkage relative](#) [Show all](#)

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Chrome for Android
8		38	31					4.1	
9		39	43			7.1		4.3	
10		40	44		31	8.4		4.4.4	
11	12	41	45	8	32	9	8	44	44
	13	42	46	9	33				
		43	47		34				
		44	48						

Can I Use ... ?

<http://www.caniuse.com/>

Programmaticamente



Modernizr

<http://www.modernizr.com/>

Novità HTML5

Con HTML5, dobbiamo usare

- HTML per il markup e la semantica
- CSS per lo stile
- Javascript per l'interazione

DocType semplificato

- `<!DOCTYPE html>`
Identifica la tipologia di documento che il browser andrà a renderizzare

Nuovi tag che danno un significato semantico alla pagina

Strutture a confronto

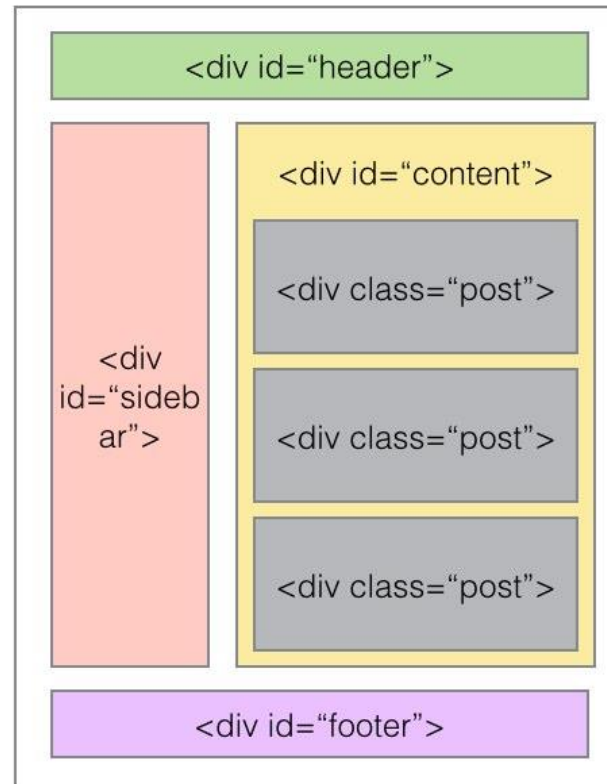
HTML4 vs HTML5 Page Structure on a Blog

Da questa immagine notiamo cosa si intende quando si parla di HTML5 come linguaggio *semantico*

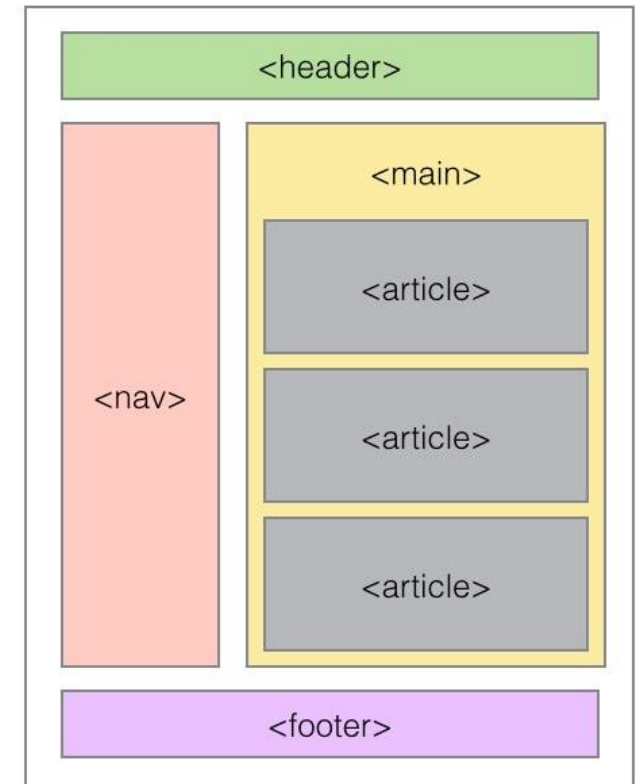
I tag semantici sono paragonabili a `<div>` che racchiudono particolari tipi di contenuti della pagina.

Non sono un requisito, ma aiutano il browser e i motori di ricerca a determinare già solo dalla struttura dell'HTML, il contenuto di una pagina Web

HTML4: Lots of Classes/IDs



HTML5: Semantic Tags/Sections



Tag <header>

Rappresenta un blocco di intestazione per una pagina o una sezione
(es. titolo e data di pubblicazione di un articolo)

```
<header>  
  <h1>Google buys Nestz</h1>  
  <p>Posted at 11:34am 13° January 2014</p>  
</header>
```

Tag <nav>

Segnala un menu di navigazione (insieme di link)

```
<nav>  
  <ul>  
    <li><a href="#chapter-one">Chapter One</a>  
    <li><a href="#chapter-two">Chapter Two</a>  
    <li><a href="#chapter-three">Chapter  
Three</a>  
  </ul>  
</nav>
```

Tag <main>

Dovrebbe racchiudere il contenuto principale della pagina.
Deve essere unico nell'intera pagina.
Tutti i contenuti ripetuti (come loghi, caselle di ricerca, link, ecc..) non devono esservi inseriti.

```
<body>
  <header>
    <div id="logo">Rocking Stone</div>
    <nav>...</nav>
  </header>
  <main role="main">
    <h1>Guitars</h1>
    <p>The greatest guitars ever built.</p>

    <article>
      <h2>Gibson SG</h2>
      <p>...</p>
    </article>

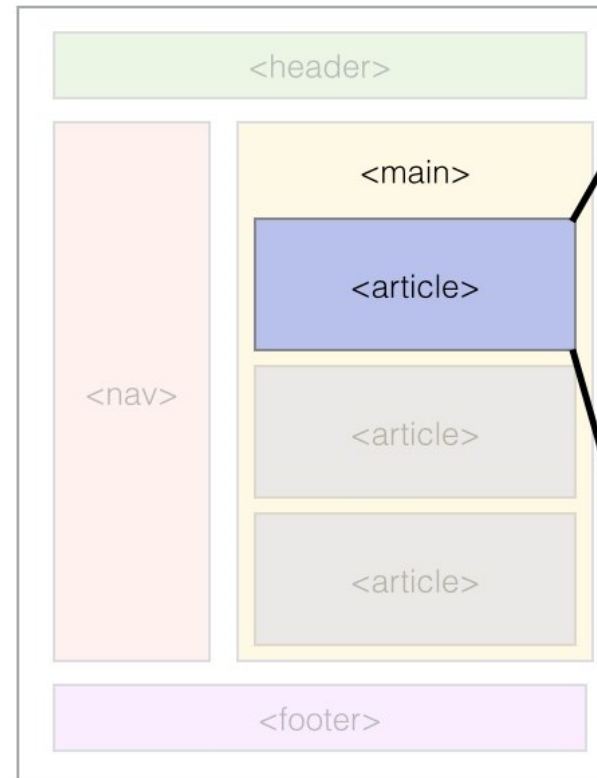
    <article>
      <h2>Fender Telecaster</h2>
      <p>...</p>
    </article>
  </main>
</body>
```

Tag <section> e <article>

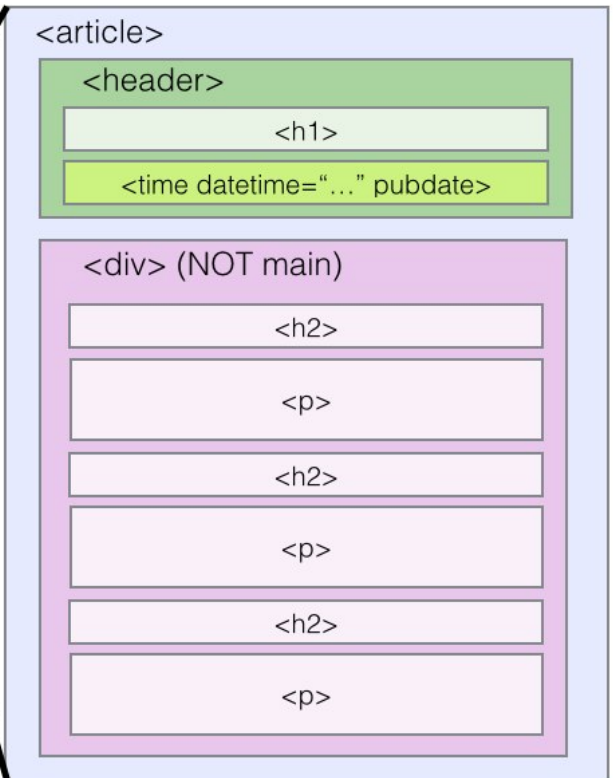
```
<article>
  <h1>How to use HTML5 Sectioning Elements</h1>
  <p>...</p>

  <section>
    <h2>The <main> Element</h2>
    <p>...</p>
  </section>
  <section>
    <h2>The <article> Element</h2>
    <p>...</p>
  </section>
  <section>
    <h2>The <section> Element</h2>
    <p>...</p>
  </section>
  ...
</article>
```

High Level Blog Index



Inside a Specific Post



Tag <footer>

Identifica un footer; contiene generalmente informazioni riguardo l'autore e copyright

```
<footer>
```

```
    Copyright Matt West 2014
```

```
</footer>
```

Tag <figure> e <figcaption>

Rappresenta un blocco distinto dal principale, che contiene immagini, diagrammi, grafici, esempi di codice.

È possibile associare una didascalia tramite <figcaption>.

```
<figure>  
<figcaption>Figura 2.1</figcaption>  
  
</figure>
```


Tag <aside>

Rappresenta una sezione che include un contenuto legato a quanto trattato nella pagina, ma in qualche modo distinto.

```
<aside>  
  <h3>Fonti</h3>  
  <p>W3C</p>  
</aside>
```

Tag <time>

Indica la data di creazione di un particolare elemento (<article>, documento, <section>, etc)

```
<time datetime="2009-10-22T13:59:47-04:00"  
pubdate>
```

October 22, 2009 at 1:59pm EDT

```
</time>
```

Con l'attributo **pubdate** indica che è la data di pubblicazione

Formattazione Testo

Elemento	Descrizione
<bdi>	Porzione di testo che deve essere isolata
<mark>	Porzione di testo evidenziato
<ruby>	Porzione di testo con ideogrammi
<wbr>	Interruzione di linea solo se lo richiede il layout

Form

Utilizzate per fornire la possibilità all'utente di inviare contenuti al server.

Il server prende in input i dati, li elabora e formula una risposta all'utente.

Il tag **<form>** racchiude il blocco di elementi che gestiscono l'input dell'utente

Struttura degli input

```
<form>
```

```
<label for="name-tbx">name</label>
```

```
<input type="text" name="name" id="name-tbx" />
```

```
<label for="email-tbx">email</label>
```

```
<input type="email" name="email" id="email-tbx" />
```

```
<input type="submit" value="Send!" />
```

```
</form>
```

Data type per gli input

`<input type="email" />`

`<input type="date" min="2010-01-01" max="2011-01-01" />`

`<input type="range" min="0" max="100" />`

`<input type="search" results="10" placeholder="Cerca..." />`

`<input type="tel" pattern="regex" />` (<http://html5pattern.com/Phones>)

`<input type="color" placeholder="es: #333333" />`

`<input type="number" step="1" min="-5" max="10" />`

Controllo di una form

Attributo **required**, che consente la validazione

```
<style>  
    [required] { background-color: red; }  
    :invalid { background-color: red; }  
</style>  
  
<input type=text required />
```

* non supportati da IE 9, supportate da IE 10

E poi...

Meter



```
<meter min="0" low="10" optimum="50" high="90" max="100"></meter>
```

Rappresenta quanto siamo vicini ad un valore massimo stabilito

Progress



```
<progress max="100" value="80"></progress>
```

Rappresenta a che punto siamo di un determinato processo visto come un ideale 100%

Entrambi producono la visualizzazione di barre di avanzamento, la cui colorazione dipende dal browser e dal OS

CSS (Cascading style sheets)

- Utilizzati per stilizzare delle pagine web
- Passano in stato «Recommended» del W3C nel 1996
- Si inseriscono all'interno del tag <head> di una pagina HTML e collegano un file esterno

```
<html>  
    <head>  
        <link rel="stylesheet" type="text/css" href="style.css" >  
    </head>  
</html>
```

- Questa soluzione garantisce il massimo riutilizzo di codice possibile potendo essere richiamato in ciascuna delle pagine web HTML desiderate.



Regole CSS

E' costituito da una sequenza di *regole*, costituite da un *selettore* e da un *blocco di dichiarazioni*, racchiuso tra parentesi graffe

```
selettore {  
    proprietà1:valore1;  
    proprietà2:valore2;  
}
```

Le proprietà CSS sono numerose, le più utilizzare sono circa 60, tra cui:

background-color: definisce il colore di sfondo di un elemento;

color: definisce il colore del testo di un elemento;

font-size: definisce la dimensione di un testo;

float: definisce un blocco *flottante*, spostabile su uno dei lati sx/dx del suo contenitore

margin e *padding*: definiscono lo spazio circostante agli elementi

text-align: definisce l'allineamento degli elementi, tra cui il testo.


Selettori

Stile per i tag

```
body {  
  font-family: arial;  
  font-size: 15px;  
}  
p, div {  
  color: ■ red;  
  background: ■ white;  
}  
* {  
  color: ■ green;  
}
```

* Applica lo stile specificato a tutti i tag
Viene chiamato selettore universale

Selettori di attributi

```
input[type="button"] {  
  color:  red;  
  background:  black;  
}  
input[type="checkbox"], input[type="radio"] {  
  color:  black;  
  background: transparent;  
}  
input[disabled], input[readonly] {  
  background:  gray;  
}
```

Consentono di selezionare un elemento da un suo attributo.

Es: attributo delle form

Selettore di attributo class o id

```
/*Selettore tramite id*/
#par {
    color: cornflowerblue;
}

/*Selettore tramite CLASS name*/
.paragrafo {
    font-weight: bold;
}
```

```
<p id="par"> ciao </p>
```

```
<p class="paragrafo"> ciao </p>
```

Classe di stile

definisce uno stile richiamabile da ogni tag con l'attributo class.

Permette di definire uno stile per tipologie di tag differenti.



Mescolare tag e classi

```
p, .warning{  
  font-weight: bold;  
}
```

Mescolando i selettori per tag e per classe, lo stile verrà applicato se il tag:

- è un p
- ha class warning
- è entrambi

Selettori Combinati


```
p strong {  
  color:  red;  
}  
div * p {  
  background:  yellow;  
}
```

Tag1 Tag2

TUTTI gli elementi strong in elementi p avranno colore red

Tag1 * Tag2

Esempio: tutti i tag p all'interno di un tag div (anche se innestati)

```
p > strong {  
  color:  green;  
}
```

Tag1 > Tag2

Esempio: tutti i tag strong 'figli' di p (non innestati)

Selettori Combinati

```
tag1 + tag2 {
```

```
  proprietàZ: value;  
}
```

per selezionare tag2 che si trova DOPO un elemento tag1

```
tag1 ~ tag2 {
```

```
  proprietàZ: value;  
}
```

per selezionare OGNI elemento <tag2> preceduto da un elemento <tag1>

```
div + h3 {  
  text-align: right;  
  color: firebrick;  
}
```

/*seleziona il tag h3 DOPO un tag div di chiusura*/

```
h2 ~ h3 {  
  background-color: aqua  
}
```


/*seleziona tutti i tag h3 preceduti da un tag h2*/

Pseudo-Classi (:)

Le pseudo-classi sono usati per definire uno stato particolare di un tag.

Esempio: tag <a>

Possono essere anche combinate con classi CSS!


`a.classNameElemento:hover {
 color: yellowgreen;
}`

```
a:link {  
    color: darkblue;  
} /*Link non visitato*/  
  
a:visited {  
    color: mediumpurple;  
} /*Link visitato*/  
a:hover {  
    color: yellow;  
} /*Mouse sul link */  
  
a:active {  
    color: magenta;  
} /*Link selezionato (clic con  
il destro sopra per es.)*/
```

Pseudo-Elementi

Gli pseudo-elementi sono elementi per consentire una formattazione più semplice.

Pseudo Elemento	Descrizione
::first-name	Prima riga di un elemento
::first-letter	Prima lettera di un elemento
::before	Inserire del contenuto prima dell'elemento
::after	Inserire del contenuto dopo dell'elemento

Esempio

```
p::first-letter {  
  color: blueviolet;  
  font-size: xx-large;  
}
```

Disposizione degli elementi

La disposizione degli elementi viene gestita tramite tre proprietà:

- Float
- Visibility: visibilità dell'elemento (hidden per nascondere)
- Display: gestisce rendering dell'elemento (inline, block, inline-block, none)

Proprietà Display

Ricorda:

- In-line: elementi in linea (img, span, a)
- Block: contenitori che delimitano spazio (div, section, p, footer, h1...h6, form...)

Mostrare un elemento in-line come block:

```
span {  
  display: block;  
}
```

Mostrare un elemento block come in-line:

```
a {  
  display: in-line;  
}
```

display: none; → elemento nascosto

Proprietà Float e Clear

- Float : per ridisporre l'elemento con altri
- Clear: “pulire” lo spazio attorno all'elemento

```
tag {  
  float: left; /*right, none*/  
}
```

```
tag {  
  clear: left; /*right, none, both*/  
}
```

Posizionamento Assoluto

Elementi potrebbero essere sovrapposti.

Le istruzioni sono:

- Position: tipo di posizionamento (static, absolute)

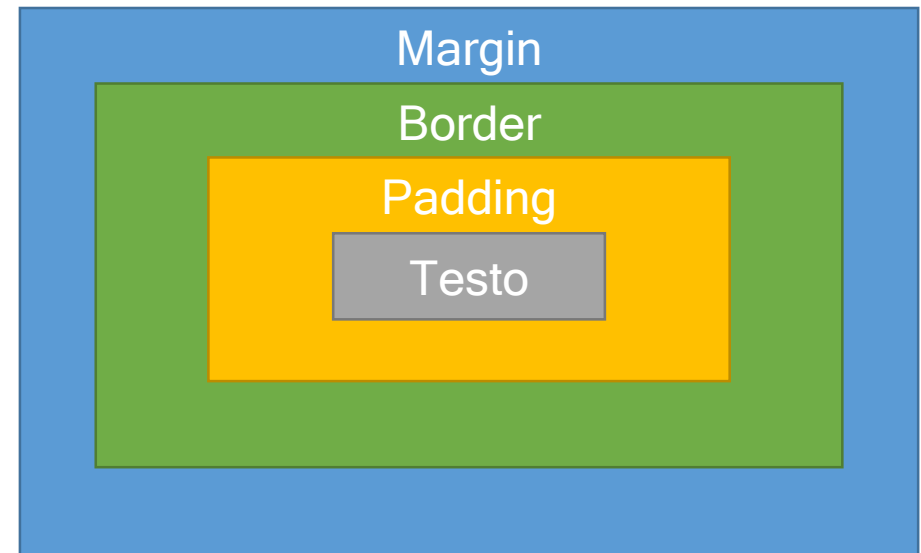
```
tag{  
  position: relative;  
  top: -20px;  
}
```

- Z-Index: gestione degli elementi in primo piano, secondo piano.
- Top, Left...: specifica l'allineamento

Box Model

Ogni elemento ha una dimensione data dalla somma dei seguenti parametri:

- **Border:** bordo dell'elemento
- **Padding:** spazio tra contenuto dell'elemento e il bordo
- **Margin:** spazio tra il bordo dell'elemento e quelli circostanti



Gestione del Border

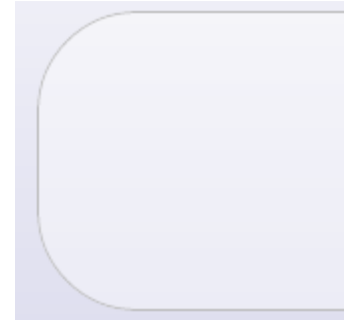
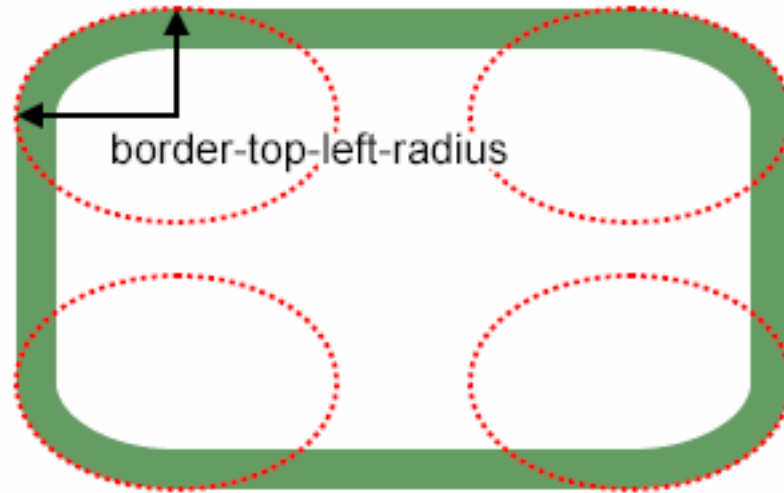
È possibile impostare le seguenti caratteristiche:

Proprietà	Descrizione
Border-width	Dimensione del bordo
Border-Style	Stile del bordo (dotted line, solid line)
Border-Color	Colore del bordo
Border Radius	Curvatura degli angoli

Bordi arrotondati

Attraverso l'attributo border-radius

```
.border{  
  border-radius:50px;  
}
```



Media Query

Utilizzate per cambiare lo stile e adattare il contenuto in base al dispositivo che lo utilizza.

L'idea è creare un CSS per ogni tipo di visualizzazione che si intende supportare creando delle regole in base alle quali sarà il browser a decider quale CSS utilizzare.

Tali regole includono la definizione del tipo di device e delle sue caratteristiche.

Gestione Tipografica

Alcune delle principali impostazioni per la formattazione sono le seguenti:

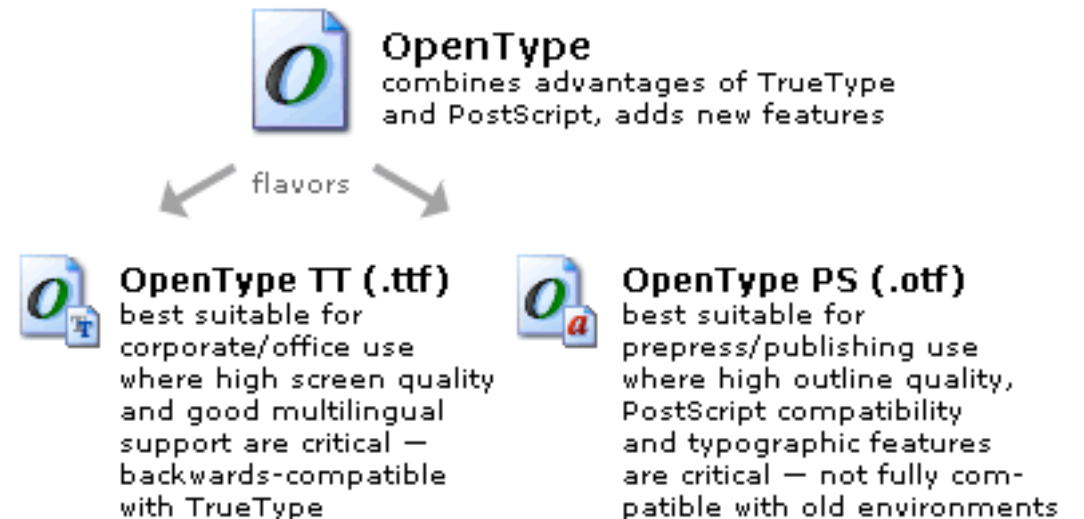
Proprietà	Descrizione
Font-family	Tipo di carattere (font)
Font-size	Dimensione del carattere
Color	Colore del testo
Font-weight	Grassetto
Font-style	Stile del font (italic)
Text-transform	Trasformazione testo (capitalize, uppercase...)
Text-decoration	Sottolineatura e Barrato
Text-Align	Allineamento testo

Supporto tipografico

Font in formato OTF (Open Type Font, evoluzione di True Type Font)

[<http://en.wikipedia.org/wiki/OpenType>]

```
@font-face {  
    font-family: 'MyFont';  
    src: url(MyFont.otf);  
}
```



Suddivisione testo in colonne

È possibile definire il testo in più colonne tramite alcune proprietà di stile del contenitore testuale quali:

- Column-count: conteggio delle colonne
- Column-gap: spazio tra colonne
- Column-rule: stile per la divisione in colonne

Troncamento di testo

Già supportato da IE, diventa parte delle specifiche:

```
.text{  
    text-overflow:ellipsis;  
}
```

Without overflow: Looooooooooooooooooooooooooooo text
.overflow-box: Looooooooooooooooooooo...

© 2019 iCubed Srl

La diffusione di questo materiale per scopi differenti da quelli per cui se ne è venuti in possesso è vietata.

iCubed s.r.l. • Piazza Durante, 8 - 20131, Milano
• Phone: +39 02 57501057 • P.IVA 07284390965

