




La tecnologia per le applicazioni Web

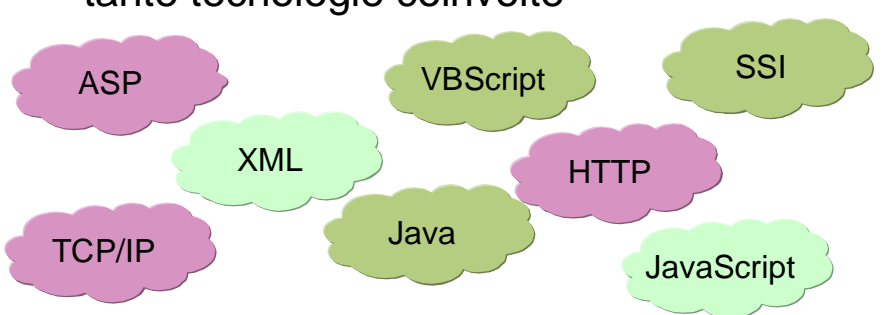
Laura Farinetti
Dip. Automatica e Informatica
Politecnico di Torino
farinetti@polito.it

L. Farinetti - Politecnico di Torino



Obiettivo

- Capire il **funzionamento** delle applicazioni Web
- Avere un **quadro di riferimento** delle tante tecnologie coinvolte



L. Farinetti - Politecnico di Torino



Strumenti per sviluppare applicazioni Web

- **Tutte** le applicazioni Web fanno all'incirca le stesse cose:
 - forniscono un'interfaccia per la richiesta di informazioni (query interface); es: ricerca in un database, richiesta di un file, acquisto di un libro, ...
 - trasmettono al Web server i dati dell'utente
 - eseguono l'elaborazione dei dati sul lato server (server-side processing)

L. Farinetti - Politecnico di Torino

3

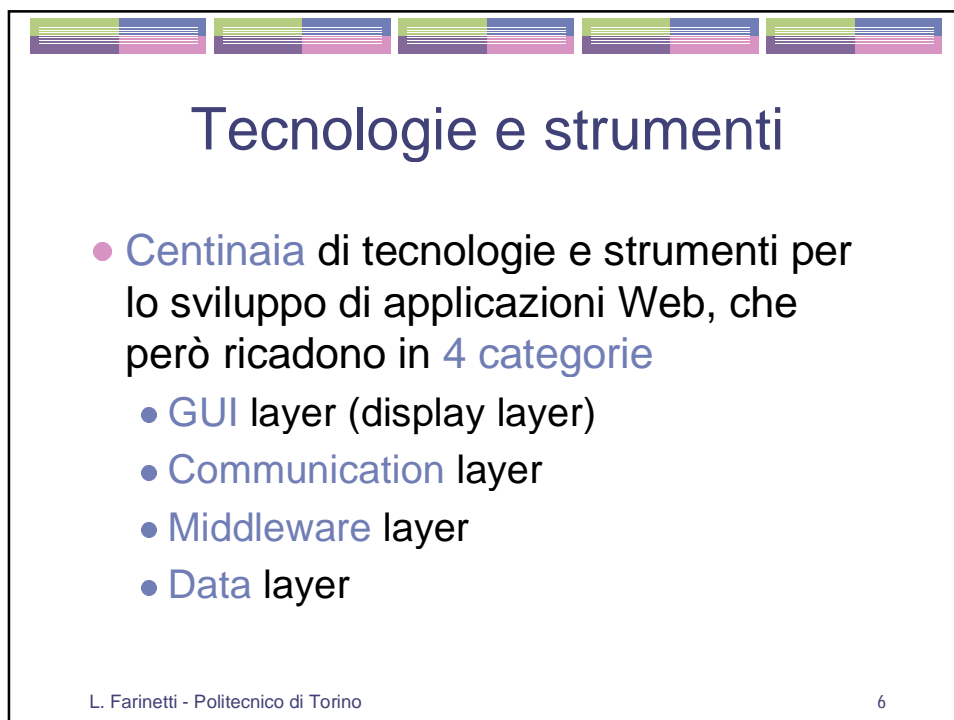
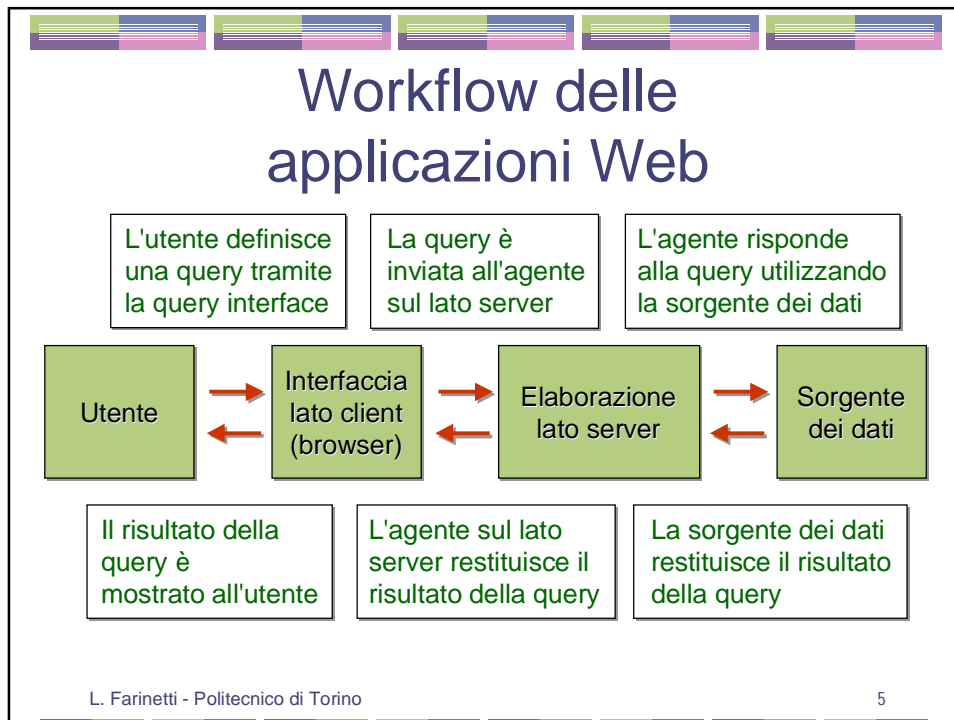


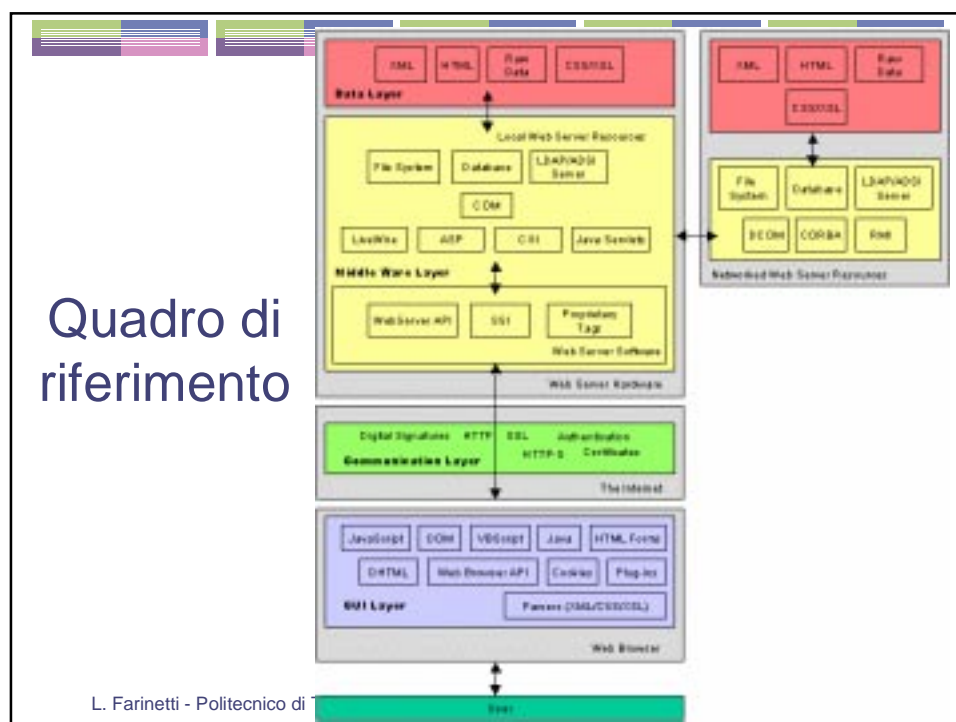
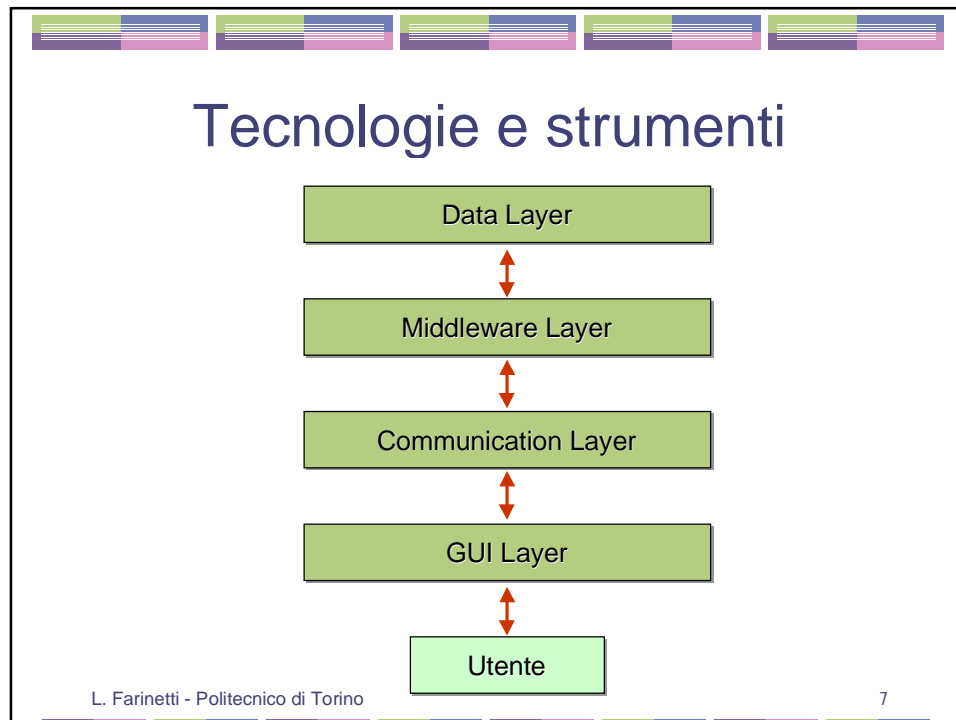
Strumenti per sviluppare applicazioni Web

- **Tutte** le applicazioni Web fanno all'incirca le stesse cose:
 - trasmettono al client i risultati delle query
 - eseguono l'elaborazione dei dati sul lato client (client-side processing)

L. Farinetti - Politecnico di Torino

4





GUI (display) layer

- Detto anche "font-end"
- **Interfaccia** fra l'utente e il calcolatore
 - **traduttore** fra i due linguaggi
- **Compito**
 - tradurre intenzioni, desideri, necessità, richieste dell'utente in **istruzioni comprensibili dal calcolatore**
 - tradurre i dati generati dal calcolatore in **informazioni comprensibili all'utente**

L. Farinetti - Politecnico di Torino

9

GUI (display) layer

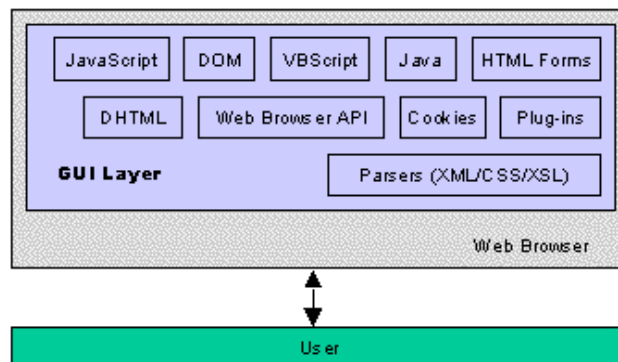
- Creazione e modifica di "**UI widgets**", o "controlli"
 - strumenti per la **creazione di interfacce** che permettano agli utenti di inserire richieste, informazioni, ...
 - possono contenere **bottoni**, menu, finestre di dialogo, immagini "cliccabili",...
- Parole chiave: **intuitività** e **facilità d'uso**
 - fondamentali per la **qualità** percepita dall'utente

L. Farinetti - Politecnico di Torino

10

Tecnologie per il GUI layer

- Quelle che consentono di **creare interfacce** basate su “widgets”



L. Farinetti - Politecnico di Torino

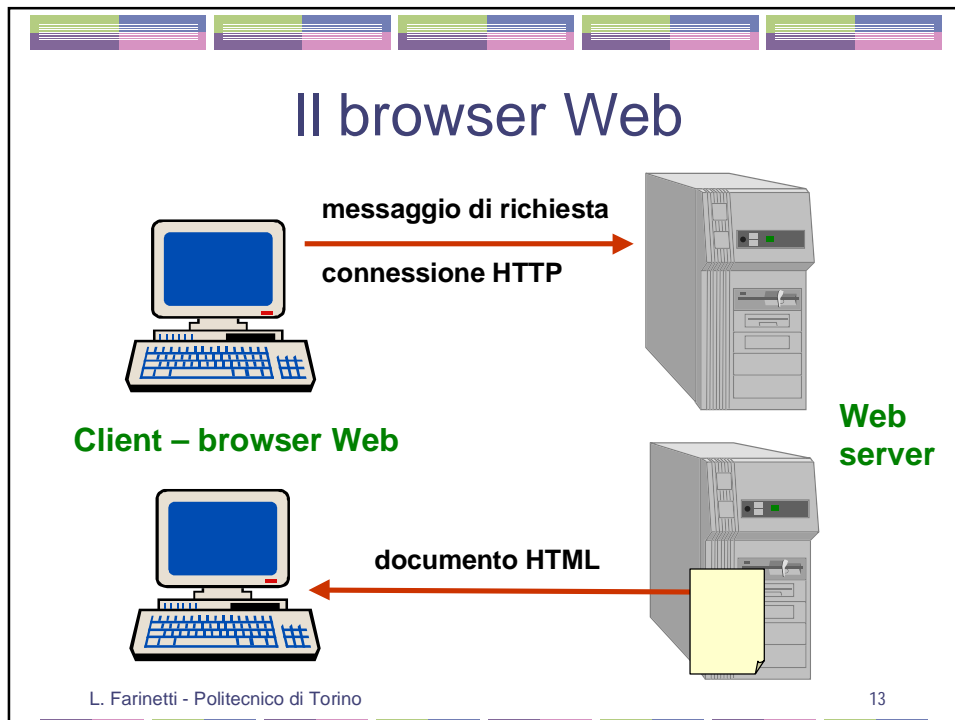
11

Il browser Web

- È un **programma** che sa
 - come **contattare** il **Web server** (usando il **protocollo HTTP**)
 - come **richiedere** un documento al Web server
 - come **visualizzare** per l'utente il documento restituito dal server
- Il documento restituito dal Web server è **“formattato”** (linguaggio HTML)

L. Farinetti - Politecnico di Torino

12



Tecnologie per il GUI layer

- Le **form HTML** sono un esempio di “widget”
 - consentono di definire **formati standard di input** (test box, bottoni, menu, quiz, ...)
- **Cookies**
 - mantengono lo **stato dell'applicazione**
 - HTTP è un protocollo “**stateless**” (che non conserva lo stato)
- Questo è quanto può fare il Web browser
 - per andare oltre: **plug-in**

L. Farinetti - Politecnico di Torino

14

Plug-in

- Ai browser Web sono richieste sempre nuove caratteristiche
 - supporto alla visualizzazione di audio, video, animazioni, interattività, ...
- Browser più potenti? Meglio di no
 - meglio “leggeri” perché siano veloci
 - necessario un aggiornamento continuo
- Soluzione: plug-in
 - aggiunta di nuove componenti, di dimensioni ridotte, solo quando richiesto

L. Farinetti - Politecnico di Torino

15

Java

- Linguaggio di programmazione sviluppato dalla Sun Microsystems nel 1991
 - nasce come plug-in, ma è ben presto integrato nei browser
 - in realtà Sun intendeva creare un linguaggio di programmazione indipendente dai sistemi operativi, orientato anche all'elettronica di consumo (TV, VCR, elettrodomestici, ...)
- Java Virtual Machine: interpreta il codice Java scaricato dal Web server

L. Farinetti - Politecnico di Torino

16

Java

- In Java si possono scrivere **due tipi** di programmi
 - programmi classici, object-oriented
 - **applet Java**: programmi appositi per Web
- **Libreria di “widget”** estremamente completa

L. Farinetti - Politecnico di Torino

17

JavaScript

- **Linguaggio di script** sviluppato da Netscape
- Il codice JavaScript è **parte integrante del documento HTML**
 - è **interpretato dal browser Web (client)**: non richiede programmazione sul lato server
 - permette al client di intraprendere azioni direttamente **senza** passare dalla rete
 - consente la creazione di **“widgets”**

L. Farinetti - Politecnico di Torino

18

VBScript

- Risposta di Microsoft a JavaScript
 - inizialmente Microsoft aveva sviluppato Jscript, poi in gran parte abbandonato in favore di VBScript
- Linguaggio di script basato su Visual Basic

L. Farinetti - Politecnico di Torino

19

DHTML

- Dynamic HTML, consente di combinare l'HTML con i fogli stile (CSS, Cascading Style Sheets) e con i linguaggi di script
- I fogli di stile (Style Sheet) consentono di controllare la visualizzazione di un documento Web senza comprometterne la struttura
 - font, colori, margini, ...

L. Farinetti - Politecnico di Torino

20

DHTML

- Utilizza il **Document Object Model (DOM)**, che descrive l' "anatomia" di una pagina Web
- Il DOM è utilizzato per creare dei "widget" virtuali a partire dalle componenti di una pagina Web
 - **accessibilità di qualunque porzione** di una pagina Web (paragrafo, tabella, cella di una tabella, singolo carattere...)

L. Farinetti - Politecnico di Torino

21

DHTML

- Le **componenti di una pagina** diventano "widget"
 - a qualunque componente può essere applicato uno **stile**
 - a qualunque componente può essere associato uno **script in JavaScript o VBScript** per modificarne gli attributi o definirne un comportamento
- Tecnologia nuova, **non ancora standard** per i browser ma **molto promettente**

L. Farinetti - Politecnico di Torino

22



GUI layer - riepilogo

- Molte tecnologie, ma il loro scopo è comunque sempre **creare e manipolare** i "widget" sul lato Web browser (client)
- A questo punto la maggior parte delle applicazioni Web deve **spedire i dati** inseriti dall'utente nella GUI al Web server per un'ulteriore elaborazione
- **Accesso al Web server**: tecnologie del communication layer

L. Farinetti - Politecnico di Torino

23

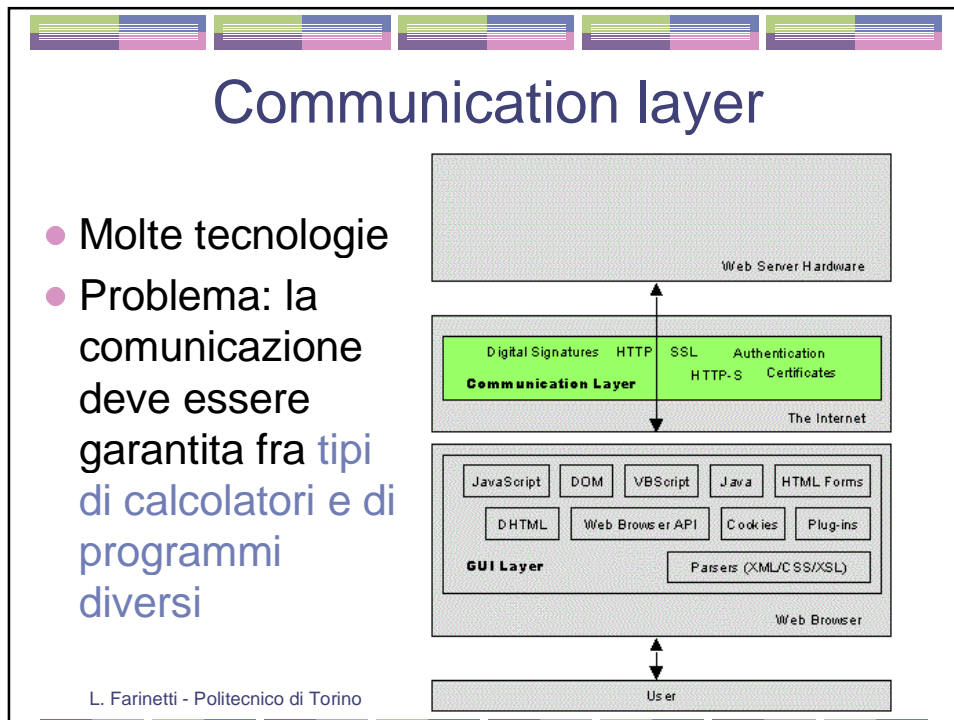


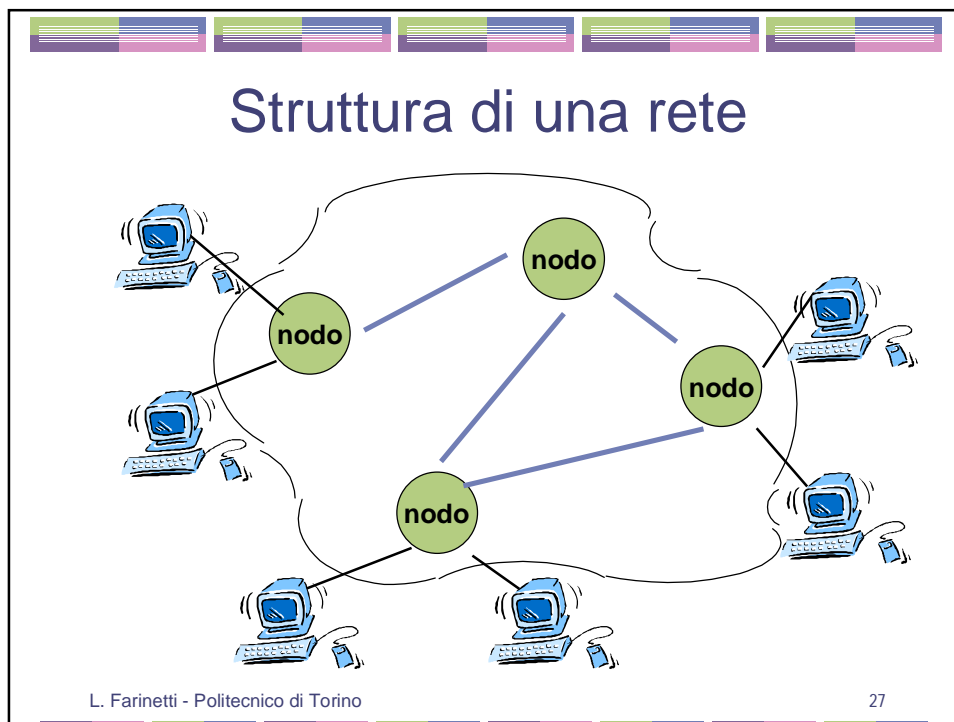
Communication layer

- Compito: **trasferire i dati** inseriti dall'utente dal browser Web al Web server
- L'informazione deve essere trasferita **interamente, correttamente, e nella sequenza** in cui è stata inserita
- In più l'informazione può (o deve) essere trasferita **in modo sicuro**
 - nessuno deve poterla leggere, modificare, o spacciarsi come autore

L. Farinetti - Politecnico di Torino

24





- ### I protocolli
- **Protocollo HTTP** (Hypertext Protocol)
 - protocollo utilizzato dai **browser** e dai **server Web** per scambiarsi informazioni
 - protocollo privo di stato (stateless), di tipo “richiesta/risposta” che gestisce il **trasporto** ed il **mantenimento dell'integrità** dei documenti multimediali
 - **Protocollo HTTPS** (HTTP sicuro)
 - versione di HTTP che utilizza la **comunicazione criptata**
- L. Farinetti - Politecnico di Torino
- 28

La crittografia

- È il metodo più diffuso per proteggere i dati
- I dati vengono alterati tramite una formula matematica prima di essere trasmessi, in modo da essere illeggibili
- I dati originali possono essere recuperati applicando la formula matematica inversa
 - es: HAL → IBM (2001: Odissea nello spazio)

L. Farinetti - Politecnico di Torino

29

La crittografia

- Due tipi di crittografia
- Crittografia a chiave segreta
 - mittente e ricevente hanno la stessa chiave per codificare/decodificare i dati
 - problema: trasmettere la chiave segreta
- Crittografia a chiave pubblica e privata
 - due chiavi, una privata e una pubblica

L. Farinetti - Politecnico di Torino

30

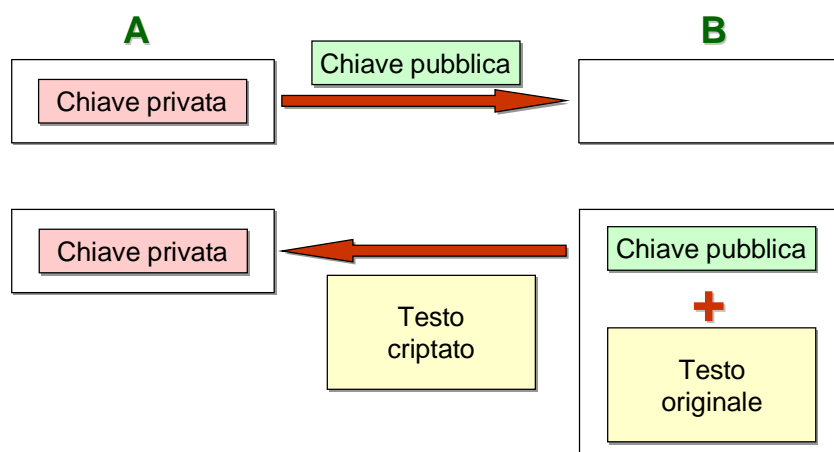
Crittografia a chiave pubblica e privata

- Coppia di chiavi, una “**privata**” (mai trasmessa) ed una “**pubblica**” (trasmessa a tutti)
 - i testi codificati con la chiave pubblica possono essere decodificati con la chiave privata: **messaggi protetti**
 - i testi codificati con la chiave privata possono essere decodificati con la chiave pubblica: **certificazione del mittente, firma digitale**

L. Farinetti - Politecnico di Torino

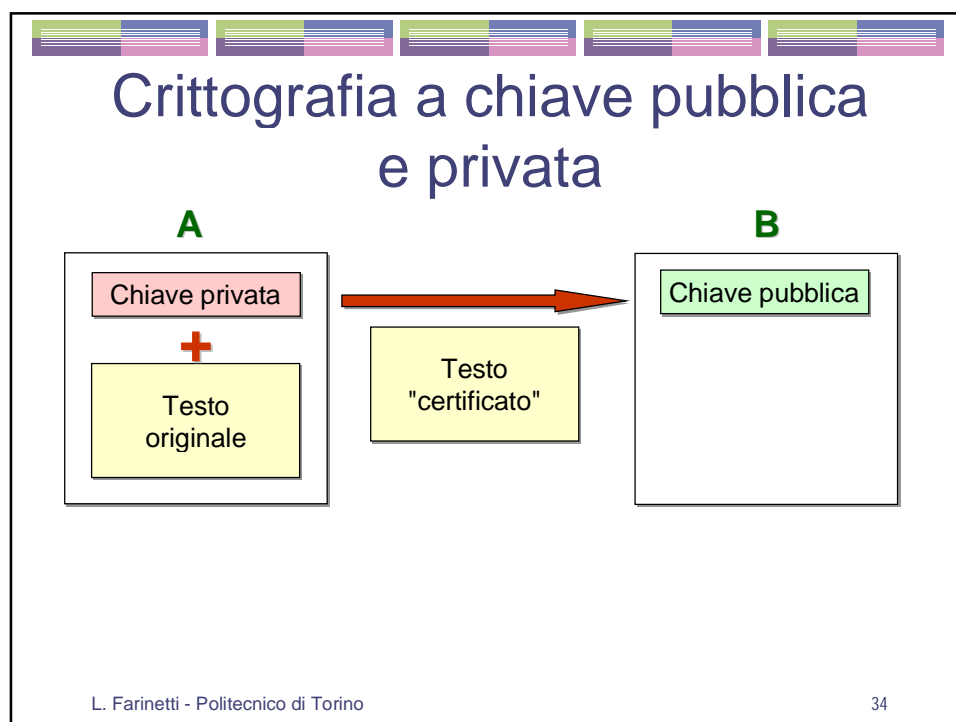
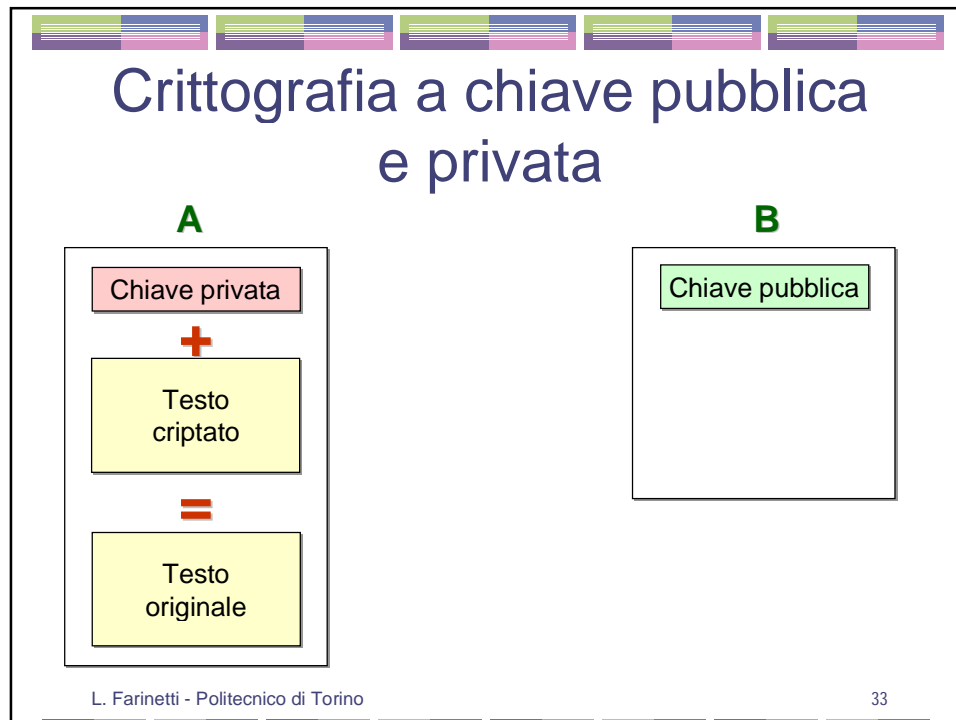
31

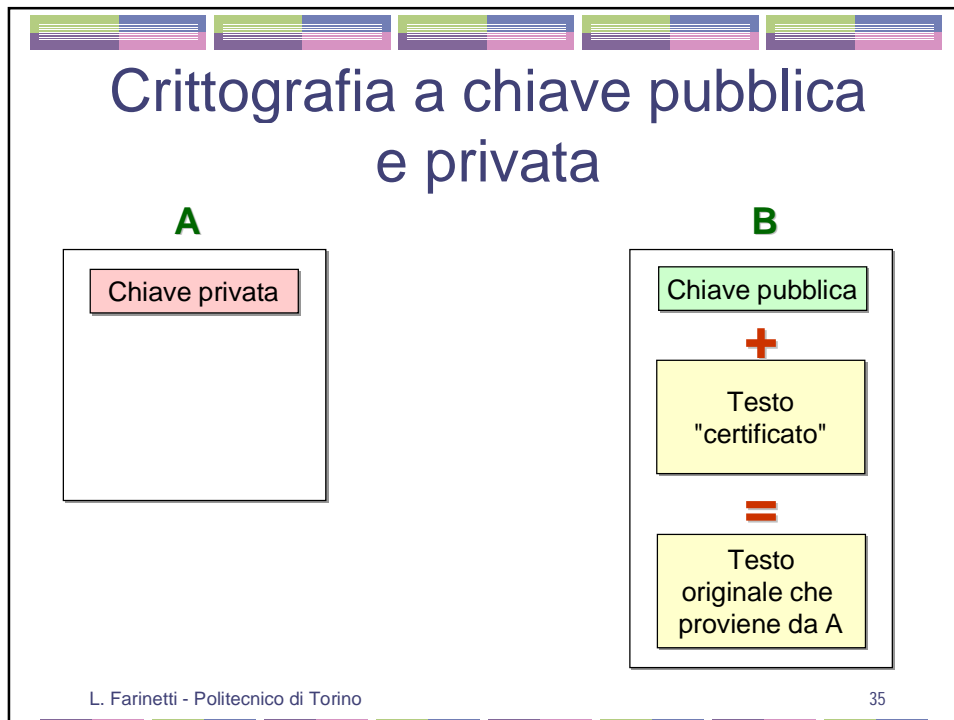
Crittografia a chiave pubblica e privata

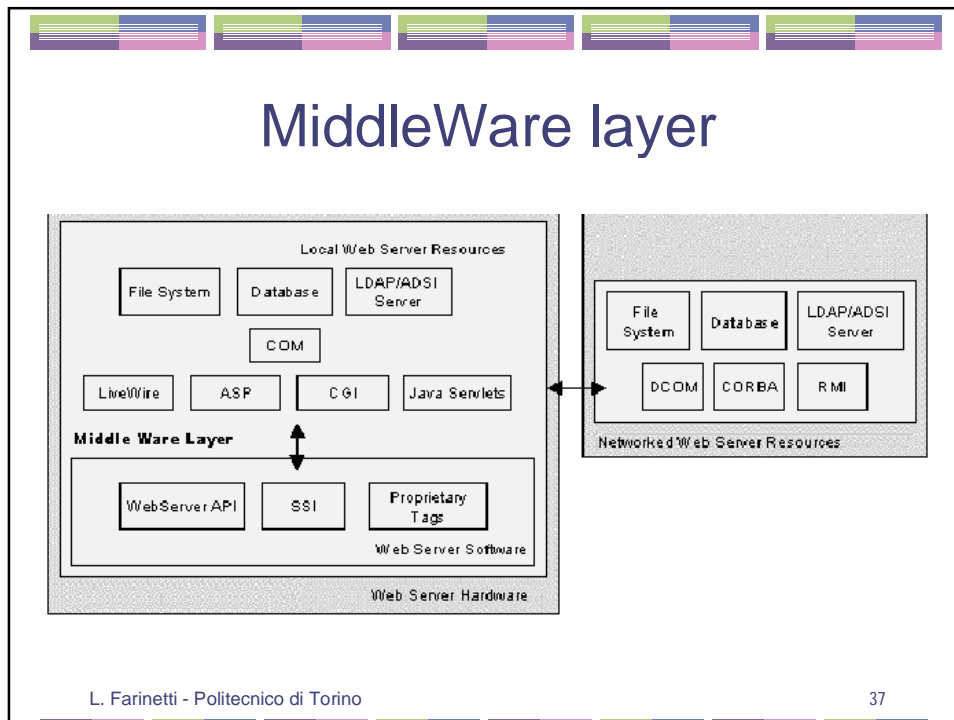


L. Farinetti - Politecnico di Torino

32







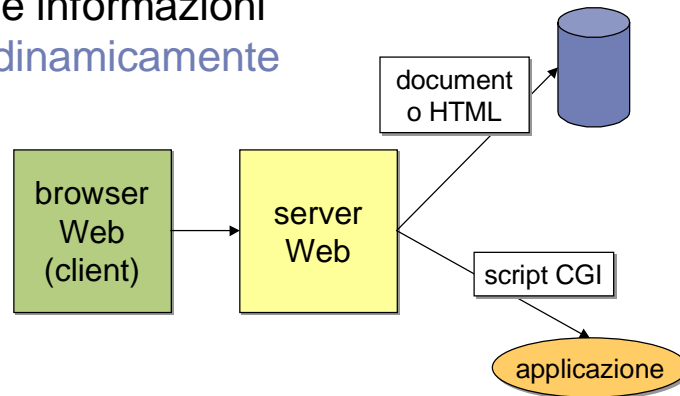
CGI

- **Common Gateway Interface:** strumento di base per accedere alle risorse del Web server
- Servizio fornito da **tutti i Web server** che consente di
 - creare uno **script eseguibile** che il Web server può richiamare su richiesta
 - **trasferire dati**
 - **filtrare le risposte** generate dallo script e **trasmetterle** al browser

L. Farinetti - Politecnico di Torino 38

HTML vs. CGI

- Differenza chiave: è possibile generare le informazioni **dinamicamente**



L. Farinetti - Politecnico di Torino

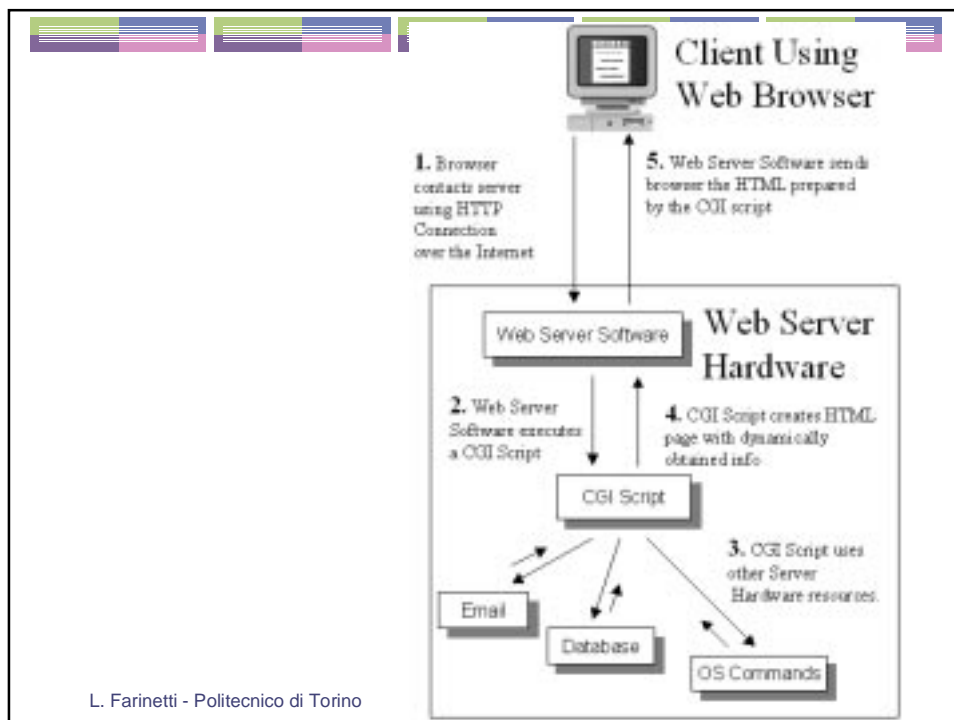
39

CGI

- CGI non è un linguaggio di programmazione, ma un modo standard di accedere alle risorse
 - definisce un modo di far dialogare il Web server con le risorse
- Le applicazioni CGI possono essere scritte in un qualunque linguaggio di programmazione: C, C++, Perl, Python, ...

L. Farinetti - Politecnico di Torino

40



CGI

- Problema degli script CGI: la **velocità di esecuzione**
 - ogni volta che il Web server riceve una richiesta deve **eseguire** lo script CGI
- Soluzione: **estendere le funzionalità del Web server**, aggiungendo capacità logiche e di elaborazione

SSI

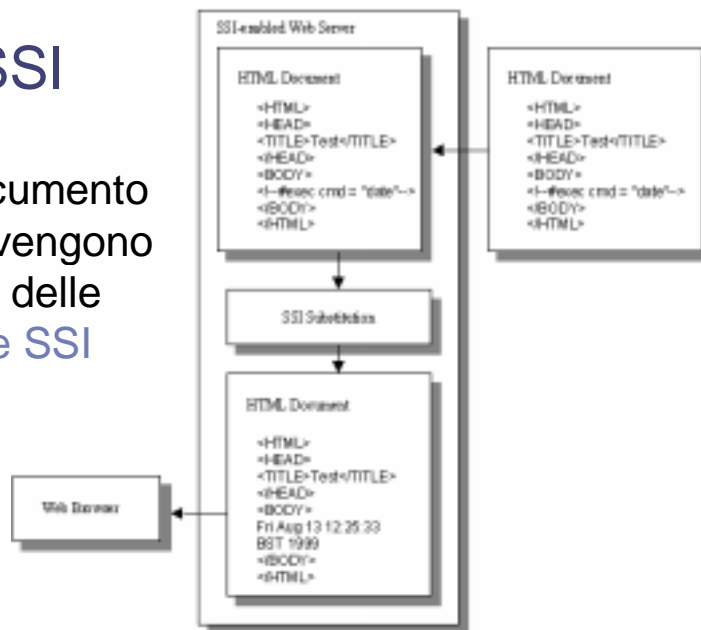
- **Server Side Includes**
 - la prima delle tecnologie sviluppate in questa direzione
- **Tag particolari** sono inseriti nel documento HTML
- Il server Web è in grado di **interpretare questi tag**, e di **tradurli “al volo”** mentre il documento HTML viene trasferito al browser

L. Farinetti - Politecnico di Torino

43

SSI

- Nel documento HTML vengono inserite delle **direttive SSI**



L. Farinetti - Politecnico di Torino

44



SSI con insieme di tag proprietari

- Partendo dall'idea di SSI alcune aziende hanno sviluppato dei Web server dotati di moltissime funzionalità per rendere le pagine dinamiche
 - es: Cold Fusion (la più conosciuta, oltre 70 tag custom definiti), Net Objects

L. Farinetti - Politecnico di Torino

45



ASP: Active Server Pages

- La tecnologia SSI è limitata all'insieme di comandi/tag offerti da chi ha implementato il Web server "SSI-enabled"
- Come inserire in una pagina HTML del codice che su richiesta venga interpretato dinamicamente dal Web Server?
- Per questo è nata la tecnologia ASP
 - una specie di ibrido CGI-SSI

L. Farinetti - Politecnico di Torino

46



ASP: Active Server Pages

- ASP è un'estensione dell' IIS Web server rilasciata da Microsoft
 - anche il Web server Apache oggi ha un'estensione che permette di utilizzare la tecnologia ASP
- Consente di definire tag di tipo custom in JavaScript o VBScript
- Tali tag vengono interpretati da IIS prima che le pagine vengano trasmesse al browser per la visualizzazione

L. Farinetti - Politecnico di Torino

47



ASP: Active Server Pages

- Un pagina ASP è semplicemente un file di testo con estensione .asp che contiene codice HTML e script
 - gli script sono interpretati dinamicamente
- Come tutte le tecnologie del lato server (server-side) le funzionalità sono compatibili con qualunque browser
 - l'elaborazione è fatta dal server, e il risultato è visualizzato come semplice HTML

L. Farinetti - Politecnico di Torino

48

Java server-side

- Alternativa alle tecnologie CGI o SSI/ASP per creare **servizi sul Web server** in grado di ricevere gli input delle applicazioni Web, di elaborarli e di restituire i dati elaborati
- Due tecnologie legate a Java
 - **Java Server Pages (JSP)**: simile ad ASP
 - **Java Servlet**: simile a CGI, ma scritti in Java

L. Farinetti - Politecnico di Torino

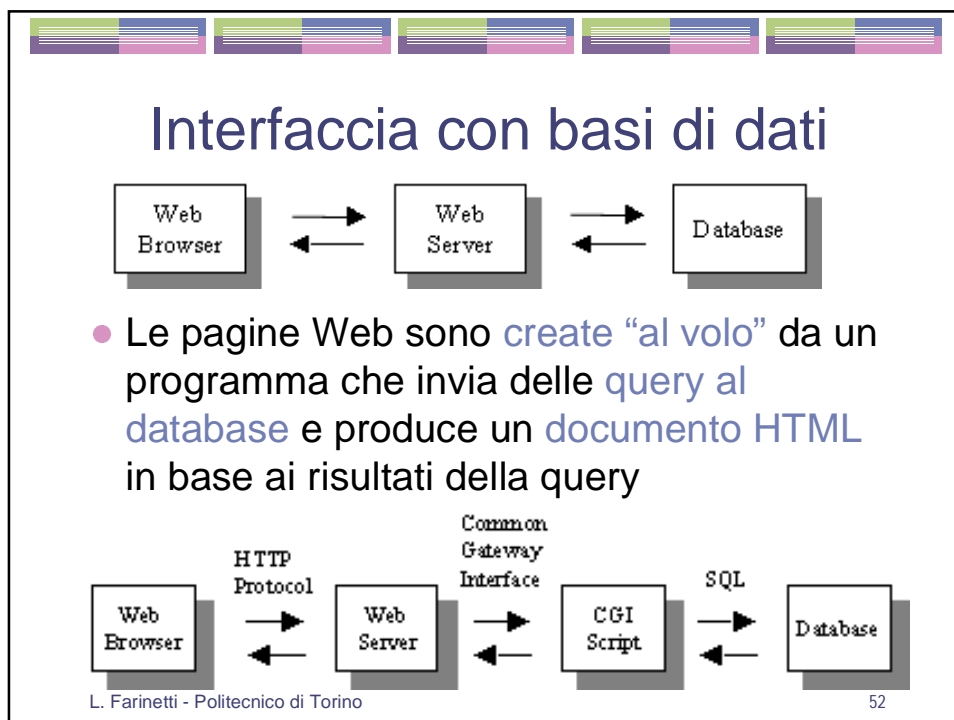
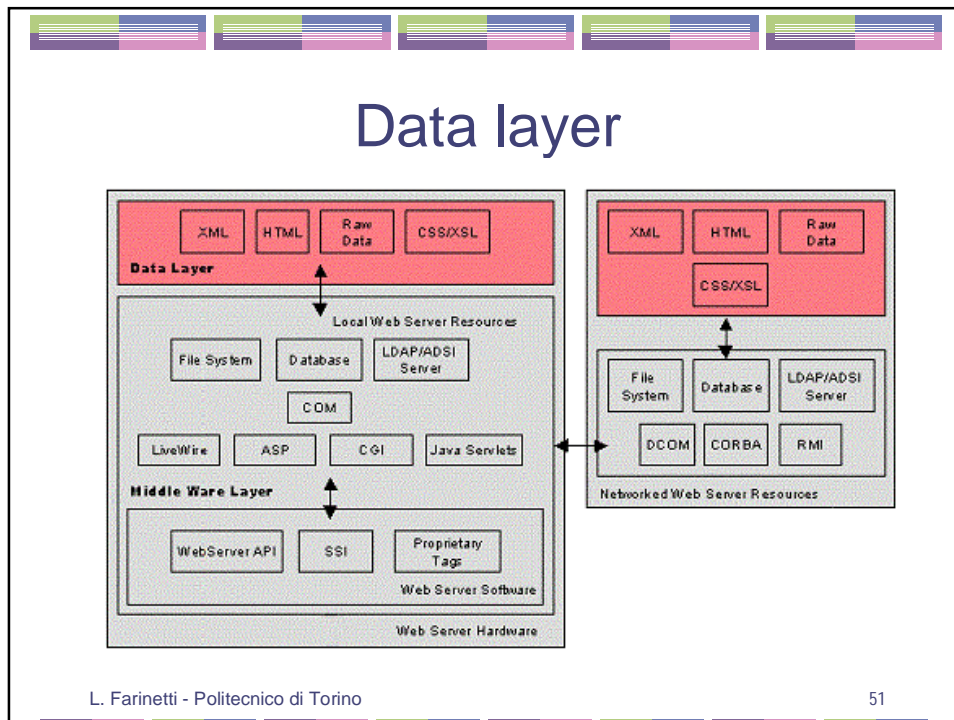
49

Data layer

- I **dati** sono un **elemento fondamentale** delle applicazioni Web
- Caratteristiche dei dati
 - hanno un **valore**, un **tipo**, una **descrizione**, un **formato**
- Le tecnologie del Data Layer hanno lo scopo di **definire, descrivere o standardizzare** una o più di queste caratteristiche

L. Farinetti - Politecnico di Torino

50



XML

- XML: eXtensible Markup Language
- Linguaggio estensibile realizzato per elaborare i documenti **strutturati**
- Pensato per il **Web** e per superare i limiti dell' HTML
- HTML è un **linguaggio predefinito**
- XML è un **metalinguaggio**
 - non ha tag predefiniti
 - consente di definire nuovi linguaggi

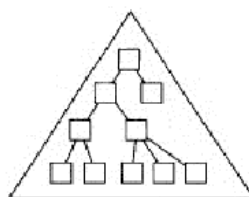
L. Farinetti - Politecnico di Torino

53

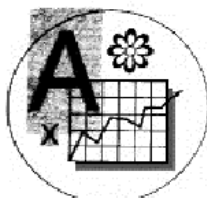
Le componenti di XML

"La Discovery stava accelerando verso Giove lungo un'orbita complessa calcolata alcuni mesi prima dagli astronomi sulla Terra e controllata costantemente da Hal"
A.C. Clarke

Contenuto



Struttura



Stile

L. Farinetti - Politecnico di Torino

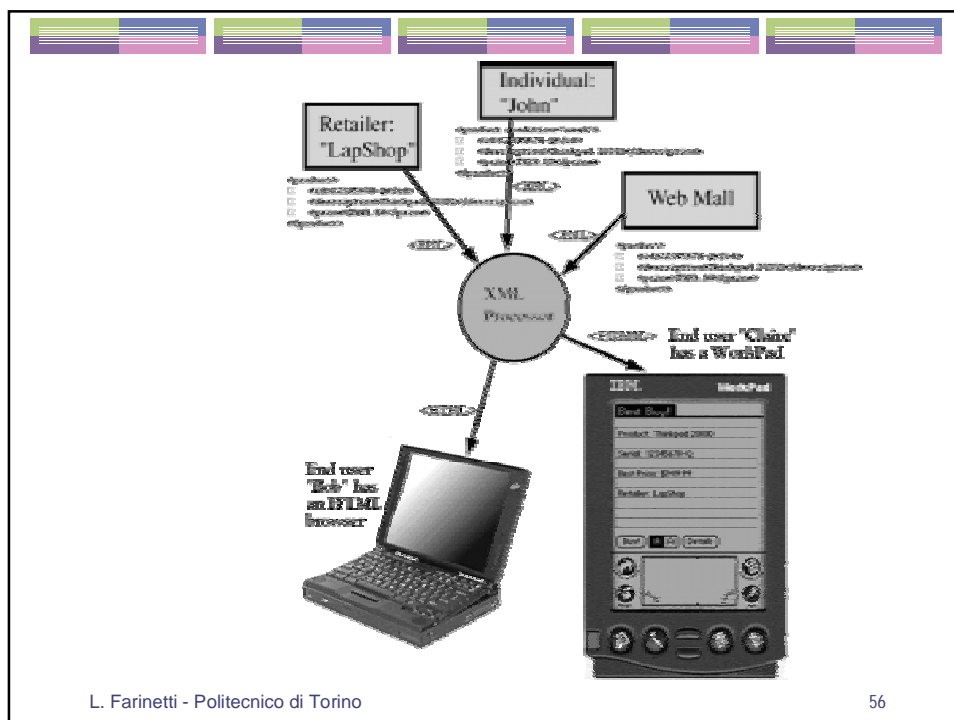
54

Le componenti di XML

- Tre parti fondamentali di un documento che dovrebbero essere gestite separatamente
 - Il **contenuto** (XML)
 - Le specifiche relative agli elementi: la **struttura** (DTD: Document Type Definition o XML Schema)
 - Le specifiche relative alla visualizzazione: lo **stile** (XSL)

L. Farinetti - Politecnico di Torino

55



L. Farinetti - Politecnico di Torino

56