Estensioni UML per il Web

Michele Zennaro
14-05-2004

Scaletta

- Le applicazioni web
- Scopo di un'estensione UML per il web
- Due punti di vista…
  - Uno più astratto
  - Uno più vicino ai file fisici
- Esempio conclusivo
- Commenti

Applicazioni web

- Che cos'è un'applicazione web?
- È un sito internet dove il contenuto delle pagine dipende dall'input ricevuto dall'utente
- È composta di più componenti diversi:
  - Pagine statiche (HTML, XHTML…)
  - Pagine dinamiche (JSP, ASP…)
  - Relazioni tra le pagine (link, submit…)
- Questo è quello che vorremmo modellare, cioè astrarre per capire meglio

Applicazioni web - 2

- Ma come rappresentare un'applicazione web in un modello astratto?
- Non con UML semplicemente …
- Bisogna estenderlo per poter rappresentare i nuovi pezzi

⇩

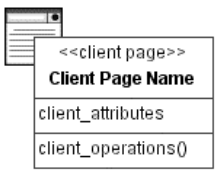
- WAE: Web Application Extension for UML

WAE: Web Application Extension for UML

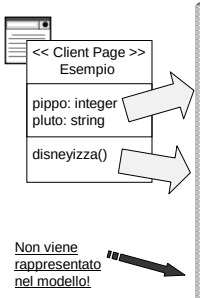
- Ci serviranno allora nuovi componenti per modellare:
  - Pagine client
  - Pagine server
  - Form
  - Le associazioni tra le pagine

« Client page »

- Una pagina client è una pagina web statica (tipo HTML), che può contenere script interpretati dai browser



« Client page »: esempio




```
<html>
<script
language="JavaScript">
var pippo
var pluto

function disneyizza(){
  //...
};
...
<body
onLoad="alert('Ciao')">
...
```

Non viene rappresentato nel modello!

7

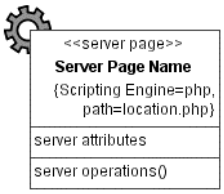
WAE: Web Application Extension for UML

- Ci serviranno allora nuovi componenti per modellare:
  - Pagine client
  - Pagine server
  - Form
  - Le associazioni tra le pagine

8

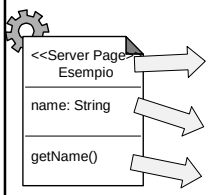
« Server page »

- Una pagina server rappresenta una pagina web che ha contenuti costruiti dal server a ogni richiesta



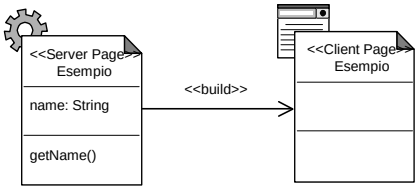
9

« Server page » : esempio




```
<% @page Language="Java" %>
<html>
<head><title>Esempio</title>
</head>
<body>
      <%! String name = new
           String("Bob"); %>
      <%! public String
           getName(){return
           name;} %>
      <p>Ciao <%=getName() %>
      </p>
</body></html>
```

<<Server Page>> e <<Client Page>>



11

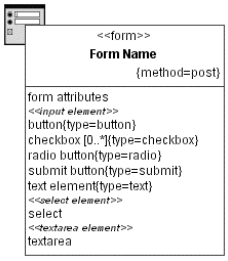
WAE: Web Application Extension for UML

- Ci serviranno allora nuovi componenti per modellare:
  - Pagine client
  - Pagine server
  - Form
  - Le associazioni tra le pagine

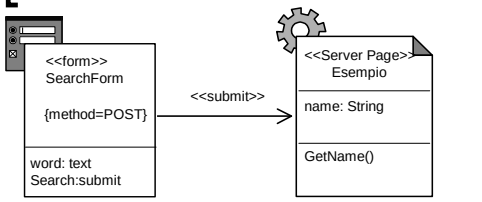
12

« Form»

- Una classe form corrisponde direttamente al tag <form> di HTML.
- Attributi = campi di input.
- Operazioni non ce ne sono



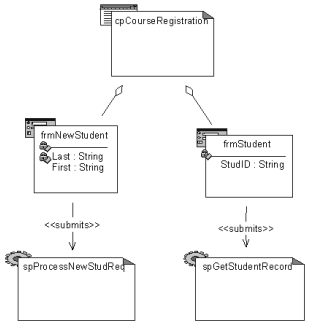
« Form HTML » : esempio




```
<form name="SearchForm" method="POST"
      action="ProcessSearch.asp">
  <input name="Word" type="text" size="30">
<input name="Search" type="submit" value="Search">
</form>
```


« Form HTML » - 2

- La classe stereotipata «form» va contenuta nella pagina client.



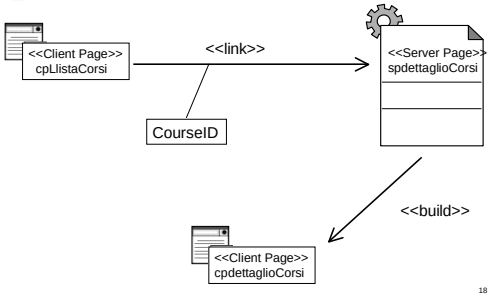
WAE: Web Application Extension for UML

- Ci serviranno allora nuovi componenti per modellare:
  - Pagine client
  - Pagine server
  - Form
  - Le associazioni tra le pagine

Stereotipi per le associazioni

Stereotipo	Descrizione
« link »	Relazione tra una pagina client e una risorsa lato server, o una pagina web.
« build »	Relazione fra una pagina server e una pagina client. Questa relazione identifica l'output HTML dell'esecuzione di una pagina server.
« submit »	Relazione fra un modulo «HTML form» e una pagina server. Quando la risorsa viene richiesta dal server, tutti i valori degli attributi associati ai campi del modulo sono inviati insieme alla richiesta per essere successivamente elaborati.

Stereotipi: esempio



Vista logica

- Tutto quello che abbiamo visto serve per definire un'astrazione del sito web…
- Ma in un web server la disposizione fisica dei file e quella logica può non corrispondere
- Come modellare allora questa differenza?

19

Vista dei componenti

- due viste diverse
  - Nella vista logica si astrae dai file fisici
  - Nella vista dei componenti
    - si legano le astrazioni definite nella vista logica ai file fisici del sistema eseguibile
    - si descrivono i moduli che costituiscono il sistema eseguibile

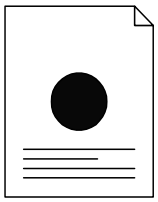
20

Vista dei componenti – 2

- Cosa ci servirà allora? Modelli per
  - File fisici statici (".html"…)
  - File fisici lato server (".jsp"…)
  - Physical root
  - Altri file che comporranno le pagine

21

« Static page »

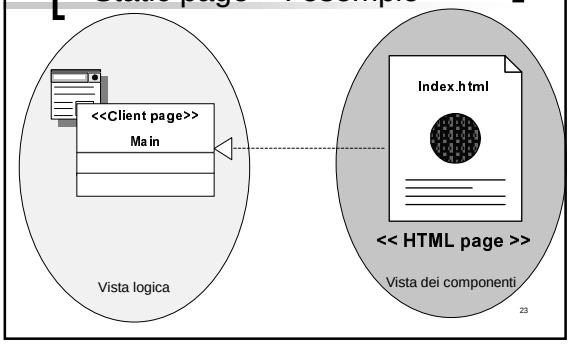


<<HTML Page>>

- Una risorsa che può essere richiesta direttamente da un browser client.
- È trasferita inalterata al client direttamente dal file system. (Non richiede elaborazione lato server)

22

« Static page » : esempio



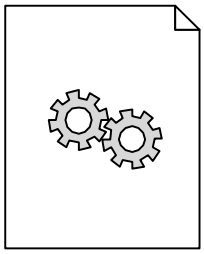
23

Vista dei componenti – 2

- Cosa ci servirà allora? Modelli per
  - File fisici statici (".html")
  - File fisici lato server (".jsp")
  - Physical root
  - Altri file che comporranno le pagine

24

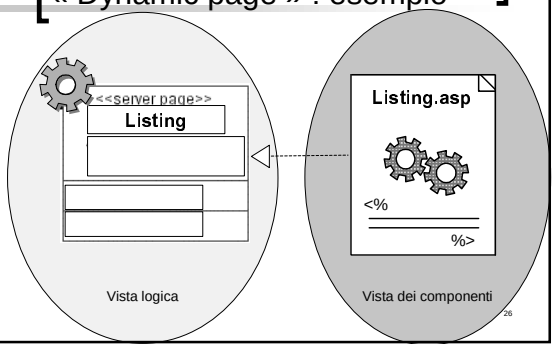
« Dynamic page »



- Una risorsa che può essere richiesta da un browser client
- Alla richiesta viene eseguita l'elaborazione lato server
- Possono elaborare l'input dell'utente inviato tramite form
- Dovrebbero venir sostituite dall'icona del linguaggio (se c'è)

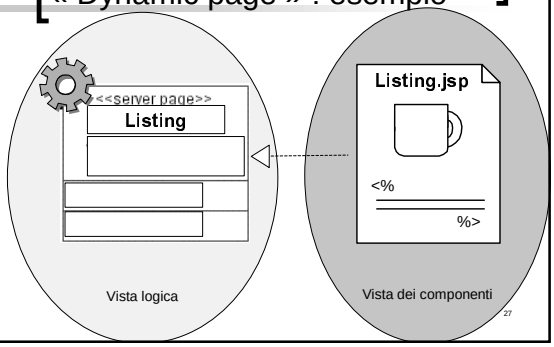
25

« Dynamic page » : esempio



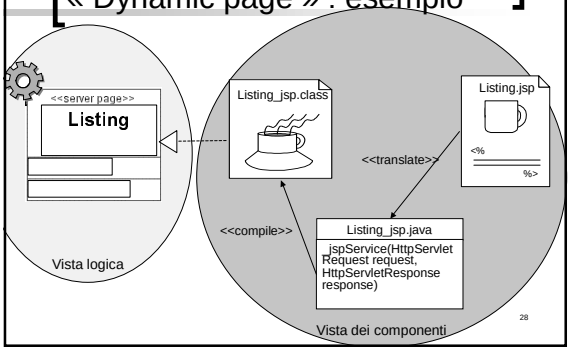
26

« Dynamic page » : esempio



27

« Dynamic page » : esempio



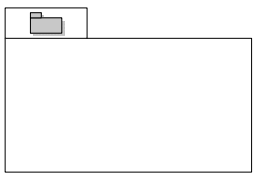
28

Vista dei componenti – 2

- Cosa ci servirà allora? Modelli per
  - File fisici statici (".html")
  - File fisici lato server (".jsp")
  - Physical root
  - Altri file che comporranno le pagine

29

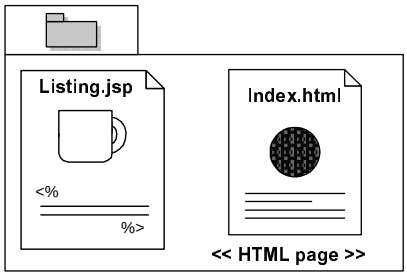
« Physical root »



- Un'astrazione di una gerarchia di file.
- I client richiedono pagine statiche e dinamiche riferendosi a questa gerarchia.
- « Physical root » ~ cartella del file system

30

« Physical root » : esempio



31

Vista dei componenti – 2

- Cosa ci servirà allora? Modelli per
  - File fisici statici (“.html”)
  - File fisici lato server (“.jsp”)
  - Physical root
  - Altri file che comporranno le pagine (da includere del package dei componenti)

32

Altri file…

- « Script Library »
- « Image »
- « Stylesheet »
- ...

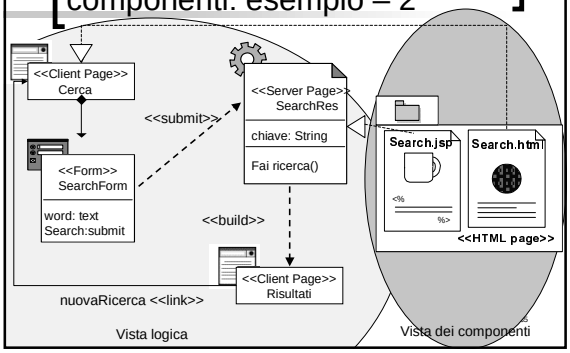
33

Vista logica vs. vista dei componenti: esempio

- Supponiamo di dover modellare un’applicazione web, dove c’è:
  - una pagina html di ricerca con un campo di input (input type=text);
  - una pagina jsp che effettua una ricerca,
  - e costruisce dinamicamente una pagina con i risultati,
  - che contiene un link alla pagina iniziale
- (tipo google ma molto più semplice…)

34

Vista logica vs. vista dei componenti: esempio – 2



Conclusioni

- Con le estensioni web per l’UML siamo in grado di modellare l’applicazione web a due livelli:
  - un livello fisico, con la vista dei componenti
  - un livello più astratto, con la vista logica
- Siamo in grado di mantenere la tracciabilità ed integrità anche in applicazioni web, ancora più soggette a cambiamenti delle normali applicazioni.

36

(mie) conclusioni

- Ma serve? Per fare quattro pagine personali no…
- Ma su sistemi complessi?
  - È difficile cambiare una form senza avere ripercussioni sulle pagine jsp, servlet, DB…
  - Meglio allora che il sistema sia chiaro da un punto di vista più astratto per poi metterci le mani
- Tool automatici? C'è qualcosa, ma poco…

37

Bibliografia

- Applicazioni web con UML (2ed) – *Jim Conallen* – 2003
- Building Web Applications with UML 2ed – *Jim Conallen* – 2002
- Building Web Applications with UML 1ed – *Jim Conallen* – 2000
- Modeling Web Application Architectures with UML – *Jim Conallen* – 1999
- Web:
  - www.conallen.com
- (Conallen ha le mani in pasta su tutto…)

38

Bibliografia

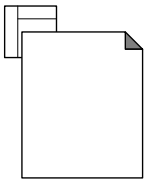
39

Frameset e target

- Un frameset è un tipo di pagina web speciale che divide la sua area di visualizzazione in più pannelli (frame), ognuno contenente la propria pagina web.
- Ognuno dei frame è un target. Un target in un frameset è un frame con un nome per il quale altre pagine client possono richiedere pagine web.

40

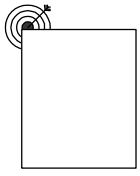
Stereotipi per frameset



- Astrae una pagina HTML che contiene un elemento frameset. I frame sono opzionalmente identificati da un target.

41

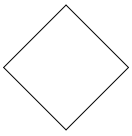
Stereotipi per target



- È uno specifico frame individuato da un nome all'interno di un frameset. È una classe che rappresenta la "destinazione" degli hyperlink

42

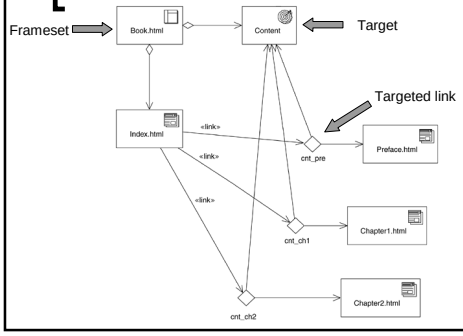
Stereotipi per Targeted link



- È simile ad una associazione «link» tradizionale; ma qui la pagina web richiesta non viene visualizzata nello stesso contenitore della pagina che ha inviato la richiesta, ma nel target specificato

43

Frameset, target…: esempio



44

Stereotipi per le associazioni 2

Stereotipo	Descrizione
«redirect»	Relazione fra una pagina client o server e un'altra pagina client o server. Indica un comando al client affinché richieda un'altra risorsa.
«object»	Relazione di contenimento tra una pagina client e un'altra classe della vista logica, (per es. un'applet, un controllo ActiveX...). Rappresenta un'astrazione degli elementi HTML <object> e <applet>.
«include»	Associazione direzionale fra una «Server page » e un'altra «Server page» o «Client page». Indica che la pagina inclusa viene elaborata, se di tipo dinamico, e che i suoi componenti o risultati vengono utilizzati nella classe contenitore (o classe padre).

45

« Virtual Root »

- Come si risolvono gli URL?
- Prima con il « physical root », che contiene tutti i componenti associati ai file che possono essere richiesti direttamente dal client.
- Ma se bisogna interporre un livello aggiuntivo di corrispondenza fra le risorse HTTP e i file fisici?

46

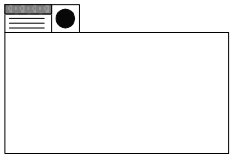
« Virtual Root » - 2

- Si crea una gerarchia virtuale di URL attraverso la creazione di package « Virtual Root » contenenti elementi « HTTP resource » associati alle singole risorse fisiche.
- Il processo di risoluzione di un URL inizia con la ricerca di un package « Virtual Root » con valori di host e contesto corrispondenti a quelli specificati nell'URL.
- Se si trova OK…
- Altrimenti si và nella « Physical root »

47

« Virtual Root » - 3

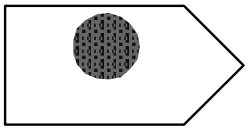
- Contiene componenti Web in una gerarchia del file system visibile ai client. La gerarchia di directory per le risorse richieste corrisponde esattamente alla struttura fisica di directory nel file system



48

« HTTP resource »

- Individua un proxy di un componente reale che si mappa su un URL del sistema. Ogni elemento « HTTP resource » è associato ad un URL valido che può essere richiesto dal sistema



49