

## Capitolo 6

### Test del prototipo

In questo capitolo vengono presentati alcuni siti che sono stati generati dal nuovo prototipo al fine di testare l'intero lavoro sviluppato in questa tesi. Il test è stato condotto definendo per ogni sito un PL e generando i vari programmi e le varie pagine dinamiche corrispondenti alla struttura definita. Gli esempi sviluppati sono:

- 1) un sito complesso: il Sigmod;
- 2) un sito contenente mappe cliccabili;
- 3) un sito contenente form;
- 4) un sito generato utilizzando sia dati provenienti da una base di dati relazionale sia provenienti da una sorgente esterna.

Per ciascun esempio sono stati generati tre siti, uno costituito da sole pagine dinamiche, uno costituito da sole pagine statiche e infine un terzo misto (questo implica che i singoli page-scheme sono stati dichiarati una volta *STATIC* e una volta *DYNAMIC*).

Per ogni sito si riportano le istruzioni Penelope utilizzate per definirlo e alcune figure rappresentanti alcune pagine del sito generato.

#### 6.1 Generazione di un sito complesso: il Sigmod

```

$-----
$                S I G M O D
$                S I T E   S C H E M E
$-----

SCHEME file:/
ON sigmodDiTomas

```

```

$-----
$ I)   HOME PAGE
$-----

DEFINE PAGE Dynamic HomePage : "Issues"
STYLE g:\users\ditomass\wbms43\sites\styleSigmod\SigmodStyle.STY()
AS URL ("index.html");
  yearList: LIST-OF (
    year      : TEXT = <RECORDYEARS.YEAR>;
    numberList : LIST-OF (
      toIssues      : LINK-TO OrdinaryIssuePage UNION
      ProceedingsPage IF ISSUES.CONFNAME LIKE '%'
      (
        URL (<ISSUES.VOLUME>,<ISSUES.NUM>);
        month : TEXT = <ISSUES.MONTH>;
      );
      volume      : TEXT = <ISSUES.VOLUME>;
      number      : TEXT = <ISSUES.NUM>;
      conferenceName : TEXT = <ISSUES.CONFNAME>;
      conferenceYear : TEXT = <ISSUES.CONFYEAR>;
    );
  );
USING RECORDYEARS : (SELECT DISTINCT YEAR
  FROM RECORDISSUES
  ORDER BY YEAR DESC),
  ISSUES : { SELECT RECORDISSUES.*, CONFNAME, CONFYEAR
  FROM RECORDISSUES LEFT JOIN CONFERENCES
  ON (RECORDISSUES.NUM = CONFERENCES.NUM )
  AND (RECORDISSUES.VOLUME = CONFERENCES.VOLUME)}
ORDER BY ISSUES.VOLUME, ISSUES.NUM
END

$-----
$ II)  ORDINARY ISSUE PAGE
$-----

DEFINE PAGE Dynamic OrdinaryIssuePage
AS URL (<RECORDISSUES.VOLUME>,<RECORDISSUES.NUM>);
  volume      : TEXT = <VOLUME>;
  number      : TEXT = <NUM>;
  month       : TEXT = <MONTH>;
  year       : TEXT = <YEAR>;
  sectionList : LIST-OF (
    sectionName : TEXT = <SECTIONS.SECTION>,
  OFFSET(<SECTIONS.SECTIONPOSITION>);
    articles    : LIST-OF (
      toArticle : LINK-TO ExternalPage (
        URL(<ARTICLES.REF>);
        title : TEXT = <ARTICLES.TITLE>,
  OFFSET(<ARTICLES.ARTICLECODE>,<ARTICLES.POSITION>);
      );
      initPage  : TEXT = <INITPAGE>;
      endPage   : TEXT = <ENDPAGE>;
      authors   : LIST-OF (
        author: TEXT = <AUTHORS.AUTHOR>, OFFSET(<AUTHORS.AUTHORPOSITION>);
      );
    );
USING RECORDISSUES,
  SECTIONS: {SELECT ARTICLES.VOLUME, ARTICLES.NUM, ARTICLES.SECTION, MIN(POSITION) AS
  SECTIONPOSITION
  FROM ARTICLES
  GROUP BY VOLUME, NUM, SECTION} DISTINCT,
  ARTICLES DISTINCT,
  AUTHORS DISTINCT
WHERE RECORDISSUES.TYPE LIKE 'Ordinary%'
ORDER BY SECTIONS.SECTIONPOSITION, ARTICLES.POSITION, AUTHORS.AUTHORPOSITION
END

$-----

```

```

$ III) PROCEEDINGS PAGE SCHEME
$-----

DEFINE PAGE Dynamic ProceedingsPage : "Proceedings Page"
AS URL (<CONF.VOLUME>, <CONF.NUM>);
  volume      : TEXT = <VOLUME>;
  number      : TEXT = <NUM>;
  month       : TEXT = <MONTH>;
  year        : TEXT = <YEAR>;
  conference  : TEXT = <CONFNAME>;
  date        : TEXT = <CONFDATE>;
  confyear    : TEXT = <CONFYEAR>;
  location    : TEXT = <LOCATION>;
  sectionList : LIST-OF (
    sectionName : TEXT = <SECTIONS.SECTION>, OFFSET(<SECTIONS.SECTIONPOSITION>;
      articles : LIST-OF (
        title : TEXT = <ARTICLES.TITLE>, OFFSET(<ARTICLES.ARTICLECODE>,<ARTICLES.POSITION>);
          authors : LIST-OF (
            author : TEXT = <AUTHORS.AUTHOR>,
            OFFSET(<AUTHORS.AUTHORPOSITION>);
          );
          initPage : TEXT = <INITPAGE>;
          endPage : TEXT = <ENDPAGE>;
          Toindex : LINK-TO IndexTermsPage (
            URL(<ARTICLECODE>);
          );
          index : TEXT = "Abstract, Index Terms and Categories";
          fullText : LINK-TO ExternalPage (
            URL(<REF>);
            size : TEXT = <DIM> ;
          );
        );
      );
    );
);

USING CONF: (SELECT CONFERENCES.*, RECORDISSUES.MONTH, RECORDISSUES.YEAR
FROM CONFERENCES, RECORDISSUES
WHERE CONFERENCES.VOLUME=RECORDISSUES.VOLUME AND
CONFERENCES.NUM=RECORDISSUES.NUM) DISTINCT,
SECTIONS: {SELECT ARTICLES.VOLUME, ARTICLES.NUM, ARTICLES.SECTION, MIN(POSITION) AS
SECTIONPOSITION
FROM ARTICLES
GROUP BY VOLUME, NUM, SECTION} DISTINCT,
ARTICLES DISTINCT,
AUTHORS DISTINCT
ORDER BY SECTIONS.SECTIONPOSITION, ARTICLES.POSITION, AUTHORS.AUTHORPOSITION
END

$-----
$ IV) INDEX TERM PAGE
$-----

DEFINE PAGE Dynamic IndexTermsPage : "Index Terms and Categories"
AS URL (<ArtsView.ARTICLECODE>);
  title : TEXT = <TITLE>, OFFSET(<ARTICLECODE>,<POSITION>);
  authors : LIST-OF (
    author : TEXT = <AUTHORS.AUTHOR>, OFFSET(<AUTHORS.AUTHORPOSITION>);
  );
  confName : TEXT = <CONFNAME>;
  confYear : TEXT = <CONFYEAR>;
  volume : TEXT = <VOLUME>;
  number : TEXT = <NUM>;
  initPage : TEXT = <INITPAGE>;
  endPage : TEXT = <ENDPAGE>;
  fullText : LINK-TO ExternalPage (
    URL (<REF>);
    size : TEXT = <DIM>;
  );
  abstract : TEXT = <ABSTRACT>;
  generalTerms : LIST-OF (
    term : TEXT = <ARTICLETERMS.TERM>;
  );
);

```

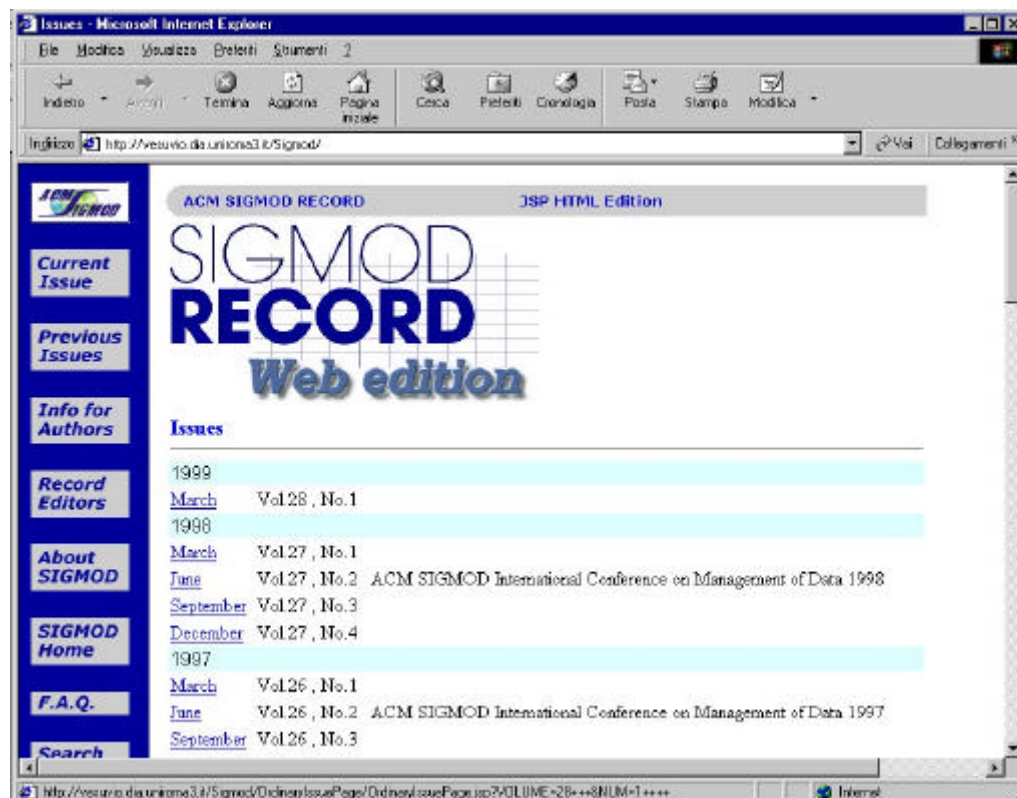
```

categoryAndSubjectDescriptors : LIST-OF (
    category : TEXT = <ARTICLECATEGORIES.CATEGORY>;
    content : TEXT = <ARTICLECATEGORIES.DESCRPTION>;
);
USING ArtsView : (SELECT ARTICLES.*, CONFERENCES.CONFNAME, CONFERENCES.CONFYEAR,
    ABSTRACTS.ABSTRACT
    FROM ARTICLES, CONFERENCES, ABSTRACTS
    WHERE ARTICLES.VOLUME = CONFERENCES.VOLUME AND
    ARTICLES.NUM = CONFERENCES.NUM AND
    ARTICLES.ARTICLECODE = ABSTRACTS.ARTICLECODE),
    AUTHORS,
    ARTICLETERMS,
    ARTICLECATEGORIES
ORDER BY AUTHORPOSITION
END

```

Le figure 6.1, 6.2, 6.3, 6.4 rappresentano le quattro pagine dinamiche generate, corrispondenti ai page-scheme *HomePage*, *OrdinaryIssuePage*, *ProceedingsPage*, *IndexTermsPage*. Come può essere osservato solo la *HomePage* è una pagina unica che pertanto non richiede parametri d'istanza.

In figura 6.5 viene riportata la pagina statica relativa al page-scheme *HomePage* all'interno della quale sono presenti dei link alle altre pagine che sono dinamiche (si osservi l'URL che compare nella barra di stato).



**Figura 6.1:** La HomePage dinamica del sito

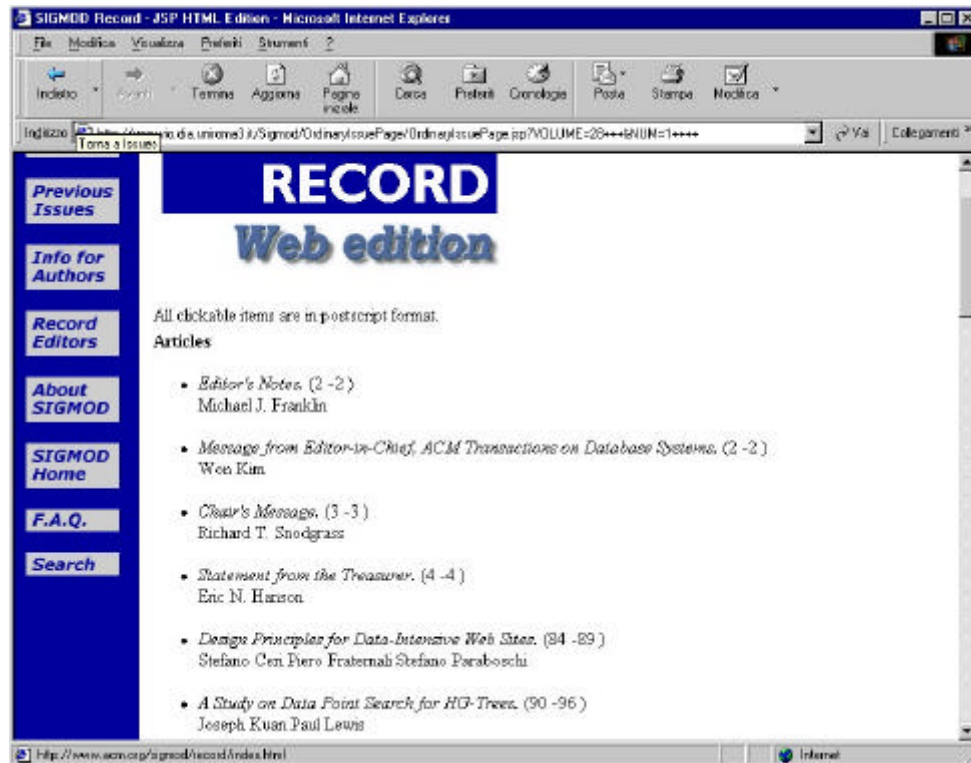


Figura 6.2: La pagina dinamica generata “*OrdinaryIssuePage.jsp*”

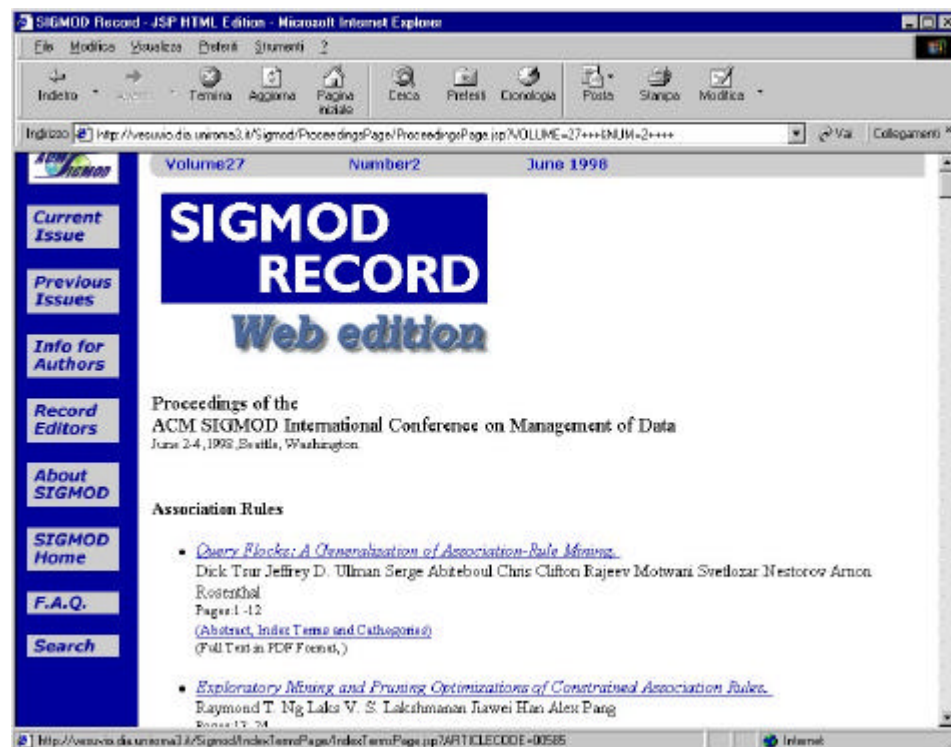


Figura 6.3: La pagina dinamica generata “*ProceedingsPage.jsp*”

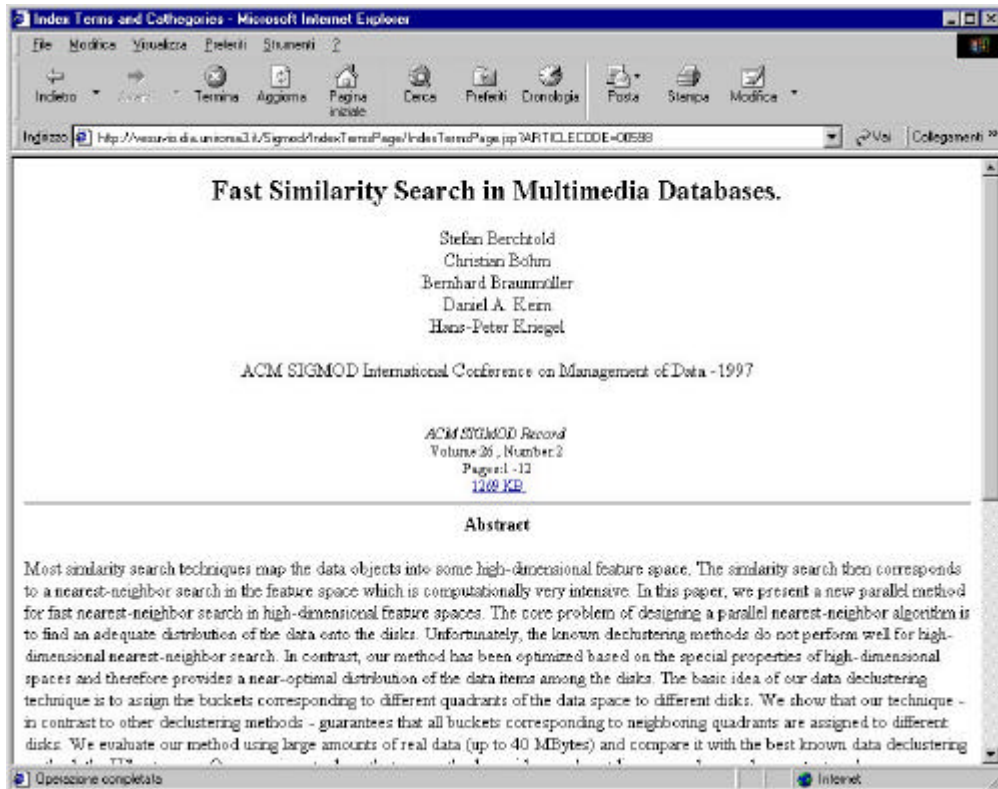


Figura 6.4: la pagina dinamica generata "IndexTermsPage.jsp"

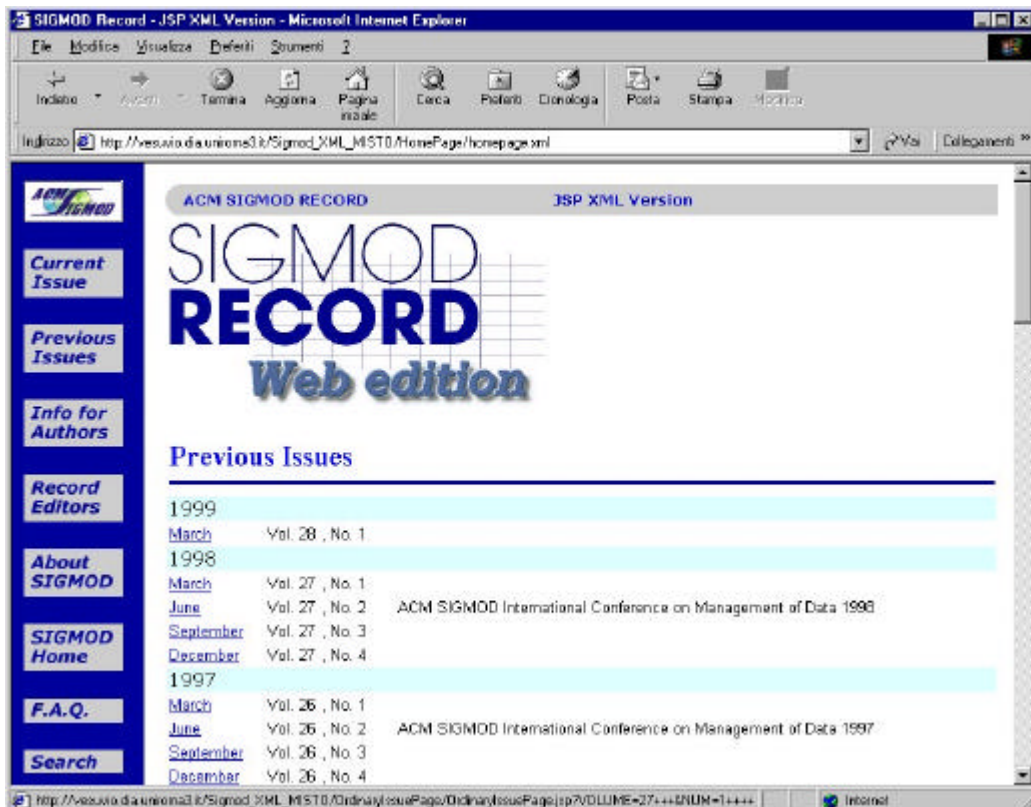


Figura 6.5: La HomePage statica del sito al cui interno sono presenti link dinamici

## 6.2 Generazione di un sito contenente mappe *cliccabili*

Riportiamo in questo caso i singoli page-scheme con le pagine generate.

```

$-----$
$                               M A P
$                               S I T E   S C H E M E
$-----$

ON linea07

DEFINE PAGE Dynamic Home : "La.T.I.B.I. - Prodotti - Home Page"
STYLE d:\pop\prodotti\dati\07coll01\Template\prodottiPop.STY(Bacino del Camastra: Aste
Fluviali)
AS URL("index.html");
  Mappa : MAP (
    Immagine : IMAGE = <FILE600x500>;
    AllaCarta : LINK-TO CartaAstaP(
      URL (<IDASTA>);
      Coord : TEXT = <COORDINATE>;
    );
  );
  Linea : LINK-TO ExternalP (
    URL (<ALLALINEA>);
    AllaLinea : TEXT = <CODICELINEA>;
  );
  TitoloLinea : TEXT = <TITOLO>;
  Descrizione : TEXT = <DESCRIZIONE>;
USING TmpQuadroGenerale : (select coordinateQuadroGenerale.*, quadroGenerale.FILE600x500
  from quadroGenerale, coordinateQuadroGenerale
  where quadroGenerale.IDQUADRO=coordinateQuadroGenerale.IDQUADRO) DISTINCT,
  InformazioniGenerali
END

```

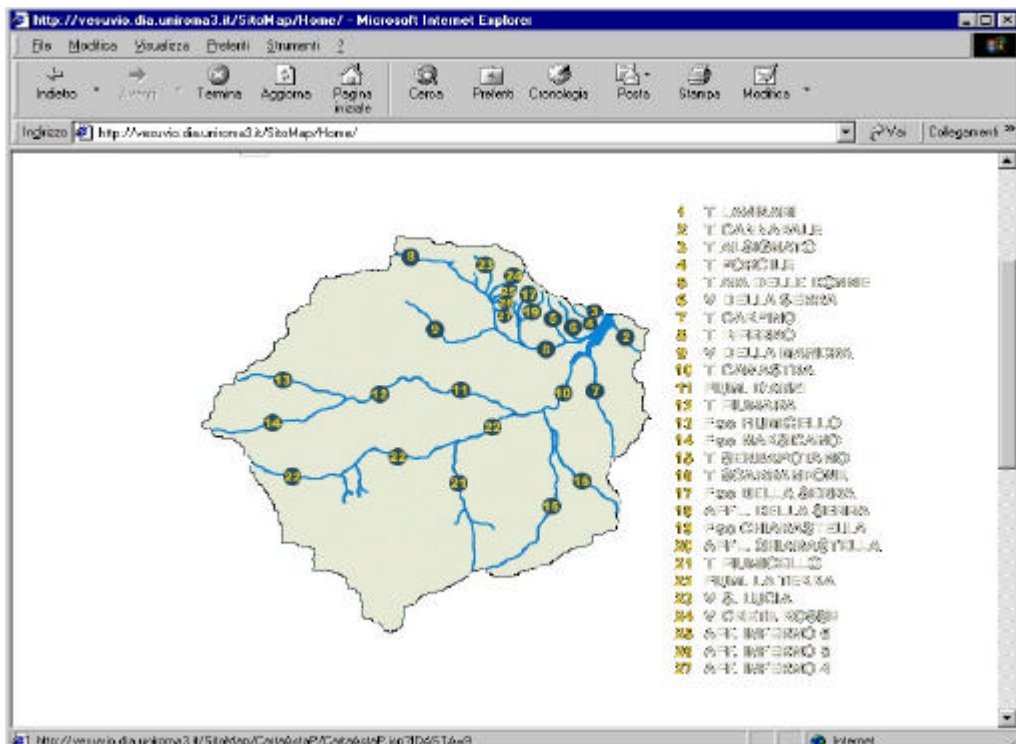


Figura 6.6: La pagina dinamica generata "Home.jsp"

```

DEFINE PAGE Dynamic CartaAstaP : "La.T.I.B.I. - Prodotti"
STYLE d:\pop\prodotti\dati\07coll101\Template\prodottiPop.STY(Mappa in scala 1:50000
AS URL(<IDASTA>);
Nome : TEXT = <NOMEASTA>;
Lunghezza : TEXT = <LUNGHEZZA>;
QuotaMonte : TEXT = <QUOTAMONTE>;
QuotaValle : TEXT = <QUOTAVALLE>;
Morfologia : TEXT = <MORFOLOGIA>;
Superficie : TEXT = <SUPERBACINO>;
Pendenza : TEXT = <PENDENZAMEDIA>;
Profilo : LINK-TO ProfiloAstaP (
    URL (<IDASTA>);
    AlProfilo : TEXT = "Profilo";
);

Mappa : MAP (
    Immagine : IMAGE = <FILEASTA>, SIZE <LARGHASTA> <ALTASTA>;
    AllaBriglia : LINK-TO BrigliaP
        UNION SogliaP
            IF TIPO LIKE '%soglia%'
        UNION ArgineP
            IF TIPO LIKE '%argine%'
        UNION SpondaleP
            IF TIPO LIKE '%spondale%'
        UNION PennelloP
            IF TIPO LIKE '%pennello%' (
                URL (<IDOPERA>);
                Coord : TEXT = <COORDINATE>;
            );
);

Note : TEXT = <ANNOTAZIONI>;
USING TmpAste : (select coordinateAste.COORDINATE, coordinateAste.IDOPERA,
coordinateAste.TIPO, ASTE.IDASTA, ASTE.NOMEASTA, ASTE.LUNGHEZZA, ASTE.QUOTAMONTE,
ASTE.QUOTAVALLE, ASTE.MORFOLOGIA, ASTE.SUPERBACINO, ASTE.PENDENZAMEDIA, ASTE.FILEASTA,
Aste.LARGHASTA, ASTE.ALTASTA, ASTE.ANNOTAZIONI
from ASTE, coordinateAste
where ASTE.IDASTA=coordinateAste.IDASTA) DISTINCT
END
    
```

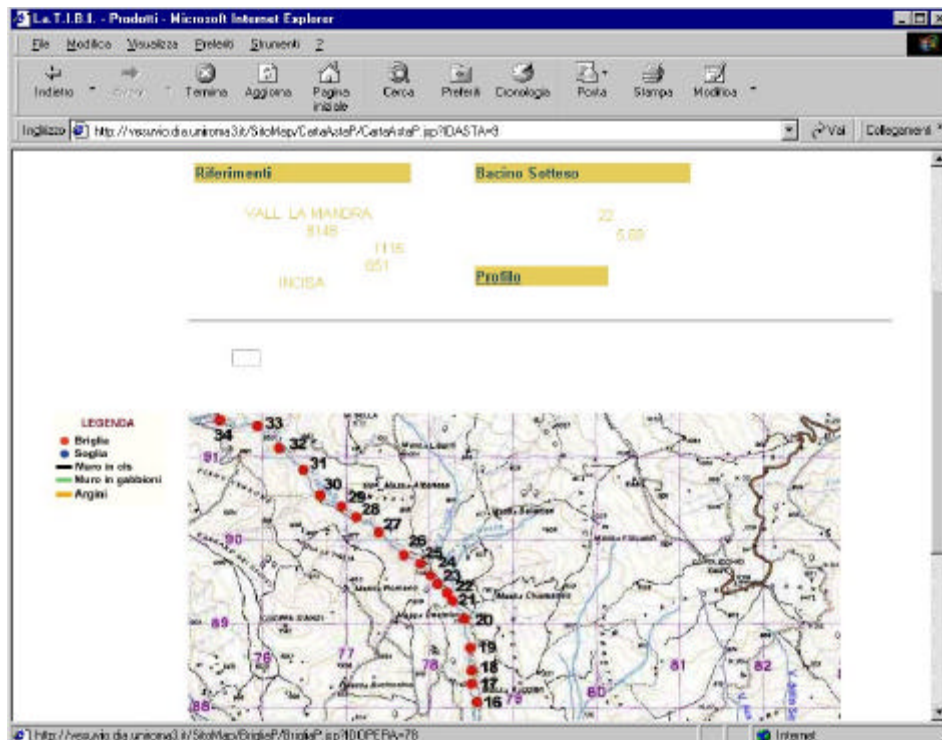


Figura 6.7: La pagina dinamica generata "CartaAstaP.jsp"



## 6.3 Generazione di un sito contenente FORM

```

$-----
$                               F O R M
$                               S I T E   S C H E M E
$-----
ON convdifa

DEFINE PAGE Dynamic AbstractPage : "Web Base Management System"
AS URL ("index.html");
  Area : TEXT = <ABSTRACT.AREA>;
  PaperList : LIST-OF (
    Autori : TEXT = <ABSTRACT.AUTORI>;
    Titolo : TEXT = <ABSTRACT.TITOLO>;
    RifFile : TEXT = <ABSTRACT.FILE>;
    FormatoFile : TEXT = <ABSTRACT.FORMATOFILE>;
    Elimina : FORM (
      action : TEXT = "EliminaArticolo.bat";
      method : TEXT = "POST";
      enctype: TEXT = "text/html";
      inputHide : HIDDEN = <ABSTRACT.CODICE>;
      inputSub : SUBMIT = "Remove Paper";
    );
    Modifica : FORM (
      action : TEXT = "ModificaArticolo.bat";
      method : TEXT = "GET";
      enctype: TEXT = "text/html";
      inputHide : HIDDEN = <ABSTRACT.CODICE>;
      inputSub : SUBMIT = "Modify Paper";
    );
  );
USING ABSTRACT DISTINCT
ORDER BY ABSTRACT.AUTORI, ABSTRACT.TITOLO
END

```



Figura 6.8: La pagina dinamica generata "AbstractPage.jsp"

## 6.4 Generazione di un sito con dati provenienti da sorgenti esterne

Per testare il funzionamento del prototipo quando i dati provengono da sorgenti esterne sono stati creati una serie di PL che testano, in maniera esaustiva, il funzionamento, vagliando tutte le possibili combinazioni che si possono avere, sia nella clausola USING sia a livello di attributi.

Nel PL sono stati definiti due page-scheme *EducationPageList* e *EducationPage*. E' su quest'ultimo che viene condotta la sperimentazione, al suo interno sono definiti in sequenza tre attributi semplici, tre liste nidificate, un altro attributo semplice ed infine una lista. All'interno della clausola USING sono definite 4 viste, due delle quali utilizzano dati provenienti da sorgenti esterne.

```

$-----
$           E X T E R N A L  -- S O U R C E
$           S I T E           S C H E M E
$-----
$ Penelope Code written by
$ G. Di Tomassi (ditomass@dia.uniroma3.it)
$-----

SCHEME file:/d:/users/ditomass/Demo_Tesi/ExternalSources\
ON CourseList

$ -----
$  EDUCATION PAGE LIST
$ -----

DEFINE PAGE Static EducationPageList : "Education List"
STYLE sites\styles\verticalXML.STY(Our Courses)
AS URL("EducationList_Modificato2");
    ListaEducation: LIST-OF ( ToEducation : LINK-TO EducationPage(
                                URL (<Universita>);
                                GroupName : TEXT = <Universita>;
                                );
USING      Info
END

$ -----
$  EDUCATION
$ -----

DEFINE PAGE Dynamic EducationPage : "Education"
STYLE sites\styles\verticalXML.STY(Our Courses)
AS URL(<Universita>);

universita      : TEXT = <Universita>;
sede            : TEXT = <Sede>;
citta          : TEXT = <Citta>;
ListaCorsodilaurea: LIST-OF ( Corsolaurea : TEXT = <CL.Cl>;

```

```

ListaProfessori: LIST-OF (  Cognome : TEXT = <Cognome>;
                           nome : TEXT = <Nome>;
                           email : TEXT = <email>;
                           ListaCorsi : LIST-OF (
                               CourseNome : TEXT = <CorsoNome>;
                                               );
                           );

AttributoCostante : TEXT = "Attributo di livello 0 Costante";

ListaOrario : LIST-OF (
    orario : TEXT = <Orario>;
);

USING  Info :(SELECT Sede, Citta, Universita
             FROM http://vesuvio.dia.uniroma3.it/External_Sources/Query_Info.txt),

       CL,
       Prof : (SELECT Cl, Professore.Cognome, Nome, email
              FROM Professore),

       ProfCourse : (SELECT  Cognome, CorsoNome
                   FROM http://vesuvio.dia.uniroma3.it/External_Sources/Query_ProfCourse.txt),

       OrarioCourse: (SELECT Orario FROM OrarioProf)

END

```

L'input stream ritornato dalle due sorgenti esterne è il seguente:

```

<table>
<tuple><Citta>ROMA</Citta><Sede>via della vasca navale 76</Sede><Universita>Roma Tre</Universita></tuple>
<tuple><Citta>ROMA</Citta><Sede>via Salaria</Sede><Universita>La Sapienza</Universita></tuple>
</table>

```

**Figura 6.9:** InputStream ritornato dalla sorgente dati “*Query\_Info.txt*”

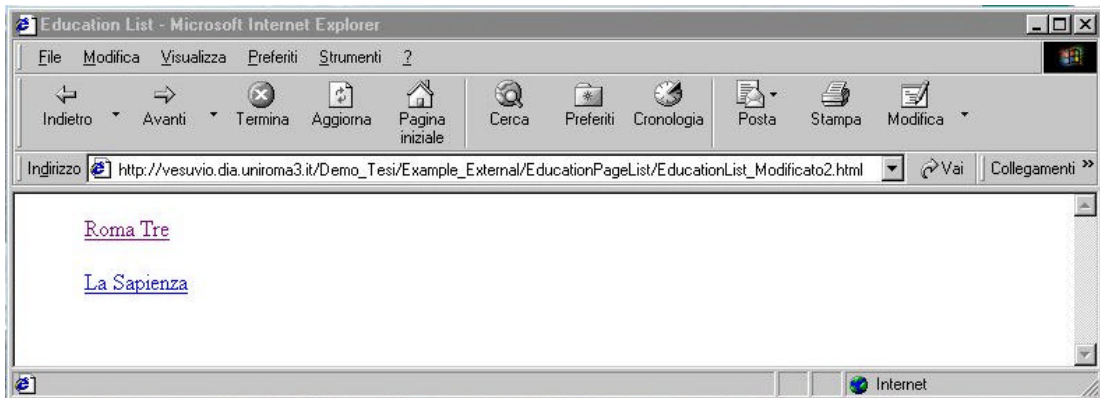
```

<table>
<tuple><Cognome>Di Battista</Cognome><CorsoNome>Informatica Teorica</CorsoNome></tuple>
<tuple><Cognome>Di Battista</Cognome><CorsoNome>Impianti di Elaborazione</CorsoNome></tuple>
<tuple><Cognome>Atzeni</Cognome><CorsoNome>Sistemi Informativi</CorsoNome></tuple>
<tuple><Cognome>Atzeni</Cognome><CorsoNome>Basi di Dati</CorsoNome></tuple>
<tuple><Cognome>Neri</Cognome><CorsoNome>Teoria dei Segnali</CorsoNome></tuple>
<tuple><Cognome>Cerri</Cognome><CorsoNome>Meccanica Applicata</CorsoNome></tuple>
<tuple><Cognome>Assanto</Cognome><CorsoNome>Optoelettronica</CorsoNome></tuple>
<tuple><Cognome>Cialdea</Cognome><CorsoNome>Fondamenti II</CorsoNome></tuple>
<tuple><Cognome>Cialdea</Cognome><CorsoNome>Intelligenza Artificiale</CorsoNome></tuple>
</table>

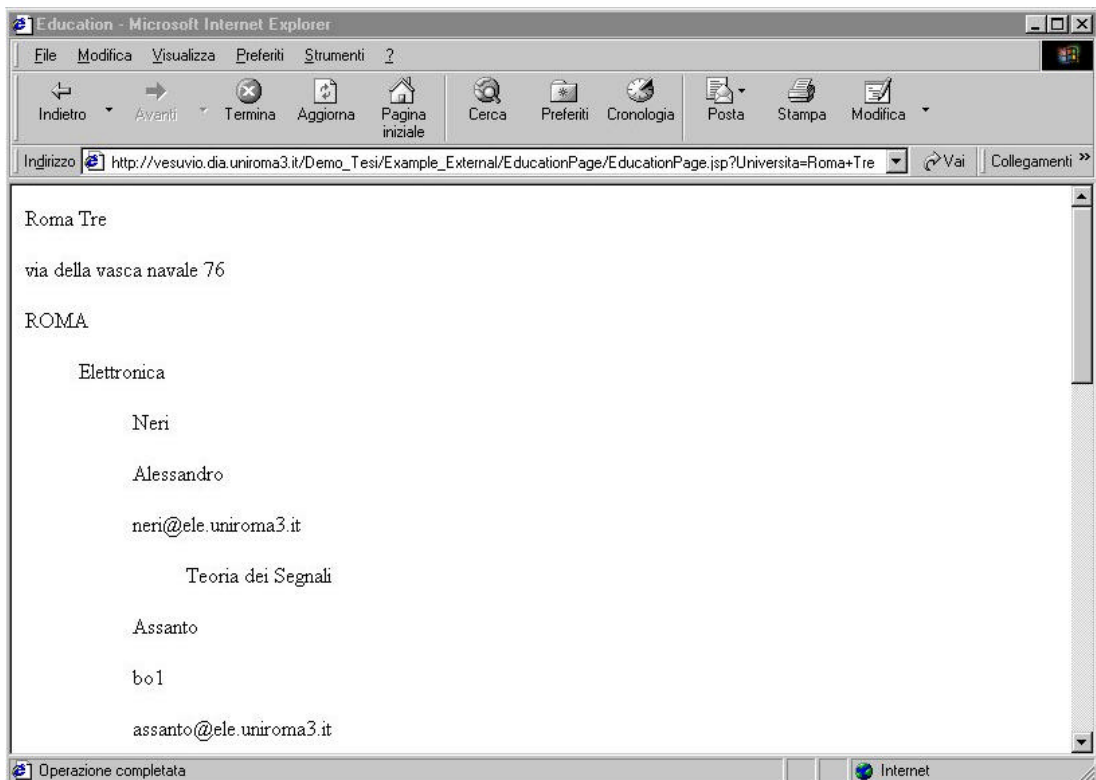
```

**Figura 6.10:** InputStream ritornato dalla sorgente dati “*Query\_ProfCourse.txt*”

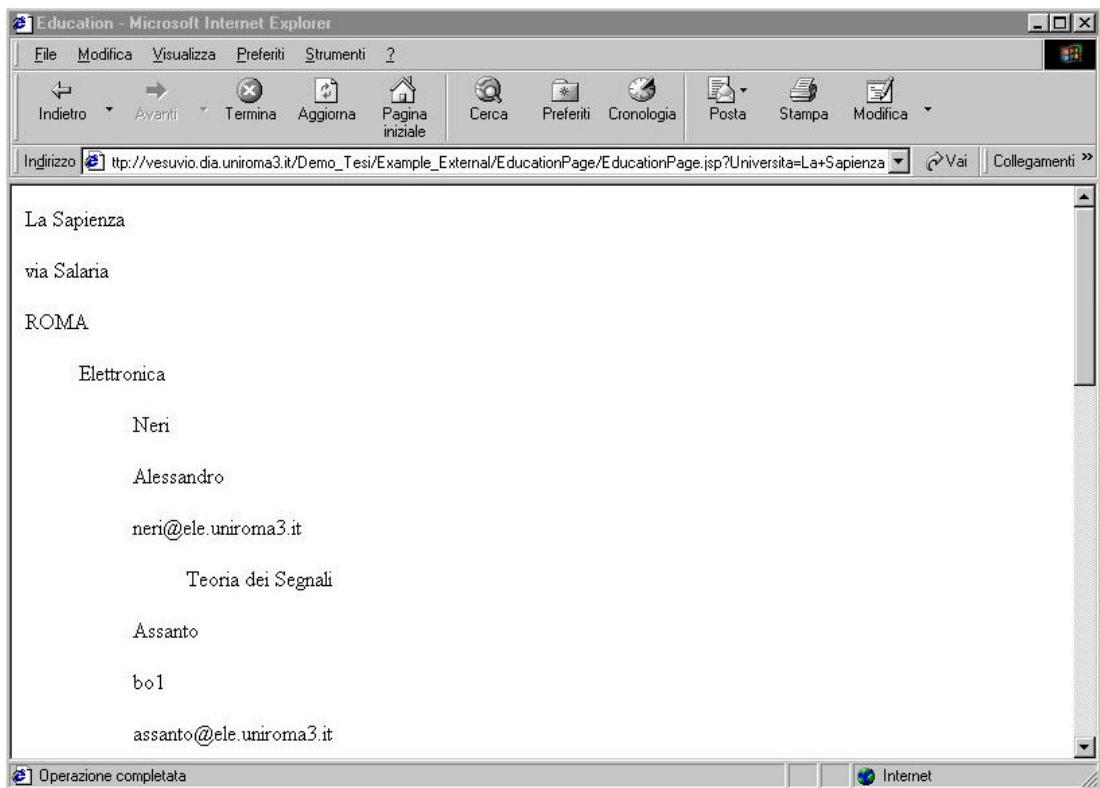
Nelle seguenti figure vengono riportate le pagine generate con Penelope e visualizzate attraverso un browser.



**Figura 6.11:** Pagina statica generata relativa al page-scheme “*EducationPageList*”



**Figura 6.12:** Pagina dinamica generata relativa al page-scheme “*EducationPageList*” ed invocata con parametro “Roma Tre”



**Figura 6.13:** Pagina dinamica generata relativa al page-scheme  
“*EducationPageList*” ed invocata con parametro “La Sapienza”