Lorenzo Cioni

Il repository: uno strumento per l'archiviazione e la distribuzione sul web

(work in progress)

Introduzione

In queste brevi note viene descritto a grandi linee un progetto portato avanti dall'autore presso il Laboratorio di Linguistica e mirante alla definizione e alla implementazione di un servizio di distribuzione di articoli sul web.

Il progetto, nel seguito detto QLL-Repository, ha portato alla definizione e alla implementazione di un semplice sistema per l'interrogazione e/o il download degli articoli pubblicati sui *Quaderni del Laboratorio di Linguistica* (QLL) a partire dal 1994.

QLL-Repository: scopo del progetto

Il QLL-Repository rappresenta uno strumento per l'interrogazione di un semplice "data base" contenente gli articoli pubblicati sugli ultimi numeri dei QLL, articoli che continuano ad essere accessibili attraverso un normale ftp server.

Il repository (vedi figura 1) consente l'esecuzione di interrogazioni di tipo ***query*** e di tipo ***download*** e permette di ricevere via e-mail sia il risultato della query sia (nel secondo caso) gli articoli selezionati.

L'interrogazione di tipo ***query*** può essere di tipo generico (ovvero per tutti i numeri dei QLL presenti nel data base per tutti gli autori e su tutti gli articoli) oppure è possibile specificare indipendentemente:

- il volume desiderato fra quelli disponibili,
- un pattern per la ricerca per nome nel campo "Autore" e
- un pattern per la ricerca per titolo nel campo "Titolo".

Il pattern per la ricerca per nome prevede che sia specificata almeno una delle iniziali del nome dell'autore di cui si vogliono ricercare gli articoli presenti nel data base. La ricerca produce come risultato l'elenco dei file corrispondenti che contengono il pattern specificato nella codifica del titolo nel nome del file (vedi oltre).

Il pattern per la ricerca per titolo consiste in una stringa che viene ricercata come insieme di caratteri contigui all'interno della porzione del nome dei singoli file che codifica il nome dei corrispondenti articoli. Il campo non prevede l'uso di operatori logici.

Il risultato della ricerca viene presentato al richiedente tramite una pagina HTML dinamica e, se questi ha fornito un indirizzo di e-mail "sintatticamente corretto[1]", gli/le viene spedito via e-mail.

In modo analogo l'interrogazione di tipo ***download*** può essere di tipo generico, sebbene debba essere limitata ad uno solo dei numeri disponibili dei Quaderni, oppure può essere ristretta ulteriormente mediante l'introduzione di analoghi vincoli sul titolo e/o sull'autore o sugli autori.

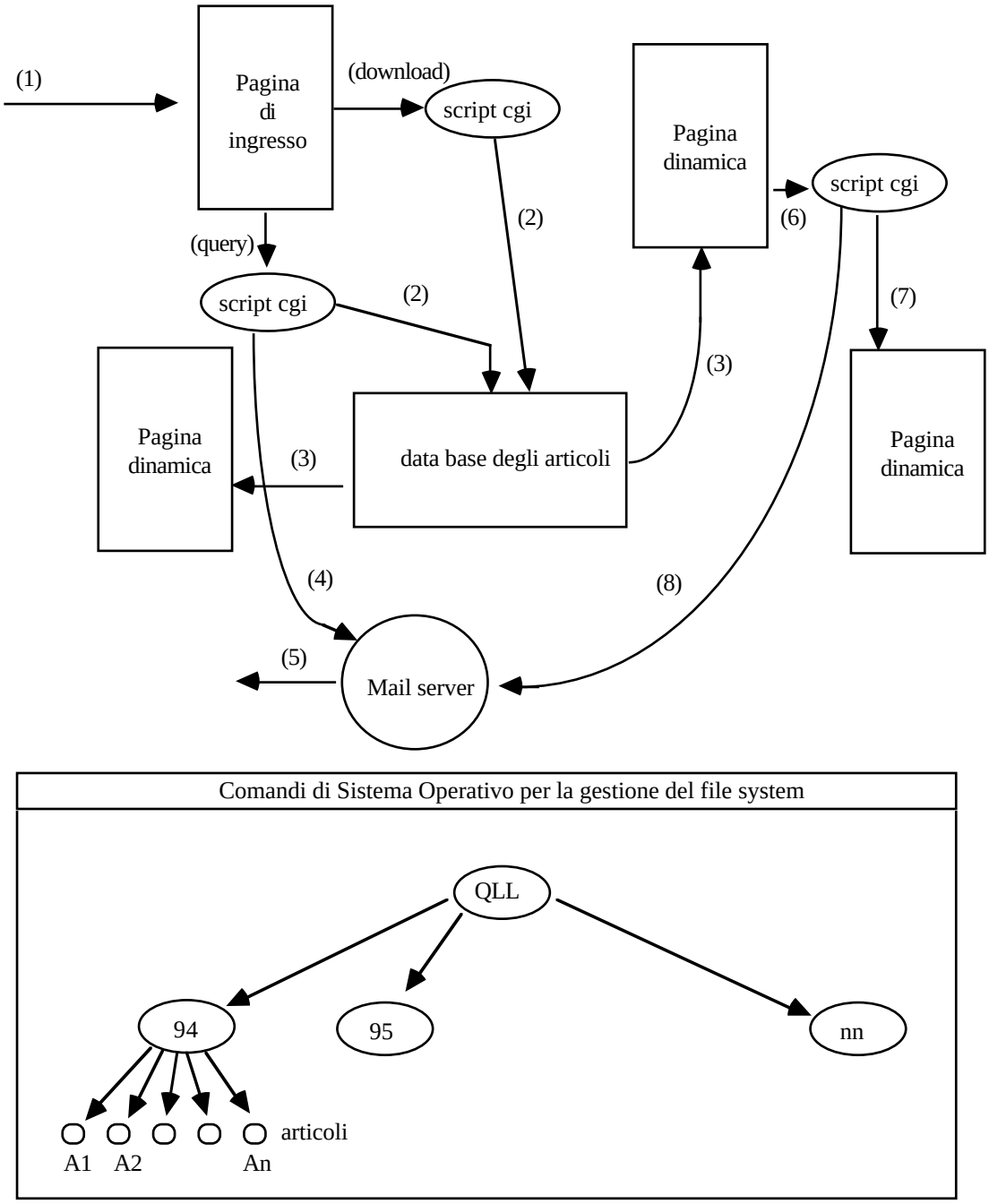
Come risultato di una richiesta di tipo ***download*** l'utente ottiene una pagina HTML dinamica contenente l'elenco dei file che soddisfano i criteri di scelta da lui/lei impostati e che gli/le saranno spediti all'indirizzo di posta elettronica da lui/lei specificato al momento di impostare i dati per il download.

Tale pagina consente inoltre la selezione di un metodo per la codifica di ogni singolo file[2] prima che questi siano spediti al richiedente all'indirizzo da lui/lei specificato. I metodi disponibili sono:

- none (valore di default) gli articoli sono spediti nel formato di memorizzazione, ovvero postscript compresso con gzip;
- uuencode gli articoli sono codificati con l'utility Unix **uuencode** e poi spediti via e-mail;
- mime gli articoli sono codificati utilizzando l'utility Unix **mmencode** in base-64 e poi spediti via e-mail.

1 Un indirizzo di posta eletronica è sintatticamente corretto se è del tipo **nome@host.dominio** oppure **nome@dominio** dove **nome** può essere nella forma **nome.cognome** o viceversa.

2 Un articolo è contenuto in un file per cui i due termini possono essere visti come sinonimi in questo contesto.



Ai è del tipo {N*C*}.titolo_abbreviato.est1.est2 in cui N è l'iniziale del nome, C quella del cognome e sono presenti due estensioni (vedi il testo).

Figura1 Struttura interna del data base degli articoli

Qualora lo desideri, l'utente, a questo livello, può annullare il download semplicemente selezionando l'icona etichettata "Click here to go back" presente sulla pagina HTML dinamica corrente oppure può proseguire (pulsante OK) ed ottenere i file da lui selezionati via e-mail.

QLL-Repository: struttura astratta

La struttura astratta del QLL-Repository è composta da tre pagine HTML, una statica e due dinamiche create "on the fly" sulla base dei dati immessi e sia del tipo sia del risultato della interrogazione (vedi figura 1).

La pagina statica (detta *Pagina di ingresso* in figura 1) presenta all'utente i valori di default e gli/le permette di raffinare tali valori in modo sia di modificare il motivo della richiesta sia di restringere l'ambito della ricerca stessa.

I valori di default sono i seguenti:

Issues: All
Authors: All
Papers: All
e-mail: none
motivo: query

L'utente può modificare tali valori agendo su dei campi scrivibili o su dei radio button oppure su dei menù a tendina.

Nel caso di una interrogazione di tipo ***query***, sulla base dei dati immessi viene creata una pagina HTML dinamica (vedi in figura 1 i passi (2) e (3)) che visualizza il risultato della ricerca. Tale pagina dinamica viene prodotta da uno script CGI sulla base del risultato di una ricerca all'interno di una porzione del file system del sistema ospite e il suo contenuto viene spedito all'indirizzo specificato in fase di impostazione dei parametri (vedi in figura 1 i passi (4) e (5)).

Nel caso di una interrogazione di tipo ***download***, sulla base dei dati immessi viene creata una pagina HTML dinamica (vedi in figura 1 i passi (2) e (3)) la quale consente sia di annullare l'operazione sia di proseguirla selezionando uno dei metodi di codifica disponibili.

In questo caso (vedi in figura 1 i passi (6) e (7)) viene richiamato un ulteriore script CGI che si preoccupa sia di effettuare la codifica richiesta sia di interagire con il Mail server per la spedizione degli articoli richiesti (vedi in figura 1 i passi (8) e (5)).

QLL-Repository: struttura fisica, implementazione e note di utilizzo

Il QLL-Repository ha una struttura interna molto semplice, basata su una pagina HTML statica e due pagine create dinamicamente tramite script CGI. Tali script sono stati realizzati mediante due semplici programmi scritti usando il linguaggio di programmazione della Bourne Shell. Il tutto si interfaccia con una porzione del file system del sistema ospite che, gestito tramite alcuni comandi standard del sistema operativo Unix, può essere visto come un semplice data base.

Lo script CGI invocato dalla "Pagina di ingresso" (vedi figura 1) ha lo scopo di interrogare il data base degli articoli utilizzando i dati immessi dall'utente e sulla base del risultato dell'interrogazione creare una pagina dinamica (vedi figura 1, passi (2) e (3)).

Nel caso di una interrogazione di tipo ***download***, qualora l'utente imposti l'interrogazione in modo da ricevere via posta elettronica i file prodotti dalla ricerca all'interno del data base degli articoli, oltre a tale script ne viene eseguito un altro il cui compito è duplice: codificare gli articoli e spedirli al richiedente interagendo con il mail server

Gli script CGI interagiscono con il data base degli articoli (vedi figura 1, passo (2)) utilizzando comandi Unix standard quali **ls**, **grep** e simili.

Il data base è implementato in modo semplice come una porzione del file system del sistema ospite con la convenzione che ognuno dei fascicoli dei QLL disponibili è memorizzato in una directory separata. La motivazione di base di una tale organizzazione è quella di definire una struttura semplice, flessibile e gestibile direttamente anche con un server ftp. I singoli articoli sono pertanto memorizzati come file nelle directory corrispondenti ai fascicoli dei QLL. Il formato dei file è, per i file degli articoli veri e propri, postscript (suffisso **ps**) compresso con **gzip** (suffisso **gz**) mentre per altri file di testo, come gli indici e i riassunti, è testo (suffisso **txt**) compresso con **gzip**.

Il nome di ogni file contiene codificate alcune informazioni di base dell'articolo corrispondente secondo la regola seguente:

{N*C*}.titolo_abbreviato.est1.est2

in cui N e C sono rispettivamente le iniziali del nome e del cognome di un autore, le {} indicano che possono esserci più coppie N*C*, e il carattere * indica la possibilità di più di una iniziale per il nome e per il cognome (ad esempio Vittorio Di Tomaso da cui VDT ossia NCC).

Il titolo dell'articolo compare in forma abbreviata nella parte titolo_abbreviato del nome del file e le estensioni est1 e est2 sono quelle già indicate.

Da un punto di vista utente, l'accesso al QLL-Repository avviene tramite una pagina web detta "Pagina di ingresso" raggiungibile a partire dall'URL
"http://alphalinguistica.sns.it/publications.html"
seguendo i link "Quaderni del Laboratorio di Linguistica" e poi "Query & Download Engine" (vedi figura 1, passo (1)).

Tramite tale pagina l'utente può impostare i valori dei parametri per una richiesta di tipo ***query*** o ***download*** e sottoporla al server mediante un pulsante di "Submit query" (vedi figura 1, passi (query) e (download)) oppure può ripristinare i valori di default (utilizzando un pulsante di "reset to default values").

A seguito della richiesta viene svolta sul server una ricerca degli articoli che soddisfano i criteri impostati dall'utente.

Nel caso di una richiesta di tipo ***query*** , il risultato della ricerca viene presentato tramite una pagina HTML dinamica (vedi (3) in figura 1) e viene spedito via e-mail al richiedente ((5), figura 1).

Nel caso di una richiesta di tipo ***download*** , il richiedente deve interagire con una successiva pagina HTML dinamica attraverso la quale può scegliere uno dei metodi con il quale i file da lui scelti verranno codificati prima di venirgli/le spediti via e-mail all'indirizzo da lui/lei specificato al momento di impostare i parametri della richiesta ((6), (7) e (8) di figura 1).

Conclusioni

Il QLL-Repository è al momento implementato e funzionante sebbene necessiti ancora di alcuni aggiustamenti e migliorie fra le quali una migliore gestione dei pattern del nome dei singoli autori e del titolo degli articoli e la realizzazione di alcuni controlli di congruità dei dati immessi dall'utente dal lato web-client, mediante l'uso di un semplice programma in JavaScript.

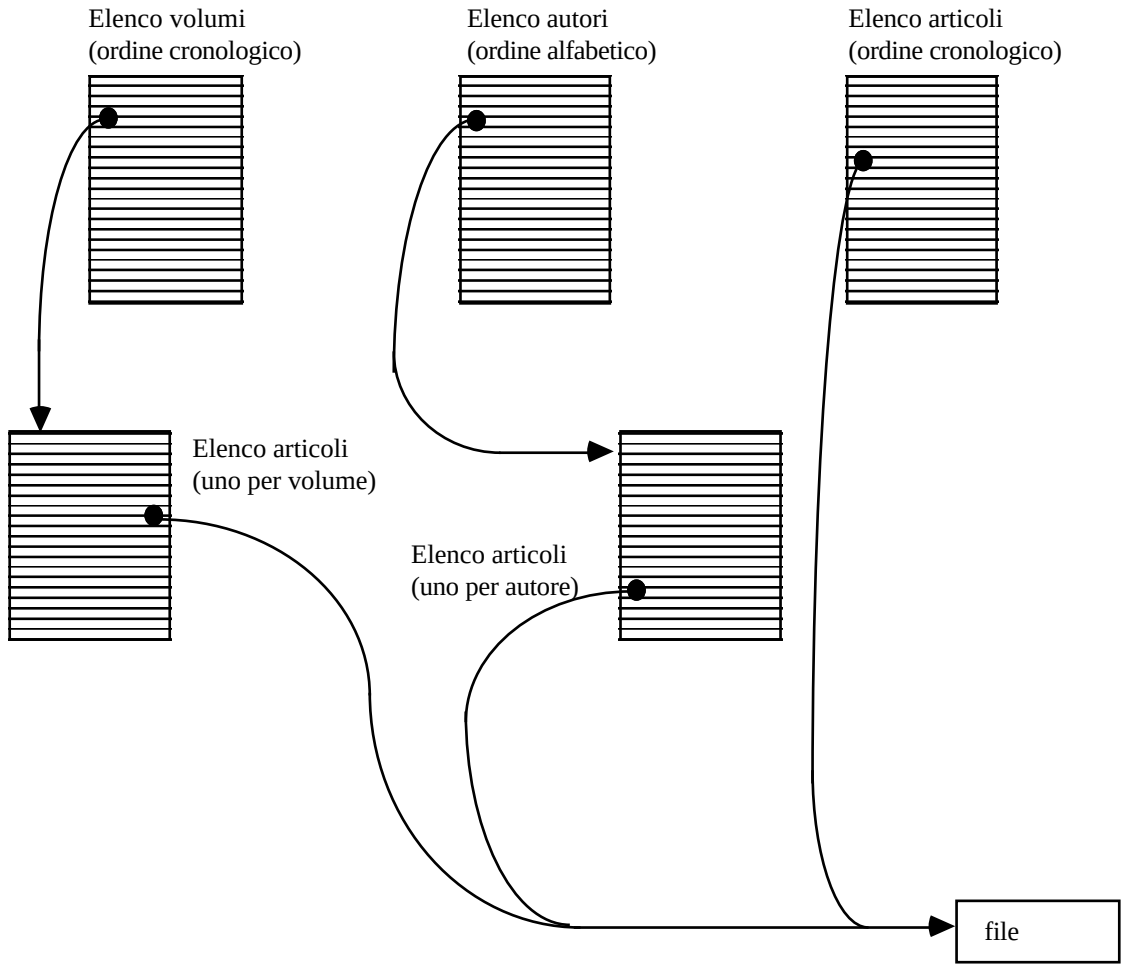


Figura 2 Evoluzione del QLL-Repository: uso di file multipli per il reperimento degli articoli nel data base (spiegazione nel testo)

Un'altra modifica possibile riguarda il miglioramento dell'interrogazione in sé sia per una query sia per un download, miglioramento ottenibile mediante l'introduzione di un insieme di strutture dati aggiuntive (vedi figura 2).

La figura 2 illustra uno dei modi in cui attraverso l'uso di strutture logiche di appoggio, dette in figura file, sia possibile migliorare le interazioni degli utenti per la consultazione del data base senza che ciò abbia ripercussioni sul requisito base di progetto, ovvero che i file siano accessibili in modo diretto attraverso un normale ftp-server, e senza l'introduzione di un vero e proprio data-base.

Le strutture dati di figura 2 hanno due scopi:

- permettere di definire dinamicamente il contenuto dei menu a tendina della Pagina di ingresso di figura 1 relativamente ai volumi disponibili e agli autori e
- consentire una mappatura più significativa e semplice fra i vari articoli e i file corrispondenti consentendo anche l'uso di operatori logici.

Si fa notare come in tale struttura il cosiddetto file "Elenco autori" consentirebbe di presentare all'utente il nome per esteso mappandolo, in modo del tutto trasparente, nella stringa corrispondente utilizzata nel nome dei file che corrispondono agli articoli che hanno quella persona fra gli autori. Lo stesso dicasi per i file "Elenco articoli".

Bibliografia

Arthur L. J. & T. Burns, *Shell Unix. Guida alla programmazione*, Mc Graw Hill, 1998

December J. & M. Ginsburg, *HTML & CGI Unleashed*, Sams.net, 1995

Savola T., A. Westenbroek & J. Heck, *Using HTML*, QUE Corporations, 1995