

Documenti Istat

Una Banca dati congiunturale in tecnologia WEB

(*) Francesco Rizzo

Sommario

Già da qualche anni l'ISTAT propone la banca dati Conlstat su CDROM. Tale banca dati è realizzata in architettura client-server a due livelli (two-tier) in ambiente Windows.

L'evolversi della tecnologia legata ad Internet, la disponibilità di strumenti di sviluppo in ambiente WEB sempre più sofisticati, efficienti ed economici, la disponibilità di hardware ad alte prestazioni e a basso costo, ha portato a riprogettare tale banca in ambiente Web.

L'interfaccia utente e le funzionalità lato server sono state accuratamente progettate arrivando al giusto compromesso tra ciò che la versione su CDROM allo stato attuale offre, e quello che è possibile offrire nel momento in cui la banca dati diventa on-line.

Nel progettare il sistema in ambiente Web sono state prese in considerazione alcune caratteristiche che non hanno molta importanza in ambiente stand-alone ed off-line e, quindi, non considerate nella progettazione di Conlstat in versione CDROM, quali gestione dell'utenza, gestione della concorrenza, utilizzo di un protocollo stateless quale l'HTTP.

Il problema di fondo, che è stato alla base dell'intero progetto, è stato quello di adattare una tecnologia nata per utilizzare documenti organizzati in ipertesti, alla ricerca di informazioni in una banca dati che è, per sua natura, organizzata in forma strutturata.

Abstract

For several years, ISTAT (Italian Central Institute of Statistics) has proposed the Conlstat databank on a CD-ROM. This databank is implemented with a two-tier client-server architecture operating under Windows.

The development of the technology tied to the Internet, as well as the availability of increasingly more sophisticated, efficient and cost-effective Web development tools and high-performance and low-cost hardware, has led the redesigning of this data bank in a Web environment.

The user interface and the server-side functions have been designed carefully to arrive at the right compromise between what the CD-ROM version currently offers and what can be offered at the time the databank goes on-line.

In designing the system in a Web environment, due consideration was given to several characteristics that are not very important in stand-alone and off-line environments and that were thus not considered in Conlstat's design for the CD-ROM version, such as user management, management of concurrency and the use of a stateless protocol such as HTTP.

The basic problem underlying the entire project has involved taking a technology created to use documents organized as hypertexts and adapting it to seeking information in a databank that, by nature, is organized in a structured form.

1. Introduzione

Una delle esigenze storicamente sentite dagli utilizzatori di informazioni in formato elettronico, è sempre stata quella di poter accedere in tempo reale ai dati che compongono tali informazioni direttamente, senza dover utilizzare supporti magnetici (dischi e nastri) come mezzi per trasferire tali dati dalla sorgente alla destinazione.

Le banche dati nascono proprio per risolvere questi problemi garantendo all'utente l'accesso all'informazione direttamente dal sistema informativo sorgente senza coinvolgere il sistema informativo di destinazione (i dati ricevuti attraverso un supporto magnetico, devono essere poi organizzati, sul sistema informativo di destinazione, in informazioni leggibili, attraverso procedure informatiche più o meno complesse).

Gli anni '70 e '80 caratterizzati da architetture informatiche a logica concentrata (mainframe e minicomputer) hanno visto il proliferare di banche dati basate su tale tecnologia; la prima metà degli anni '90 con lo sviluppo dell'informatica distribuita: reti locali (LAN) e geografiche (WAN), protocolli di comunicazione standard (TCP/IP) e logica di partizionamento del livello applicativo (client-server), ha visto un nuovo modo di progettare e realizzare le banche dati. Infine l'introduzione e il consolidamento della rete Internet, come globalizzazione dell'informazione, sta ulteriormente modificando il modo a cui accedere alle informazioni, dovunque esse siano dislocate.

Comunque, qualunque organizzazione che intenda fornire informazioni ad un utenza esterna, dovrà necessariamente tenere conto delle esigenze di mercato, adeguando i propri sistemi informativi per aumentare, attraverso un miglioramento dei servizi offerti, il vantaggio competitivo, diminuire i costi e ottimizzare gli investimenti.

Questo documento espone la realizzazione di banca dati basata su tecnologia WEB (ConIstat-on-line) con il chiaro intento di essere un punto di riferimento per tutte quelle esigenze che su tale tecnologia dovranno basare il trasferimento delle loro informazioni.

Il prototipo rappresenta una evoluzione di una banca dati (ConIstat: banca dati congiunturale dell'ISTAT) off-line¹ verso una banca dati on-line². La tecnologia utilizzata è quella tipica della rete Internet (WEB) che nella forma più avanzata rappresenta un

¹ Nel documento si intende per banca dati off-line quella ottenuta da un procedimento batch di trasferimento dei dati dal sistema informativo **sorgente** ad un sistema informativo **destinazione** e, quindi, ad una navigazione degli stessi sul sistema informativo destinazione.

² Per banca dati on-line si intende quella direttamente accessibile nel sistema informativo sorgente, in cui la navigazione avvenga su quest'ultimo, utilizzando il sistema informativo destinazione solo come strumento di presentazione dei dati ricercati.

architettura client-server a 3 livelli. In particolare nel documento viene dato risalto al livello Middleware e al livello Client³

³ Una trattazione approfondita del livello Server è effettuata nel DOCUMENTO ISTAT 11/1998: **ConIstat - sistema di gestione, memorizzazione e presentazione dei dati congiunturali – Rizzo 1997.**

2. Premessa

La realizzazione “ex-novo” o la migrazione delle proprie banche dati in tecnologia WEB, sarà un passo che tutte le organizzazioni che hanno relazioni con un pubblico esterno dovranno, subito o in un futuro prossimo, affrontare. Questo non solo perché la diffusione di tale tecnologia risulta capillare sia a livello di privato che di azienda, ma anche perché, per quanto riguarda le pubbliche amministrazioni, il regolamento “informatico e la firma digitale”, predisposto dall’AIPA⁴ nell’articolo 21 dice:

Entro il 31 dicembre 1998 le pubbliche amministrazioni dispongono per la tenuta del protocollo amministrativo e per la gestione dei documenti con procedura informatica al fine di consentire il reperimento immediato, la disponibilità degli atti archiviati e l’accesso ai documenti amministrativi per via telematica tra pubbliche amministrazioni e tra queste ed i soggetti privati aventi diritto.

L’espressione “per via telematica” alla luce del progetto della rete unitaria della pubblica amministrazione (RUPA), significa appunto Internet, dal momento che l’AIPA ha disegnato un sistema fondato sui protocolli TCP/IP aperto verso Internet.

Gli argomenti che verranno trattati nei prossimi capitoli, analizzano le varie tecnologie su cui si fondano le banche dati passando quindi ad approfondire la tecnologia WEB e le tecniche di scripting lato server, con particolare riguardo allo strumento offerto dalla Microsoft con il nome di Active Server Pages. Verranno inoltre spiegate le problematiche incontrate durante lo sviluppo e i vari modi di risolverle.

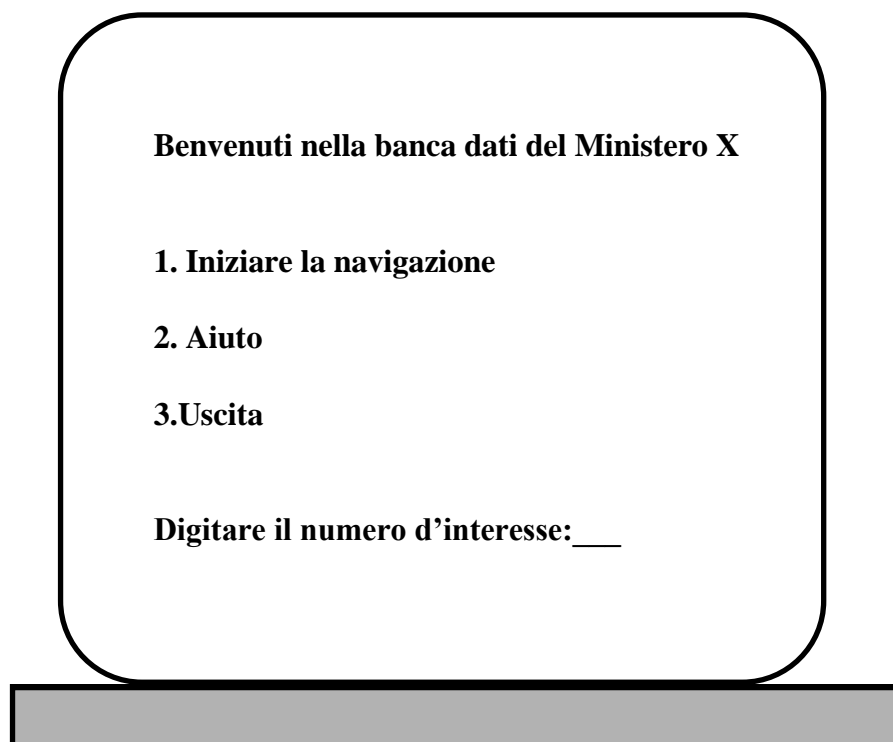
⁴ Autorità per l’Informatica nella Pubblica Amministrazione

3. Architetture delle banche dati

Per banche dati intendiamo sistemi informatici capaci di organizzare e memorizzare dati concernenti una determinata problematica e renderli disponibili sotto forma di informazione strutturata, ad un'utenza sia locale che remota.

3.1 Banche dati sviluppate su mainframe e minicomputer

Tale banche dati sono tipiche di architettura a logica concentrata in cui un Host viene acceduto da terminali "stupidi" o da personal computer in emulazione terminale. In generale sia il data base contenente i dati, sia la logica d'interrogazione e visualizzazione vengono elaborati sull'Host e i risultati inviati al terminale che ha il solo compito di rendere i bit ricevuti in una forma visibile sul monitor. La presentazione dei dati è tipicamente a carattere e si interagisce con l'Host quasi esclusivamente con la tastiera del terminale. Il colloquio tra l'utente e la banca dati avviene quasi sempre digitando dei numeri che rappresentano delle opzioni per navigare nella struttura ad albero della banca dati.

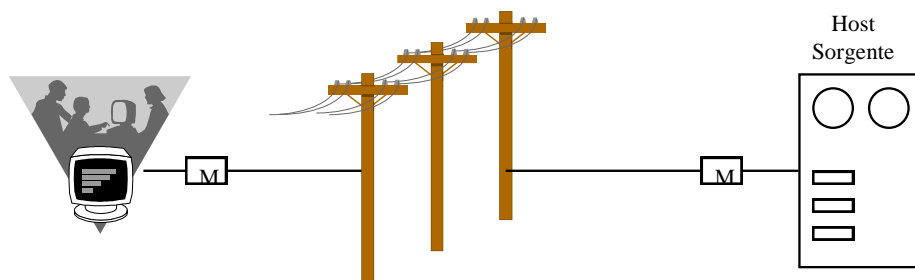


Collegarsi a queste banche dati presuppone quasi sempre un collegamento in emulazione terminale (3270 o Telnet) all'Host. Le situazioni possibili possono essere:

a) Collegamento utente singolo:

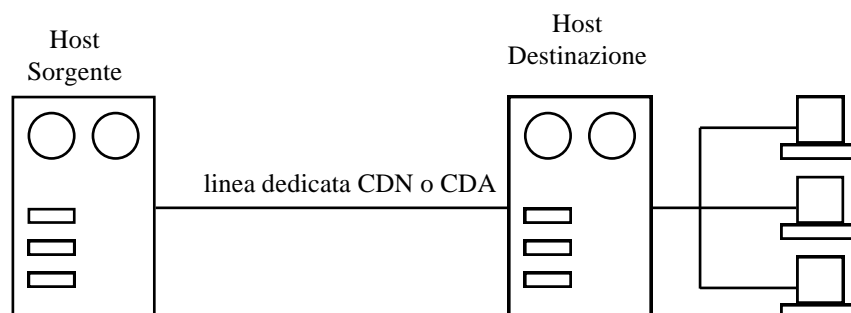
l'utente attraverso un modem e un software di emulazione terminale si collega all'Host utilizzando la rete telefonica commutata. Dopo una prima fase di autenticazione si

accede alla banca dati attraverso una sessione di lavoro che viene mantenuta per tutta la durata del collegamento. Evidentemente questo tipo di collegamento risulta poco ottimizzato sia per quanto riguarda i costi (la rete telefonica commutata è soggetta ad una tariffazione che è funzione sia del tempo che della distanza) sia per l'utilizzo delle risorse sull'Host (se l'host è predisposto per accettare un certo numero di sessioni, nel momento in cui sono tutte attive, un ulteriore utente dovrà attendere il rilascio di qualcuna prima di potersi collegare)



b) Collegamento attraverso un sistema informativo destinazione

viene attivato un collegamento permanente tra due Host (quello sorgente che contiene la banca dati e quello destinazione che permette a più terminali di collegarsi contemporaneamente attraverso un singolo collegamento fisico). Un qualunque utente potrà accedere alla banca dati passando attraverso un Host del proprio sistema informativo. Questo tipo di collegamento risulta anch'esso poco ottimizzato per ciò che riguarda i costi (nel caso di un elevato numero di terminali che si collegano per un tempo sufficientemente lungo l'utilizzo di una linea dedicata risulta più ottimizzata rispetto a un collegamento attraverso linea commutata) e l'utilizzo delle risorse dell'Host.



3.2 Banche dati in architettura client server tradizionale

Il down-sizing dei sistemi mainframe-like a sistemi distribuiti attraverso l'utilizzo di reti locali (LAN), personal computer e server SQL, ha portato alla nascita dell'architettura Client-Server intesa come:

insieme di processi e sistemi che si suddividono il compito di rendere efficiente, flessibile ed economica la memorizzazione e l'elaborazione dei dati in una architettura distribuita.

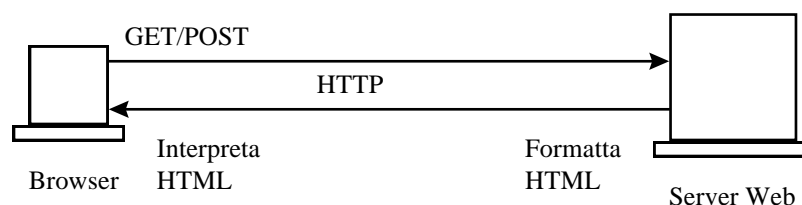
A livello applicativo il client-server descrive accuratamente una applicazione in cui due o più processi logici separati, lavorano insieme per eseguire un compito. Il processo client richiede al server di eseguire un lavoro per suo conto. I processi di una applicazione client-server possono essere eseguite o meno sulla medesima macchina fisica: potrebbero trovarsi su un singolo elaboratore oppure separati da migliaia di chilometri di cavo telefonico.

4. La tecnologia WEB come standard per realizzare Banche Dati

Il World Wide Web (WWW), altrimenti detto più semplicemente WEB, è una iniziativa partita dal CERN di Ginevra con lo scopo di fornire e ricercare informazioni sulla rete Internet. La novità è data nell'organizzare i documenti in forma ipertestuale⁵ che consentono una gestione dell'informazione più agevole sia dal lato della fornitura che dell'utenza.

L'iniziativa del WWW, oltre a costituire la W3 Organization con l'incarico di definire future evoluzioni, ha progettato sia il protocollo di trasmissione (HTTP) che il linguaggio ipertestuale (HTML).

In generale volendo schematizzare un ambiente WWW si può pensare alla seguente architettura:



⁵ Gli ipertesti sono documenti che rispondono a determinati requisiti di costruzione strutturata e che vengono creati in modo da offrire un collegamento tra un punto del testo e altri documenti o altri punti dello stesso documento. In altre parole, invece di leggere un normale testo elettronico come una struttura rigida, lineare e sequenziale, negli ipertesti si può saltare facilmente da un punto ad un altro, cercare informazioni, tornare indietro su un argomento già visto, ecc.; il tutto con la facilità e l'immediatezza del click del mouse, in base alle decisioni del momento e secondo ciò che sembra più interessante.

Il Browser, programma client, richiede (GET/POST) le informazioni posizionate sul server identificandole attraverso un indirizzo (URL⁶). Il colloquio tra il client e il server avviene attraverso il protocollo HTTP. Il server invia al client richiedente i documenti richiesti formattati secondo le specifiche HTML. Il client interpreta i marcatori HTML e presenta all'utente un'informazione leggibile.

4.1 Il linguaggio HTML

L'HTML (HyperText Markup Language) è un linguaggio di formattazione di documenti in ambiente WWW derivato dall'SGML (Standard Graphical Markup Language). Rappresenta un insieme di specifiche che permettono di creare simboli grafici in base a un sistema di marcatori che accompagnano le informazioni utili. E' possibile così centrare una frase, facendo precedere la frase stessa dal marcatore <CENTER>, e facendola seguire dal marcatore </CENTER>:

<CENTER> frase che si vuole visualizzare </CENTER>

Oppure rendere una frase o una immagine un iperlink:

clicca qui per visualizzare amerigo

Un punto chiave relativo allo sviluppo in HTML che è importante capire, è che oltre poter specificare gli attributi di formattazione, si possono definire i collegamenti tra varie pagine permettendo di impostare un percorso tra argomenti correlati.

Uno dei benefici dell'HTML è che non essendo un linguaggio di programmazione, ma un file testo, risulta indipendente dalla piattaforma che si va ad utilizzare: qualsiasi applicazione con qualsiasi sistema operativo che sia in grado di interpretare documenti formattati HTML può essere usata.

⁶ Uniform Resource Locator

4.2 Il protocollo HTTP- Hyper Text Transfer Protocol

E' il protocollo utilizzato nel trasferimento degli ipertesti.

I suoi scopi principali sono:

- Ridurre al minimo indispensabile gli accessi alla rete e migliorare i tempi di risposta.
Per far ciò viene costruita una cache locale, cioè un deposito ove vengono memorizzate le ultime pagine accedute in modo da evitare ulteriori trasferimenti. Le pagine vengono trasferite solo se nuove o aggiornate.
- Ridurre il carico di rete.
Con il WEB la maggior parte del tempo è dedicata alla lettura: prima di decidere quale altra pagina desideriamo caricare, dobbiamo leggere quella corrente. Per tutto questo tempo è inutile mantenere attiva una connessione di rete. Per questo motivo le connessioni HTTP sono ridotte al minimo indispensabile: una volta terminato il trasferimento la connessione viene chiusa, consentendo così una riduzione del carico di rete.

HTTP si basa sul modello client-server dove esiste una entità server in grado di fornire dati ai client che ne facciano esplicita richiesta. In questo caso il dialogo è sempre iniziato da un client che si rivolge ad un server per ottenere dei servizi, mai viceversa.

Inoltre il protocollo HTTP prevede un dialogo half-duplex, ossia che i partecipanti al collegamento (client-server) parlino sempre uno per volta, mai contemporaneamente.

Per identificare il server che si desidera contattare, un client usa l'URL del server, cioè il suo indirizzo di rete, che lo identifica univocamente. Un URL si compone nel seguente modo:

protocollo://indirizzo:porta/pathname

Il protocollo indica quale tipo di servizio si desidera dal server; infatti è possibile che lo stesso server possa essere contattato non solo tramite HTTP ma anche con altri protocolli: ciò che viene trasferito e visualizzato dipende dal protocollo impiegato. Il campo indirizzo contiene l'indirizzo di rete del server (secondo lo schema del TCP/IP) mentre la porta, opzionale, specifica verso quale porta va effettuata la connessione; se non viene specificata assume un valore di default dipendente dal protocollo (es. 80 per HTTP). Pathname specifica il percorso assoluto del file all'interno del server; il valore di default di questo campo è "/" cioè la directory root del server.

5. Vantaggi della tecnologia WEB nella realizzazione di Banche Dati.

I vantaggi che derivano dall'impiego della tecnologia WEB nella realizzazione di Banche Dati sono sia il miglioramento del servizio verso gli utenti, sia il miglioramento implementativo.

Per ciò che riguarda il miglioramento del servizio verso gli utenti sono da prendere in considerazione i seguenti punti:

- le Banche Dati sono facilmente raggiungibili da chiunque si connette alla rete Internet. Questo significa che per l'utente diventa economico collegarsi in quanto non deve attivare una comunicazione, nella maggior parte dei casi, "interurbana", ma arriva a destinazione attraverso un Internet Service Provider che generalmente è localizzato dentro lo stesso distretto telefonico
- l'utente non necessita di particolare software per accedere alla banca dati, ma è sufficiente un qualunque browser che spesso viene fornito gratuitamente
- facilità di navigazione nella banca dati. In ambiente WEB è possibile strutturare le informazioni secondo l'efficace sistema delle pagine e dei collegamenti ipertestuali che permettono, anche a non esperti, una veloce e facile ricerca dell'informazione.

Per ciò che riguarda il miglioramento implementativo sono da considerare i seguenti punti:

- tutti i produttori di software a livello mondiale stanno impiegando grosse risorse nello sviluppo di strumenti e oggetti legati al WEB; è logico dunque aspettarsi, in un breve tempo, disponibilità, a basso costo e buon livello tecnologico, di software e know-how
- diminuzione dei costi di gestione dell'applicazione. Nell'architettura client-server tradizionale, i programmi eseguibili ed i file ad essi correlati devono essere installati su ciascun client. Questo comporta ovviamente un'assistenza in fase di installazione. Inoltre eventuali aggiornamenti devono essere installati nuovamente su ciascun client. La tecnologia WEB necessita soltanto del browser sul client.
- portabilità su piattaforme diverse. Le differenze operative e visive tra i diversi ambienti (sistemi operativi) impongono, nella maggior parte dei casi la modifica delle applicazioni in funzione di ogni piattaforma da supportare, con conseguenti costi di sviluppo e manutenzione. Nel caso delle applicazioni WEB, l'interpretazione dei programmi avviene attraverso il browser indipendentemente dal tipo di elaboratore su cui lavora. La portabilità permette quindi agli sviluppatori di concentrarsi sulla creazione delle applicazioni senza dover pensare al porting dello stesso applicativo su piattaforme diverse.

6. La banca dati ConIstat-on-line

Una delle esigenze storicamente sentite dagli utilizzatori di dati congiunturali è sempre stata quella di poter prendere visione dei dati nel momento in cui sono disponibili. Questo perché, per loro natura, questo tipo d'informazione (congiunturale), ha il suo significato maggiore nel breve periodo e risulta, quindi, obsoleta in un lasso di tempo minimo. Inoltre questi dati hanno valenza se vengono confrontati con quelli di periodi precedenti, vedendo il loro andamento nel tempo, o studiando le variazioni che possono aver luogo: mese di riferimento sul mese precedente nello stesso anno (variazione congiunturale); mese di riferimento sullo stesso mese dell'anno precedente (variazione tendenziale).

Già da qualche anno l'ISTAT propone la banca dati ConIstat su CDROM. Tale banca dati realizzata in architettura client-server a due livelli (two-tier) in ambiente Windows: l'utente deve installare sul proprio computer sia il database che l'applicativo che permette di ricercare i dati d'interesse. A cadenza periodica (secondo il calendario dei comunicati stampa), è possibile fare il download degli aggiornamenti, collegandosi, attraverso la rete Internet, all'indirizzo bbs.istat.it. Quest'ultima operazione, però, richiede del tempo e, in ogni caso, anche se l'utente è interessato solo ad alcune serie storiche, deve necessariamente aggiornare l'intera banca dati.

L'evolversi della tecnologia legata ad Internet, la disponibilità di strumenti di sviluppo in ambiente WEB sempre più sofisticati, efficienti ed economici, la disponibilità di hardware ad alte prestazioni e a basso costo, ha portato a riprogettare tale banca in ambiente Web. Nello studio delle specifiche applicative si è constatato che molte funzioni già presenti nella versione su CDROM sono state largamente apprezzate dagli utenti e, quindi, compatibilmente con ciò che la tecnologia WEB permette, tali funzioni sono state mantenute: l'interfaccia utente e le funzionalità lato server sono state accuratamente progettate arrivando al giusto compromesso tra ciò che la versione su CDROM allo stato attuale offre, e quello che è possibile offrire nel momento in cui la banca dati diventa on-line.

Nel progettare il sistema in ambiente Web sono state prese in considerazione alcune caratteristiche che non hanno molta importanza in ambiente stand-alone ed off-line e, quindi, non considerate nella progettazione di ConIstat in versione CDROM, quali gestione dell'utenza, gestione della concorrenza, utilizzo di un protocollo stateless quale l'HTTP.

Il problema di fondo, che è stato alla base dell'intero progetto, è stato quello di adattare una tecnologia nata per utilizzare documenti organizzati in ipertesti, alla ricerca di informazioni in una banca dati che è, per sua natura, organizzata in forma strutturata.

La soluzione è stata quella di organizzare le ricerche all'interno del database attraverso un "albero di ricerca", con pochi rami, in maniera tale che l'utente possa arrivare all'informazione d'interesse soltanto dopo aver visitato poche pagine.

Un modo molto conveniente di pensare all'applicazione è stato quello di suddividerla nelle diverse funzioni che si intende poter svolgere: molte applicazioni client-server sono costruite in base al modello di elaborazione transazionale. Questo consente nel suddividere le funzionalità in transazioni discrete e gestirli dall'inizio alla fine.

Nella tecnologia WEB è molto utile pensare alle varie pagine come a un singolo gruppo di transazioni. L'unità di lavoro che verrà svolta da ogni pagina, che sia richiesta di informazioni o l'autentica delle operazioni effettuate sui dati inviati, può essere considerata una transazione separata. Adottando questo modello è semplice mappare queste transazioni basate su documenti rispetto alla propria base dei dati.

Nella progettazione del sistema sono state prese in considerazione:

- le specifiche funzionali;
- le specifiche tecnologiche.

Tale fase progettuale è stata supportata dalla realizzazione di vari prototipi che hanno accertato la fattibilità del sistema e la sua utilizzabilità

6.1 Specifiche funzionali

Le principali funzioni che un utente può svolgere utilizzando il navigatore sono:

- autenticarsi mediante l'inserimento di una Userid e una Password; ovvero registrarsi attraverso la compilazione di un modulo on-line. In tal caso riceverà immediatamente, attraverso E-mail, la Userid e Password;
- ricercare le informazioni d'interesse attraverso il codice Ateco o per Dominio di aggregazione;
- mettere a confronto, quindi, richiedere per la visualizzazione serie storiche appartenenti ad Ateco differenti o ad Aggregati differenti;
- estrarre i dati delle serie storiche selezionate attraverso la scelta dell'intervallo temporale idoneo (periodo iniziale - anno iniziale, periodo finale - anno finale);
- gestire contemporaneamente serie storiche a frequenza diversa;
- avere la piena visibilità dei metadati;
- gestire sia serie storiche grezze che destagionalizzate;
- esportare sul proprio computer le serie storiche estratte in formato testo (.txt) e in formato .csv;

- salvare uno o più percorsi di ricerca in maniera tale da poterli richiamare in seguito in altre sessioni di lavoro.

Tutte le funzionalità sopra esposte, sono state implementate attraverso un sistema di ricerca (navigazione) progettato per essere logico ed intuitivo, permettendo anche ad utenti non esperti di utilizzare il sistema. L'applicazione che permette la navigazione, presenta un unico frame suddiviso in tre zone:

- la **testata** in cui sono presenti dei link a dei sistemi di help;
- il **corpo** con il quale l'utente interagisce durante la navigazione;
- il **piè di pagina** che dà all'utente informazioni di referenze.



- [Ricerca per Aggregato](#)
- [Ricerca per Ateco](#)

Vuoi richiamare una ricerca già memorizzata? [SI](#)

[E-mail](#)

(Risoluzione grafica ottimale - 800x600)

Ciascun utente nel momento in cui richiede, attraverso un browser, la visualizzazione della home-page della banca dati, apre una sessione di lavoro. Attraverso la sessione è possibile effettuare tutta una serie di operazioni quale, per esempio, quella di depositare, prima di estrarre i dati, nel cosiddetto "carrello della spesa" le serie storiche d'interesse.

Le sessioni sono caratterizzate dai seguenti parametri:

- Time-out: nel caso l'utente non interagisce con la banca dati per un periodo superiore a quello precisato nel time-out, la sessione viene chiusa automaticamente. Tale parametro è settato al valore 5 minuti.
- Numero massimo di serie depositabili nel "carrello della spesa". Tale parametro, settato a 20, dà un limite al numero di serie che sono depositabili nel carrello della spesa. Nel caso l'utente vuole selezionare altre serie dopo aver raggiunto tale limite, deve prima togliere dal carrello della spesa alcune serie.

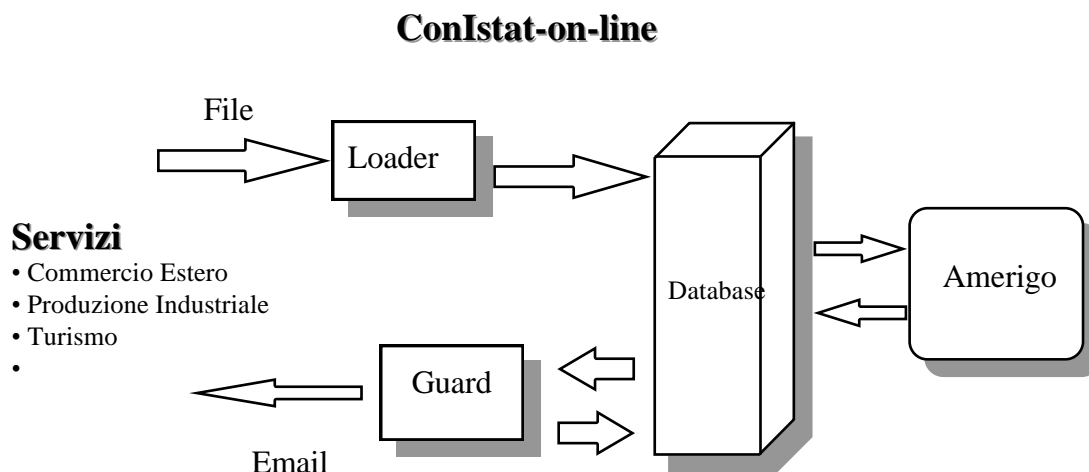
- Numero massimo di serie estraibili. Tale parametro, settato a 5, specifica il numero di serie per le quali è possibile estrarre i dati contemporaneamente. Una volta che l'utente ha depositato le serie nel carrello della spesa, potrà estrarre i dati solo per 5 serie per volta.
- Numero di ricerche memorizzabili. L'utente dopo aver depositato un certo numero di serie nel carrello della spesa, può salvarle come un'unica ricerca. E' possibile salvare fino a 5 ricerche. Tale operazione permetterà all'utente, in un sessione futura, di richiamare le ricerche salvate e, quindi, non essere costretto ad effettuare delle navigazioni all'interno della banca dati per arrivare ai dati d'interesse.

6.2 Specifiche tecnologiche

Il sistema si compone di alcuni applicativi e di un DBMS.

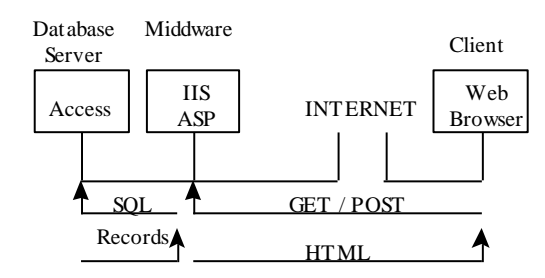
I Servizi addetti alla produzione dei dati, preparano dei file secondo delle specifiche ben definite. Tali file vengono depositati sul Server in cui risiede il database. Giornalmente il programma "Guard" interroga una tabella del data base in cui è memorizzato il calendario dei "comunicati stampa". Nel momento in cui vengono riscontrate date, all'interno del calendario del comunicato stampa, già trascorse rispetto al giorno attuale, e che non sono state interessate da operazione di caricamento dati, viene inviato, in automatico, un Email ai responsabili dei Servizi che avrebbero già dovuto inviare i file per quelle date.

Ad ogni comunicato stampa, un operatore manualmente (in futuro verrà implementata una procedura automatica) lancia il programma "Loader" che carica i nuovi dati nel data base. L'utente attraverso un qualunque browser si interfaccia alla banca dati utilizzando "Amerigo", un software che permette la navigazione tra le serie storiche.



L'architettura su cui il sistema si basa è un client-server a tre livelli in ambiente WEB così strutturato:

- **client**: un Web Browser;
- **middleware**: un Web Server ed un Application Server;
- **Server**: un Data Base Server



Il Client mediante il metodo Post o Get del protocollo HTTP fa delle richieste al Web Server.

Il Server Web analizza le richieste, ed attraverso l'Application Server, interpreta gli script che vengono richiamati dalle richieste dal browser.

Nel caso che gli script contengono delle richieste circa le informazioni contenute nel database, vengono lanciate delle query SQL al database stesso.

Il data base restituisce all'Application Server i Records richiesti.

Il Web Server formatta i risultati in HTML e li rispedisce al client. Il Browser li rende disponibile all'utente in una forma leggibile.

Il sistema è stato pensato per funzionare con tutti i database che si interfacciano tramite drive ODBC, per cui se in futuro si penserà di convertire tale data base in ambiente Oracle, i cambiamenti da attuare al sistema saranno marginali e consisteranno:

- nell'adeguata configurazione di SQLNet e del drive ODBC;
- nell'impostazione dell'opportuna stringa di connessione (attualmente posizionata nel file Global.asa);
- nella revisione di alcune query SQL.

6.3 Specifiche Hardware e Software

Nel sistema viene utilizzato il Web Server Microsoft Internet Information Server 4.0 con Windows NT Server 4.0. L'Application Server è Active Server Pages (ASP).

L'applicazione è scritta in maniera tale che qualunque browser che rispetti le specifiche HTML 3.2, possa usarla correttamente. Nel caso il browser è abilitato per supportare Javascript, l'interazione con il server risulta ottimizzata: esistono alcune routine scritte in Javascript che validano le azioni effettuate dall'utente (per esempio la scelta dell'intervallo temporale, ecc.). Tale routine vengono eseguite lato client con i conseguenti vantaggi circa la diminuzione dei tempi di risposta.

Affinché l'applicativo funzioni correttamente, l'accettazione dei cookies deve essere abilitata: la gestione della sessione di ciascun utente viene effettuata automaticamente da ASP mediante la scrittura di un cookie sul client.

Il data base utilizzato è Access 7.0 con motore SQL JET 3.5. Il sistema è pensato comunque per funzionare con tutti i data base ODBC-compliant, in particolare è stato testato con Oracle 8.05.

Per ottimizzare i costi sia il data base che il middleware sono posizionati sullo stesso computer: un Pentium Pro a 300 Mhz, 256 MB di Ram.

6.4 Architettura applicativa

Le funzioni del sistema sono state implementate tutte lato server. Però per ottimizzare i tempi di risposta, per quei browser che sono abilitati ad utilizzare Javascript, alcune di tali funzioni sono state implementate anche lato client. In base al tipo di funzione, nel sistema vengono distinte:

- funzioni di *interrogazione* dal data base e di *presentazione* dei dati
tali funzioni sono state tutte sviluppate lato server;
- funzioni di *validazione* delle operazioni che l'utente compie
tali funzioni sono state realizzate in JAVASCRIPT lato client. Per garantire, però, la compatibilità con i browser che non sopportano JavaScript, le stesse funzioni sono state sviluppate anche lato server.

6.4.1 Interrogazione del data base

Tale funzione è realizzata utilizzando gli oggetti della famiglia ADO (ActiveX Data object) che implementa a più alto livello il protocollo OLEDB. In particolare gli oggetti che sono stati principalmente utilizzati sono Connection e Recordset.

Ciascuna pagina che realizza una interrogazione verso il database contiene il seguente codice:

```
SET Conn = Server.CreateObject("ADODB.Connection")
Conn.Open Application("Connectionstring")
SQL="Select ....."
SET RS = Conn.Execute(SQL)
Do While Not RS.eof
    .....
    RS.Movenext
Loop
RS.Close
SET RS = Nothing
```

Application("Connectionstring") è una variabile a livello di Applicazione, definita nel file *Global.asa*, il cui contenuto è: "dsn=conistat;uid=admin;pwd=32517a".

Dsn=conistat è la fonte dati definita con ODBC Manager che si trova nel "Pannello di controllo" di windows.

6.4.2 Passaggio di variabili tra le pagine

Tra una pagina e la successiva vengono passate delle variabili utilizzando o il metodo GET o il metodo POST. Queste variabili vengono lette attraverso il costrutto REQUEST.QUERYSTRING("nome_variabile"), se viene utilizzato il metodo GET, o il costrutto REQUEST.FORM("nome_variabile"), se viene utilizzato il metodo POST, quindi vengono memorizzate in variabili di sessione.

Ricerca per Aggregato:

Pagina mittente	Variabile trasmessa	Metodo	Pagina destinazione	Variabile di sessione
InizioAggreg.asp	dom	GET	domAG.asp	session("dom")=descriz session("domcod")=cod
domAG.asp	sdom	GET	s_domAG.asp	session("sdom")=descriz session("sdomcod")=cod
s_domAG.asp	ssdom	GET	s_s_domAG.asp	session("ssdom")=descriz session("ssdomcod")=cod
s_s_domAG.asp	SelAggreg.asp	POST	nomericerca serie	session("iCount")=iCount session("serie")=Vett

Ricerca per Ateco:

Pagina mittente	Variabile trasmessa	Metodo	Pagina destinazione	Variabile di sessione
InizioAteco.asp		GET	maAteco.asp	
maAteco.asp		GET	Ateco.asp	
Ateco.asp		POST	SelAteco.asp	

6.4.3 Gestione del "carrello della spesa"

L'utente può selezionare una data serie, deporla nel carrello della spesa; quindi selezionare un'altra serie, deporla ancora nel carrello della spesa e così fino a quando l'utente non decide di estrarre i dati corrispondenti alle serie storiche depositate nel carrello.

Il numero di serie storiche depositabili nel carrello è fissato dalla variabile Application("MaxNumSerie") (attualmente uguale a 20), mentre una volta depositate queste n serie, è possibile estrarne soltanto un numero fissato dalla variabile Application("max") (attualmente uguale a 6). Entrambe le variabili a livello Application, sono valorizzate nel Global.asa e servono a bilanciare le performance dell'applicativo.

Il carrello della spesa permette all'utente di selezionare più serie, anche provenienti da Aggregati o Ateco diversi. E' possibile quindi confrontare serie che hanno una frequenza diversa: il sistema è in grado di presentare in forma tabellare i valori nelle giuste posizioni a secondo delle frequenze interessate.

L'algoritmo che gestisce il carrello della spesa, è in grado di riconoscere se l'utente ha scelto la stessa serie ed in tale caso non l'aggiunge per una seconda volta.

Il carrello della spesa viene gestito attraverso una matrice nelle pagine SelAggreg.asp e SelAteco.asp.

Il numero di righe della matrice rappresenta il numero di serie che è possibile scegliere.

Il numero di colonne rappresenta codici e descrizioni che caratterizzano una data serie. In particolare:

Vett(iCount, 0)	contiene la seguente stringa: dom-s_dom-s_s_dom-tipo-vs-ateco-freq-u_mis-n_dato-anno_in-
-----------------	---

	periodo_in-anno_fin-periodo_fin-descriz
Vett(iCount, 1)	domdesc=session("dom")
Vett(iCount, 2)	sdomdesc=session("sdom")
Vett(iCount, 3)	ssdomdesc=session("ssdom")
Vett(iCount, 4)	descriz
Vett(iCount, 5)	"1" o "2" (*)

(*) 1 significa che la serie è stata deposita nel carrello della spesa; 2 significa che la serie già deposita nel carrello della spesa è stata successivamente marchiata come eliminata.

6.4.4 Memorizzazione di ricerche effettuate.

L'utente dopo aver eseguito una o più ricerca di serie storiche, e dopo averle depositate nel carrello della spesa, può memorizzarle come unica *entità* specificandone un nome. In questo modo, l'utente, utilizzando il nome può, in un'altra sessione di lavoro, richiamare l'entità d'interesse e quindi le serie che essa rappresenta. Questa operazione presenta il grande vantaggio che se un utente è interessato a ricercare e visualizzare frequentemente solo alcune serie, attraverso la memorizzazione delle entità, evita di dover navigare nelle varie pagine prima di arrivare alle serie d'interesse.

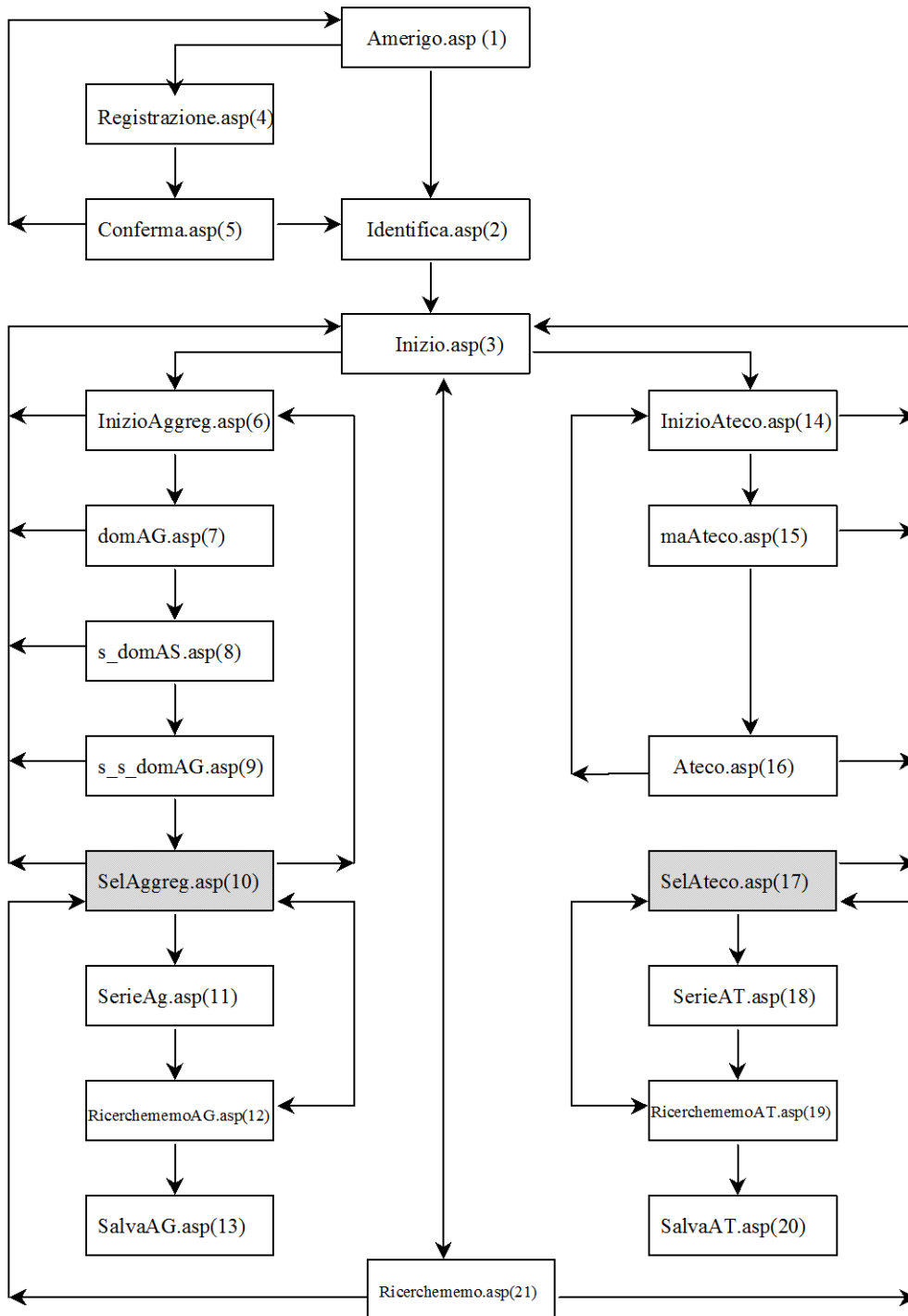
La memorizzazione dell'entità avviene nella tabella RICERCHE del database dbUTENTI.MDB.

E' possibile memorizzare un numero di entità pari alla variabile Application("ricmax"). Tale limitazione viene imposta per limitare lo spazio che ciascun utente occuperebbe.

Il sistema è in grado di riconoscere se l'entità memorizzata è stata realizzata percorrendo l'albero per Ateco o per Aggregato, per cui nel momento in cui viene richiamata dall'utente, le serie che ne fanno parte vengono depositate, automaticamente nel carrello della spesa o della pagina SelAggreg.asp (l'entità rappresenta serie ricercate per Aggregato) o della pagina SelAteco.asp (l'entità rappresenta serie ricercate per Ateco) .

6.5 La mappa delle pagine ASP

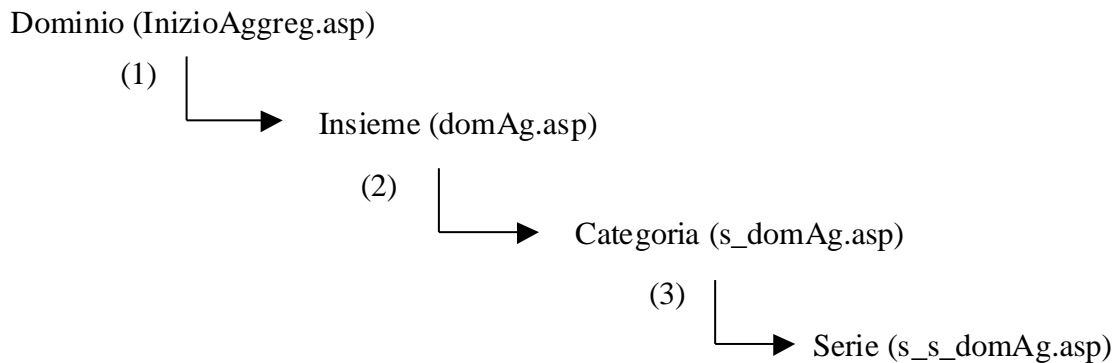
Il sistema si compone di un file d'inizializzazione (global.asa) e di un insieme di pagine, con estensione .asp, collegate fra di loro. La mappa delle pagine è la seguente:



Dallo schema si nota che il sistema è organizzato in una struttura ad albero con la radice rappresentata dalla pagina Inizio.asp e due rami principali: Ateco ed Aggregato.

Ciascun ramo termina rispettivamente nelle pagine SelAteco.asp e SelAggreg.asp che contengono i propri “carrelli della spesa”.

La ricerca attraverso il ramo Aggregato avviene seguendo i seguenti nodi:

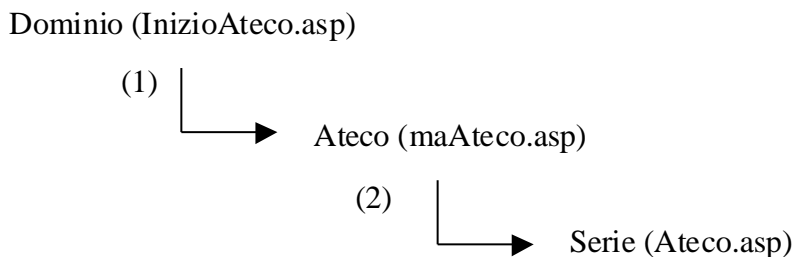


Il nodo (1) memorizza una variabile in Vett(iCount, 1)

Il nodo (2) memorizza una variabile in Vett(iCount, 2)

Il nodo (3) memorizza una variabile in Vett(iCount, 3)

La ricerca attraverso il ramo Ateco avviene seguendo i seguenti nodi:



Il nodo (1) memorizza una variabile in Vett(iCount, 1)

Il nodo (2) memorizza una variabile in Vett(iCount, 3)

Ciascuna pagina presenta una struttura così generalizzabile:

a) testata realizzata tramite un'operazione di include:

- ricerche per Aggregato: <!--#include virtual="/amerigo/logoAG.asp"-->
- ricerche per Ateco: <!--#include virtual="/amerigo/logoAG.asp"-->

b) include di alcune funzioni esterne:

- <!--#include virtual="/amerigo/Funz_Char.asp"-->

c) include dell'help: <!--#include virtual="/amerigo/help.asp"-->

d) corpo della pagina comprendente script ASP e tag HTML

e) piè di pagina realizzato attraverso un include: <!--#include virtual="/amerigo/pie.asp"-->

In seguito verranno illustrate gli algoritmi implementati nelle singole pagine.

6.5.1 Global.asa

Tale pagina non è visibile all'utente ma serve al sistema a compiere alcuni eventi nel momento in cui l'utente avvia una sessione.

SUB Application_OnStart

```
'setto la stringa di connessione ADO verso il database DBASE.MDB
Application("ConnectionString")="DSN=conistat; uid=admin; pwd=28314a"
'Nel caso si vuole la connessione verso Oracle con SID amerigo:
Application("ConnectionString")="driver={Microsoft ODBC for Oracle}; " _
& "server=amerigo; uid=amerigo; pwd=28314a"
'setto un flag a secondo che il database sia Access (2) o Oracle (1)
Application("Flag")= "2"
'setto la stringa di connessione ADO verso il database UTENTE.MDB
Application("ConnectionString2")="DSN=utente; uid=; pwd=28314a"
'gestisco il contatore per il numero di visite effettuate e per il numero di utenti contemporaneamente presenti
Application.Lock
'imposto la variabile che conta il numero di utenti contemporaneamente presenti
Application("Whoson")=0
'Leggo nel file C:\amerigo\librerie\hits.txt il valore corrente del numero di accessi e lo assegno alla variabile
Application("hits")
set fso=createobject("Scripting.FileSystemObject")
IF fso.FileExists("C:\amerigo\librerie\hits.txt") THEN
    set f = fso.getfile("C:\amerigo\librerie\hits.txt")
    set ts = f.OpenAsTextStream(1,0)
    Application("hits")=ts.readline
    ts.close
END IF
set fso = nothing
Application.Unlock
'Numero massimo di serie estraibili
Application("max")=6
'Numero massimo di ricerche massime memorizzabili
Application("ricmax")=5
'dimensione delle righe-massimo numero delle serie nel carrello
Application("MaxNumSerie")=20
'dimensione delle colonne
Application("MaxNumAttributi")=6
END SUB
```

SUB Session_OnStart

```
'Nel momento in cui apre una sessione legge il valore nella variabile Application("hits") e incrementa il contatore di una unità
Application.Lock
Application("hits") = application("hits") + 1
Application("Whoson") = Application("Whoson") + 1
Application.Unlock
'Controllo che unizi correttamente dalla prima pagina
```

```
startpage = "/amerigo.asp"  
currentpage=Request.ServerVariables("SCRIPT_NAME")  
IF strcomp(currentpage, startpage, 1) THEN  
    Response.Redirect ("/amerigo" & startpage)  
END IF
```

END SUB

SUB Session_OnEnd

'Nel momento viene chiusa una sessione utente, vengono cancellati i file da esso creati

```
FileRoot="file" & session.SessionID & ".txt"  
FileName="C:\amerigo\export\" & cstr(FileRoot)  
FileRoot1="file" & session.SessionID & ".csv"  
FileName1="C:\amerigo\export\" & cstr(FileRoot1)  
Set fso = createobject("scripting.filesystemobject")  
IF fso.FileExists(FileName) THEN  
    fso.DeleteFile(FileName)  
END IF  
IF fso.FileExists(FileName1) THEN  
    fso.DeleteFile(FileName1)  
END IF
```

'diminuisco la variabile che rappresenta gli utenti collegati contemporaneamente

```
Application.Lock  
Application("WhosOn")= Application("WhosOn") - 1  
Application.unlock
```

END SUB

SUB Application_OnEnd

'nel momento in cui il webserver viene fermato, registro nel file hits.txt il valore del contatore *'che conta gli accessi*
all'applicativo

```
Application.Lock  
Set fso = createobject("scripting.filesystemobject")  
IF fso.FileExists("C:\amerigo\librerie\hits.txt") THEN  
    set f = fso.getfile("C:\amerigo\librerie\hits.txt")  
    set ts = f.openastextstream(2, 0)  
    ts.write application("Hits")  
    ts.close  
END IF
```

```
set fso = nothing  
Application.unlock
```

END SUB

7. Il Manuale utente

ConIstat-On-Line è una banca dati congiunturale in ambiente Web.

Per poter accedere a tale banca dati, l'utente deve avere l'accesso ai servizi Web di Internet e aver installato sul proprio computer un Browser quali Microsoft Explorer o Netscape Navigator.

Il Browser deve essere conforme alle specifiche HTML 3.2 o superiore e deve essere in grado di visualizzare tabelle e immagini.

L'utente, affinché possa navigare correttamente nella banca dati, deve abilitare il Browser ad **accettare i Cookies** e ad **interpretare JavaScript**.

7.1 Per connettersi.

Per connettersi alla banca scrivere nello spazio riservato agli URL il seguente indirizzo:

<http://con.istat.it/bd/>

Verrà presentata la home page che permette l'inserimento di una UserID e una Password. E' possibile passare dalla versione Italiana alla versione Inglese e viceversa semplicemente cliccando sulla bandiera inglese o italiana.

7.2 Per registrarsi.

Ci accede per la prima volta dovrà registrarsi cliccando sulla scritta "Registrati ora". Dopo qualche minuto riceverà una E-mail nella quale vengono specificati la UserID e la Password. Tali riferimenti non possono essere cambiati, in seguito, dall'utente.

7.3 La sessione utente.

Nel momento in cui l'utente si è identificato, il sistema apre una sessione di lavoro. Tale sessione è specifica per ciascun utente ed è caratterizzata da un **time-out** di 5 minuti, cioè il periodo di tempo che l'utente ha a disposizione, non interagendo con il sistema, prima che venga sconnesso automaticamente.

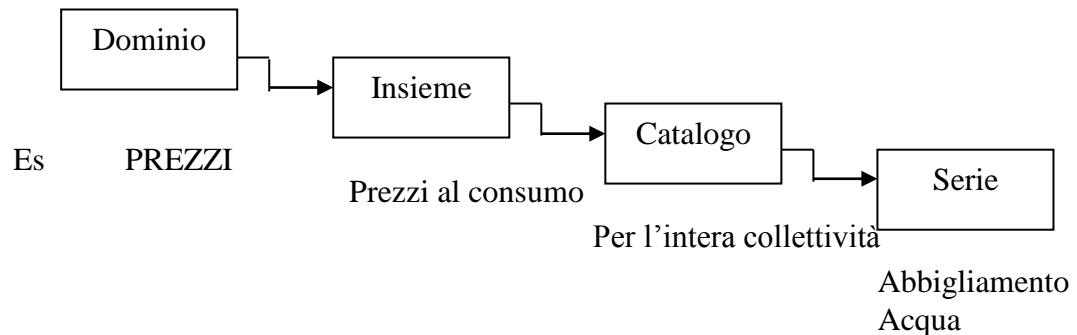
7.4 La ricerca delle serie storiche.

Per ricercare una o più serie storiche d'interesse, l'utente deve percorrere la banca dati seguendo una struttura gerarchica (albero di ricerca) in cui la radice è il punto di partenza e le foglie sono le serie.

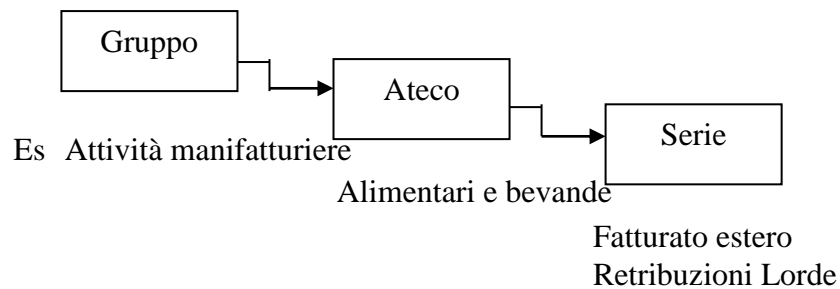
Per iniziare la ricerca è necessario scegliere una modalità. ConIstat-on-line possiede due modalità di ricerca:

- Per Dominio di aggregazione.
- Per codice ATECO.

Per Dominio di aggregazione.



Per codice ATECO.



7.5 Il Carrello della Spesa.

L'utente può selezionare serie appartenenti a più domini di aggregazione o a più gruppi di codice Ateco. Per far ciò utilizza il **Carrello della Spesa** nel quale deposita le serie d'interesse incontrate durante la navigazione.

In pratica l'utente può selezionare delle serie appartenenti ad un dato Dominio, depositarle nel Carrello della Spesa, quindi selezionare serie appartenenti ad un altro Dominio, depositarle nel Carrello della Spesa, e così via.

Il numero massimo di serie depositabili nel Carrello è uguale a 20. Per cui se l'utente decide di voler ricercare un numero di serie superiore a 20, dovrà prima depositare le prime venti, fare su di esse le operazioni necessarie, togliere dal carrello un certo numero di serie, quindi inserire quelle rimanenti.

L'utente può compiere le operazioni di deposito e di rimozione dal Carrello della Spesa quante volte vuole.

Una volta che alcune serie sono state depositate nel Carrello della Spesa, l'utente può decidere di visualizzarne i valori. E' possibile la visualizzazione contemporanea al massimo di 5 serie.

Per cui se l'utente ha depositato nel Carrello 20 serie, dovrà togliere il segno di spuntatura da 15 serie, e procedere alla visualizzazione dei dati delle serie rimaste spuntate. Quindi togliere il segno di spuntatura da quelle già visualizzate, spuntarne altre cinque e procedere alla visualizzazione, e così via.

L'utente per gestire il Carrello della spesa utilizza la seguente schermata:

The screenshot shows a web browser window with the following elements:

- Browser Title:** Amerigo ricerca per ateco - Microsoft Internet Explorer
- Address Bar:** http://sebou91/amerigo/SelAteco.asp
- Page Header:** Banca dati Congiunturale (Constat logo) and Ricerca per Ateco. A red bar contains the word 'Reference'.
- Main Content:**
 - Text: Sono state selezionate: 3 serie
 - Link: Click for Metadata
 - Table of 3 series:

Icon	Series Name	Period	Checkbox
	estrazioni di minerali; carbon fossile; importazioni intra U.E. (dichiarazioni mensili)	[inizio: 1995/1. fine: 2000/6]	<input checked="" type="checkbox"/>
	estrazioni di minerali; carbon fossile; importazioni intra UE (dichiarazioni trimestrali)	[inizio: 1995/1. fine: 1999/4]	<input checked="" type="checkbox"/>
	estrazioni di minerali; carbon fossile; importazioni intra UE (dichiarazioni annuali)	[inizio: 1995/1. fine: 1999/1]	<input checked="" type="checkbox"/>
 - Date Range Selector:

Data In	Periodo in	Data fin	Periodo fin
1995	1	2000	6
 - Buttons: Estrai serie, Altre serie, Conferma la selezione, Annulla, Salva
 - Footer: E-mail link, (Risoluzione grafica ottimale - 800x600)
- Taskbar:** Shows 'Operazione completata', 'Intranet locale', and several open applications including Avvio, Gestione risorse, Microsoft Word, Microsoft PowerPoint, and the current browser window.

Esaminiamo le funzioni che vengono attivate all'interno di questa schermata:

- Cliccando sull'immagine indicata dalla scritta [Click for Metadati] è possibile visualizzare i Metadati della serie corrispondente.
- Cliccando sul bottone [Altre serie] è possibile selezionare serie di un dominio differente, aggiungendole al carrello della spesa. Se prima di premere tale bottone, l'utente ha tolto il segno di spuntatura ad alcune serie, queste verranno rimosse dal Carrello della Spesa.
- Cliccando sul bottone [Conferma la selezione] è possibile aggiornare la situazione nel carrello della spesa. In genere è utile utilizzare tale bottone quando si toglie il segno di

spuntatura da alcune serie per rimuoverle dal carrello della spesa.

- Cliccando sul bottone [Annulla] si svuota completamente il Carrello della Spesa e si inizia una nuova ricerca.
- Cliccando sul bottone [Estrai Serie] è possibile visualizzare i valori delle serie che presentano un segno di spuntatura. Prima di premere questo bottone bisogna fare due operazioni:
 - a) Definire l'intervallo temporale delle serie scegliendo l'anno e il periodo iniziale, l'anno e il periodo finale, con l'accortezza che la data iniziale non può essere maggiore di quella finale. In ogni caso il sistema si accorge, eventualmente di tale errore segnalandolo all'utente.
 - b) Lasciare il segno di spuntatura al massimo a 5 serie. Le altre non verranno rimosse dal Carrello della Spesa, ma resteranno disponibili per le estrazioni successive.

7.6 Salvataggio delle ricerche.

L'utente che ha depositato nel carrello delle spesa un certo numero di serie, magari provenienti da domini di aggregazione o Gruppi di Ateco differenti, ha la possibilità di salvarne il contenuto (**Entità**) in maniera tale che in un sessione futura non dovrà ripercorrere nuovamente l'albero per ricercare le serie d'interesse, ma potrà richiamare l'Entità per avere nel Carrello della Spesa ciò che quella Entità possiede.

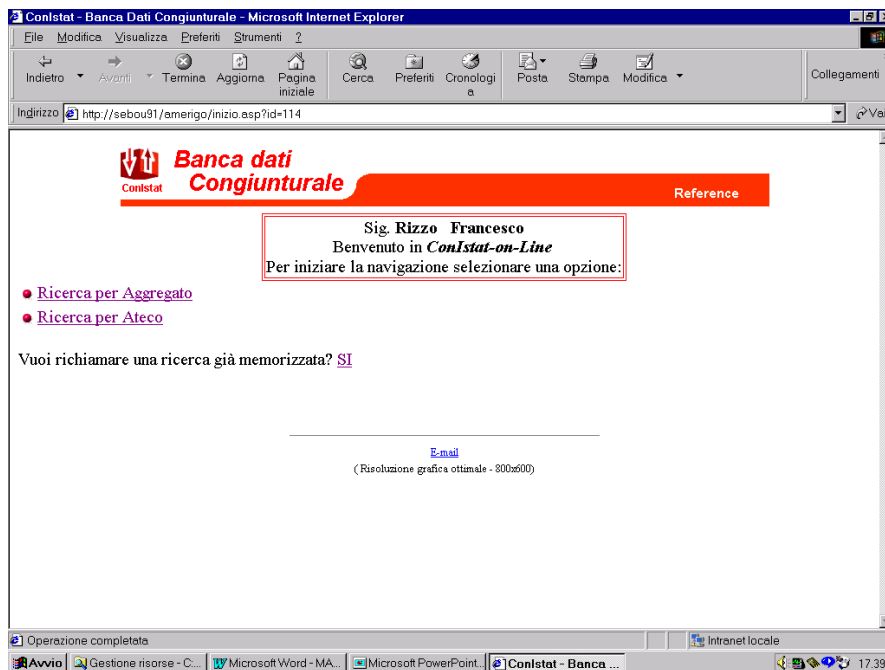
Ciascuna Entità può contenere al massimo 20 serie, ed è possibile memorizzare fino a 5 Entità.

La memorizzazione di una Entità avviene premendo il bottone [Salva] della schermata prima illustrata. Il sistema richiederà un nome di salvataggio mediante il quale l'utente potrà riconoscere una Entità dalle altre.

Se l'utente vuole memorizzare altre Entità dopo la quinta, è necessario cancellarne qualcuna già memorizzata.

7.7 Richiamo di una Entità memorizzata.

E' possibile richiamare una Entità precedentemente memorizzata facendo click sulla scritta [SI] nella schermata in cui si cominciano le ricerche:



7.8 Download di file.

Dopo aver estratto i valori di una o più serie è possibile fare il download di file in formato **TXT** o **CSV** sul proprio computer. Per far ciò è necessario fare click con il tasto destro del mouse sulle immagini sotto evidenziate e quindi scegliere l'opzione opportuna:

