

Soluzioni tecnologiche a supporto dello sviluppo di sistemi informativi sulla qualità: l'esperienza SIDI

A cura di Concetta Pellegrini

*ISTAT – Istituto Nazionale di Statistica
Servizio Metodologie, Tecnologie e Software per la produzione dell'Informazione statistica
(MTS)*

Indice

1. Introduzione	4
2. Architettura dei sottosistemi SIDI.....	5
2.1. SIDI gestione	6
2.1.1 Obiettivi	6
2.1.2 Architettura del sistema	6
<i>Processi di standardizzazione e generalizzazione</i>	7
2.2. SIDI-TOP	9
2.2.1 Obiettivi	9
2.2.2 Architettura del sistema	10
<i>Utilizzo dell'Application Server Tomcat</i>	11
<i>Utilizzo del framework Struts 1.1</i>	11
Struttura del layout di navigazione	13
<i>Utilizzo di un menu dinamico</i>	15
<i>Il framework di Accesso ai Dati</i>	15
<i>Utilizzo di package per la realizzazione dei grafici</i>	16
<i>Tool di sviluppo adottati</i>	17
2.3 ASIMET	19
2.3.1 Obiettivi	19
2.3.2. Architettura del sistema	19
2.4. SIQUAL	21
2.4.1 Obiettivi	21
2.4.2. Architettura del sistema	21
<i>Utilizzo del motore di ricerca ORACLE</i>	22
Riferimenti Bibliografici	24

Abstract

Il documento vuole offrire una panoramica sulle soluzioni tecnologiche e architetture realizzate all'interno del sistema SIDI.

Il Sistema Informativo di Documentazione delle Indagini (SIDI) è stato realizzato per fornire uno strumento generalizzato e standardizzato per il controllo di qualità dei processi statistici d'Istituto.

Nel documento vengono riportate le evoluzioni applicative e tecnologiche che nel tempo hanno caratterizzato il sistema e che hanno permesso di realizzare dei sistemi di consultazione per l'utenza interna o per la diffusione all'esterno.

Questo sistema è nato da una forte esigenza all'interno dell'Istituto di armonizzare e standardizzare la gestione e la consultazione dei metadati di indagine.

Ciò è stato possibile attraverso il lavoro congiunto di professionalità metodologiche ed informatiche che hanno permesso la progettazione e realizzazione dei sistemi che vengono descritti nel documento.

1. Introduzione

Il sistema informativo SIDI è stato realizzato per fornire uno strumento, fortemente standardizzato, per il controllo di qualità dei processi produttivi delle indagini dell'Istituto.

SIDI gestisce metadati sui processi di produzione e sul contenuto informativo delle rilevazioni e delle principali elaborazioni dell'Istituto.

Il sistema gestisce anche serie storiche di indicatori standard relativi alle principali fasi del processo produttivo delle rilevazioni, in accordo con le direttive Eurostat.

La gestione integrata di metadati e di indicatori di qualità consente una valutazione qualitativa e quantitativa dei processi statistici nel tempo e quindi si può dire che SIDI è uno strumento con un duplice ruolo: di diffusione delle informazioni relative al processo produttivo di indagine e di monitoraggio dell'attività per valutare la qualità dei processi statistici.

Accanto al sistema principale SIDI sono stati sviluppati sistemi specifici:

- SIDI per la gestione dei metadati e degli indicatori di qualità, da parte dei Responsabili di Indagine, e per la gestione e validazione dei tesauri (tesauro è un repertorio di voci che descrivono un particolare aspetto dell'indagine) da parte dei Responsabili di Qualità.
- SIDI-TOP per la Intranet d'Istituto è il sistema web-based di interrogazione e navigazione dei metadati e degli indicatori di qualità standard delle indagini.
- ASIMET è il sistema che, attraverso la metainformazione contenuta nel sistema SIDI, permette la produzione informatizzata delle note metodologiche dell'Annuario Statistico Italiano.
- SIQUAL è il sistema web-based di interrogazione dei metadati, realizzato per utenti esterni e quindi per Internet.

2. Architettura dei sottosistemi SIDI

Di seguito viene illustrata l'architettura dei sottosistemi di SIDI:

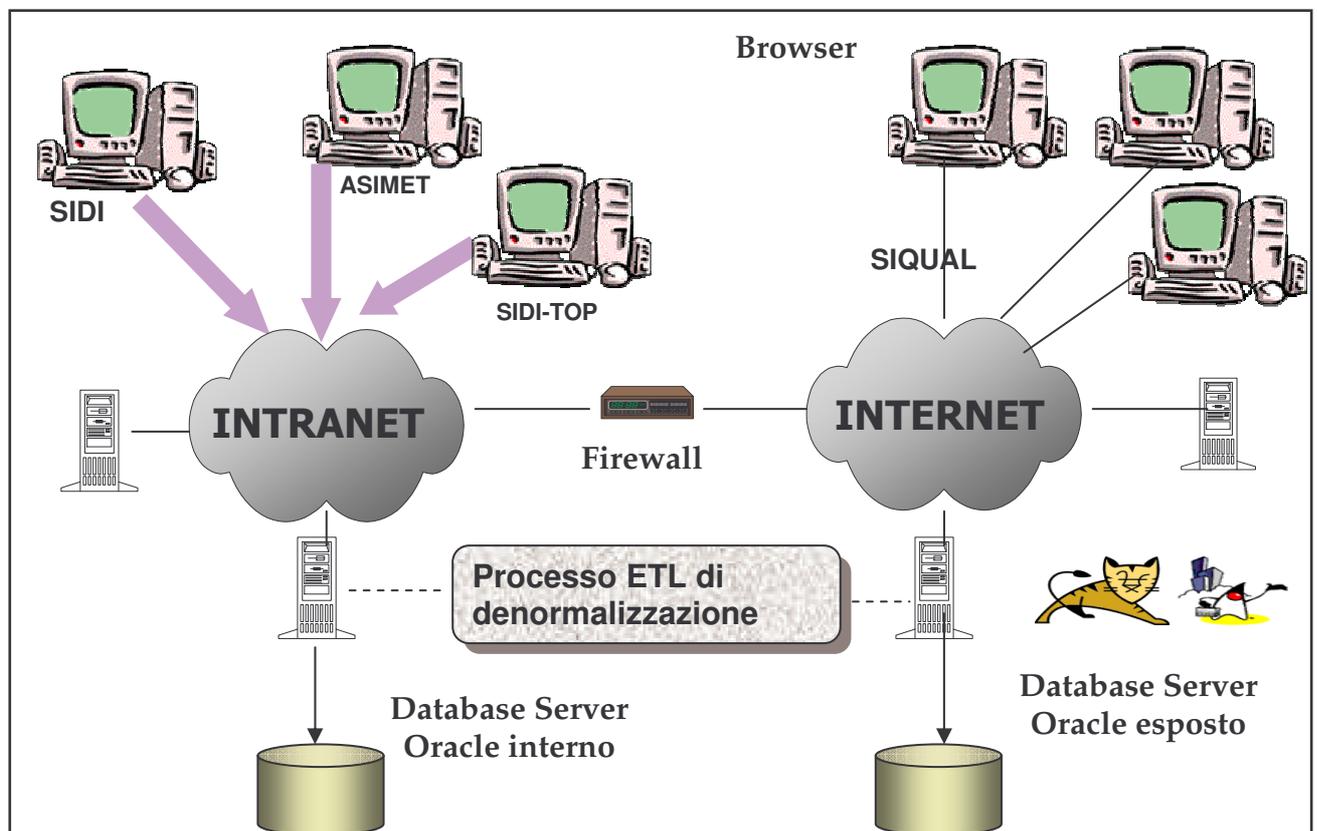


Figura: 1- Il sistema SIDI

2.1. SIDI gestione

2.1.1 Obiettivi

E' il sistema realizzato per l'acquisizione e gestione dei metadati e degli indicatori di qualità contenuti in SIDI. La prima versione del sistema è stata sviluppata nel 1998 ed è stata una delle prime esperienze in Istituto di architettura client-server realizzata utilizzando una lan di pc in una rete locale. Il sistema nel tempo, seguendo le evoluzioni tecnologiche di Istituto ha subito diverse reingegnerizzazioni, evoluzioni applicative e modifiche della piattaforma operativa utilizzata. Il sistema, si avvale di un'architettura client-server con circa un centinaio di postazioni client che utilizzano l'applicazione ed un server dove risiede l'RDBMS Oracle¹. E' in fase di realizzazione la migrazione del sistema in architettura object-oriented in ambiente web.

2.1.2 Architettura del sistema

Il sistema è stato realizzato su piattaforma Windows, sviluppato con il pacchetto applicativo Developer 2000 e utilizza un Database relazionale Oracle , situato su un server UNIX di Istituto. E' stato sviluppato su un'architettura client/server basata su due livelli (Figura 2):

- **Client Tier** (o **Front-end**) : livello dedicato all'interfaccia grafica, dove gli utenti, attraverso i PC, effettuano le richieste al lato server.
- **Server Tier** (o **Back-end**) : lato server del sistema dove è collocato il DB che risponde alle richieste dell'utente ed effettua l'aggiornamento dei dati.

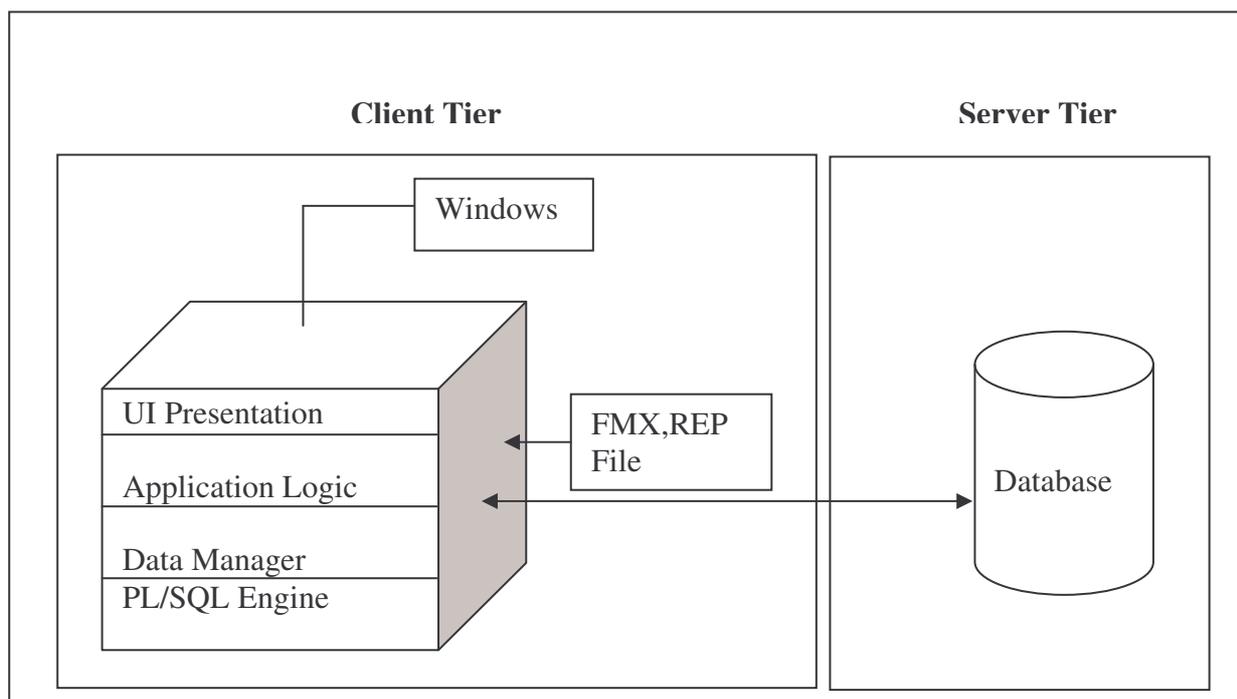


Figura 2: Architettura di SIDI gestione

¹ (<http://www.oracle.com/>)

Processi di standardizzazione e generalizzazione

Ai fini di garantire l'omogeneità e l'armonizzazione di concetti e di permettere la confrontabilità delle informazioni, relative ai diversi processi dell'Istituto, è stata necessaria una forte standardizzazione sia per i metadati che per gli indicatori di qualità.

Per gli indicatori di qualità sono stati individuati gruppi di indicatori standard, relativi alle diverse fasi del processo produttivo.

Per i metadati la standardizzazione è stata attuata attraverso l'utilizzo di "tesauri".

Tesauro è un repertorio di voci che descrivono un particolare aspetto dell'indagine ². Sono un insieme di voci standard ma ampliabili dal responsabile di indagine subendo un processo di validazione da parte dei responsabili della qualità.

I tesauri possono essere strutturati come liste elementari piatte oppure avere una struttura gerarchica questo permette, partendo da una voce sintetica, di arrivare alla voce elementare di massimo dettaglio (Esempio tesauro operazioni: si parte dalla fase del processo produttivo fino alla foglia terminale della gerarchia che l'operazione di dettaglio effettuata vedi **Figura 3**).

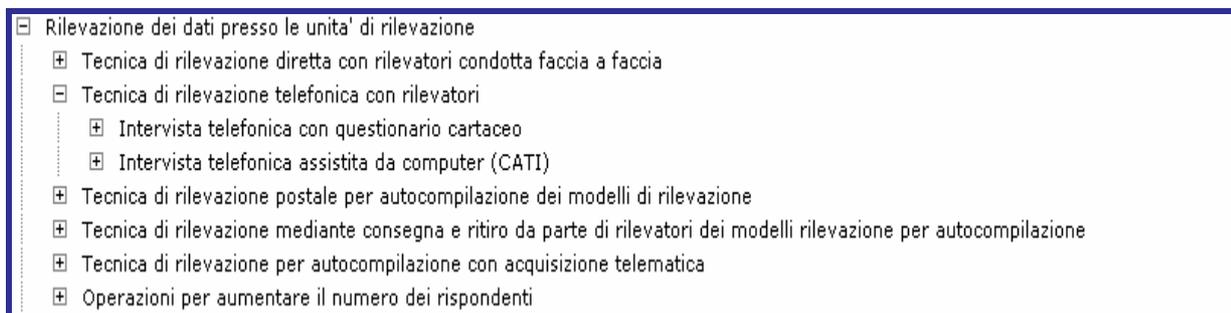


Figura 3 – Esempio di tesauro con struttura gerarchica

Per gli indicatori di qualità sono stati progettati ed attuati diversi processi a supporto dell'utente finale:

1. parametrizzazione delle definizioni e delle funzioni di calcolo degli indicatori di qualità;
2. generalizzazione della procedura di validazione dei valori assoluti e del calcolo degli indicatori realizzata con procedure PL/SQL;
3. parametrizzazione dei controlli formali e dei messaggi di errore che vengono effettuati durante il calcolo degli indicatori;
4. possibilità di acquisire i valori assoluti di singola indagine attraverso connessioni ad altri DB Oracle.

² (Tesauri del Sistema Informativo di Documentazione delle Indagini (SIDI). Istat, Documenti n. 1, Anno 2004 – Autori: Brancato G., Simeoni G.)

In riferimento a questo ultimo punto molto rilevante è stata l'integrazione realizzata in collaborazione con il servizio delle Statistiche strutturali sulle imprese dell'industria e dei servizi. All'interno di tale servizio è stato realizzato un sistema integrato per il monitoraggio delle risposte pervenute dalle imprese sullo stato delle liste di rilevazione. Tale sistema sviluppato in Oracle forms Developer2000, utilizza un RDBMS Oracle. All'interno dell'applicazione, è stata realizzata, una procedura in PL/SQL per acquisire automaticamente i valori assoluti per il calcolo degli indicatori relativi allo stato delle liste di riferimento ed alla rilevazione in modo automatico. Questo è stato possibile realizzando un DB-link tra i due sistemi e creando delle tabelle di trascodifica che hanno permesso di riassegnare coerentemente le codifiche usate nell'applicazione a quelle di SIDI.

2.2. SIDI-TOP

2.2.1 Obiettivi

SIDI-TOP è un sistema realizzato per la consultazione dei metadati e degli indicatori di qualità delle indagini realizzato per la Intranet. Il sistema utilizza per la consultazione i dati presenti nel Database Oracle popolato attraverso il sistema SIDI, ed è il sistema di accesso e interrogazione per l'analisi dei dati.

La diversità e molteplicità di utenti interessati ad interrogare il sistema, l'interfaccia grafica da prospettare, l'accessibilità e disponibilità delle informazioni, l'eliminazione dei limiti sistemistici ed hardware hanno portato a progettare e realizzare il sistema utilizzando una architettura Object Oriented per ambiente WEB.

L'architettura definita è da ritenersi frutto di una valutazione basata sui migliori strumenti software di infrastruttura e sviluppo orientati alla realizzazione di Web Applications ad oggi disponibili nel panorama dell'Open Source.

Il sistema permette di:

- accedere alle informazioni delle indagini attraverso diversi criteri di selezione
- visualizzare il contenuto informativo ed i metadati di indagine
- consultare la documentazione on-line messa a disposizione dai responsabili della qualità del sistema e dai responsabili di indagine
- gestire serie storiche di indicatori di qualità relativi alle diverse fasi del processo produttivo, quindi di valutare la qualità delle indagini nel tempo
- mettere a confronto la qualità di indagini diverse.

Il sistema visualizza i dati, relativi agli indicatori di qualità, sotto vari aspetti:

- Analisi di serie storiche di dati (Andamento temporale con grafici a spezzate)
- Analisi territoriale (con cartine dell'Italia suddivisa in 5 ripartizioni geografiche)
- Analisi per componente percentuale (con grafici a torta)
- Analisi tabellare (con report esportabili in Excel).

Un'ulteriore analisi è data attraverso la funzionalità di confronto con medie: per questo sono stati definiti dei raggruppamenti di indagini utilizzando come criteri quei fattori che maggiormente influenzano alcune tipologie di errori per le diverse fasi del processo produttivo. Vengono quindi calcolati i valori medi degli indicatori in corrispondenza di ciascun raggruppamento che sono poi visualizzati sotto forma di grafici a spezzate nel sistema.

2.2.2 Architettura del sistema

Il sistema ha una struttura a livelli per garantire un elevato tasso di flessibilità.

I livelli realizzati sono:

1. una infrastruttura software indipendente dalla piattaforma, costituita dalla Java Virtual Machine e dall'Application Server Tomcat ³.
2. un framework a cui viene affidata la gestione del flusso dell'applicazione, con il compito di separare la presentation logic dalla business logic adottata. Questo framework è costituito da Struts 1.1 ⁴.
3. una serie di moduli per effettuare le varie elaborazioni. Sono identificati: un Menu dinamico, con il compito di sincronizzare il menu con le selezioni dell'utente; un Navigation Stack, che traccia le fasi di navigazione dell'utente; i moduli di Selezione Indagini, Elaborazioni Grafici ed Elaborazione Dati.

Lo schema seguente illustra i vari livelli dell'applicazione SIDI-TOP.

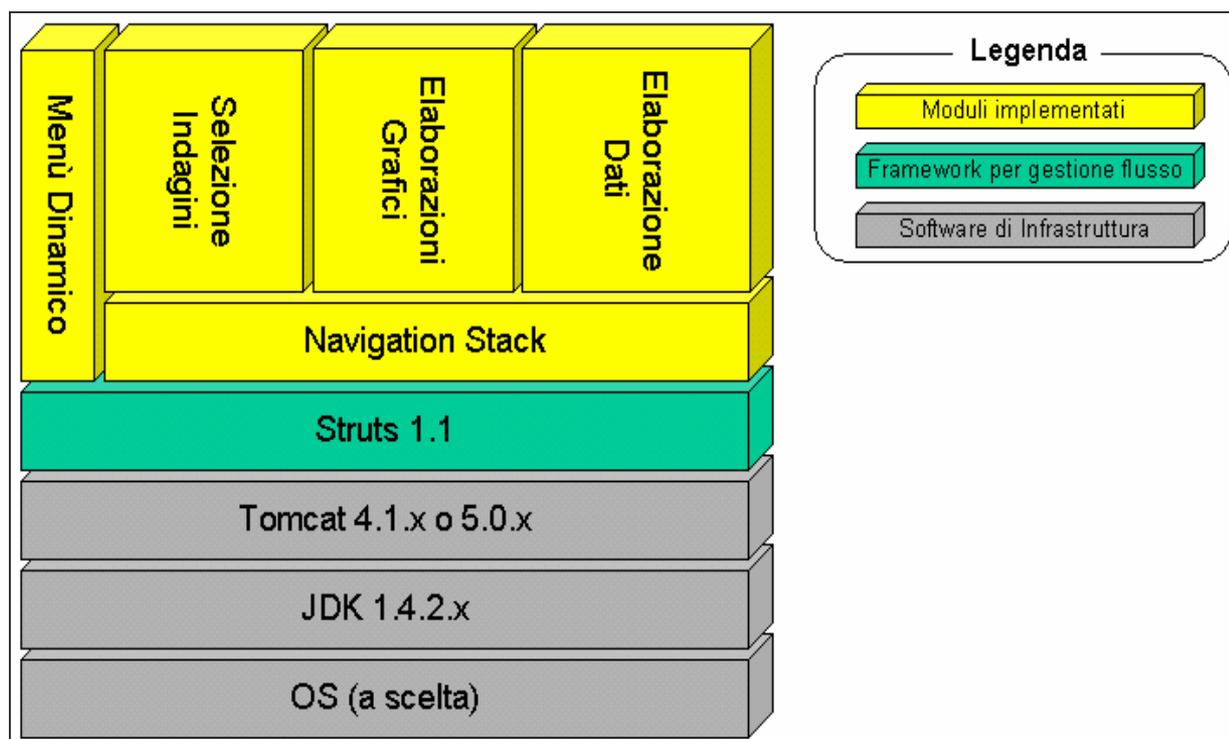


Figura 4- Layers dell'applicazione SIDI-TOP

³ (<http://tomcat.apache.org/>)

⁴ (<http://struts.apache.org/>)

Utilizzo dell'Application Server Tomcat

Gli Application Server sono delle piattaforme software in grado di erogare servizi mediante componenti distribuiti.

Ad oggi, uno dei principali standard per la realizzazione di infrastrutture di software distribuito è stato ideato dalla Sun Corporation con il nome di Java 2 Enterprise Edition (J2EE).

Un application server J2EE si pone come sistema di Middleware ideale nelle cosiddette architetture N-Tier ovvero ad N strati.

Tipicamente le architetture N-Tier delineano:

- il client Tier composto dall'insieme di piattaforme client delle applicazioni o servizi che si desidera erogare
- il Data Tier composto dall'insieme di servizi, software, basi di dati o infrastrutture software che si desidera mettere a disposizione del Client Tier
- il Middle Tier nel quale si concentrano le soluzioni server che consentono l'integrazione dei servizi/componenti del Data Tier e la loro esposizione verso il Client Tier e che mettono a disposizione ulteriori servizi.

Tomcat è un Servlet Container ed un JSP Engine, quindi un motore in grado di eseguire lato server applicazioni web basate su architetture J2EE e costituite da componenti Servlet e da pagine JSP come appunto SIDI-TOP.

Tra gli ulteriori servizi a disposizione della piattaforma è stato gestito e configurato un Connection Pool, componente che pre-istanza connessioni verso un Database per minimizzare i tempi di accesso verso i dati.

Utilizzo del framework Struts 1.1

Struts è un framework MVC per il supporto al flusso dei servizi Web realizzato su piattaforma J2EE. Il paradigma MVC prevede la separazione delle responsabilità della produzione dei contenuti dinamici in tre componenti principali: Model, View e Controller.

Il Controller (generalmente implementato mediante una Servlet), ha il compito di ricevere la "Request" proveniente dal browser client. Analizzati i parametri si occupa eventualmente di accedere ai dati e di istanziare una classe di tipo Java Bean caricandola con i dati reperiti. Successivamente trasmette la Request ricevuta ad un terzo componente, la JSP che ha il solo ruolo di rappresentare i dati che legge dal modello. Dopo aver generato la pagina HTML la JSP la trasmette infine al client browser.

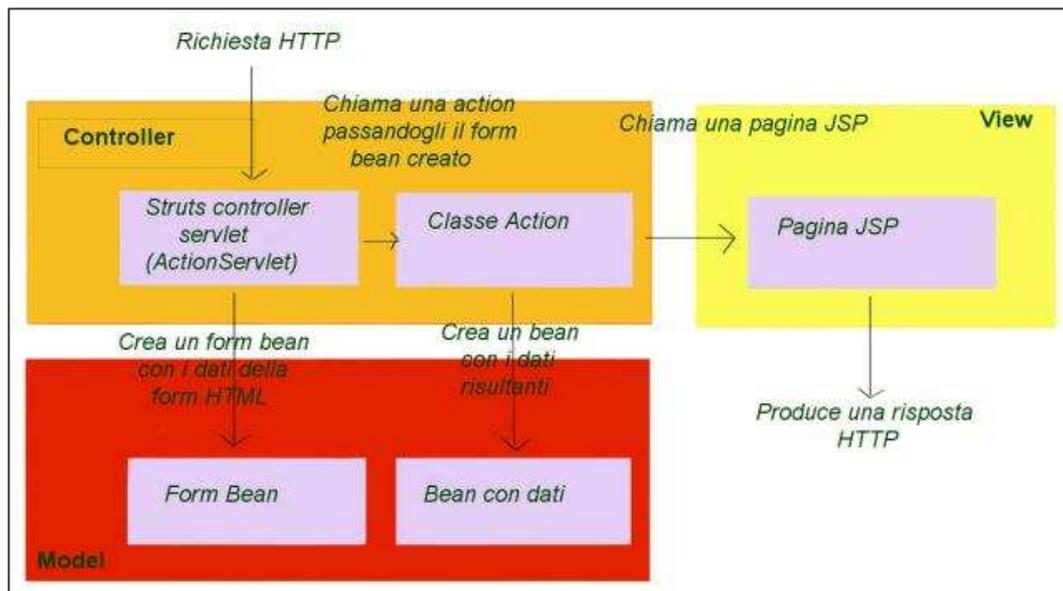


Figura 5- Paradigma MVC

La particolarità di Struts risiede nella possibilità di definire mediante un file in formato XML, “struts-config.xml”, qual è il flusso logico della navigazione del sito e quali modelli vanno caricati.

In questa sorta di “file di istruzioni” del flusso sono descritte le classi che implementano:

- i Form Bean: Java Bean destinati ad accogliere e validare i parametri della request generalmente provenienti da forms HTML;
- le Action: contenenti singole parti della Business Logic dell’applicazione che agiscono “per delega” del controller principale.

Inoltre sono anche definiti i flussi in uscita da ciascuna Action ai quali viene attribuito un identificativo logico testuale.

L’architettura di Struts prevede la registrazione della servlet principale (ActionServlet) di controllo di flusso, responsabile di prendere in carico tutte le Request che vengono identificate destinate a Struts mediante il loro URL pattern.

La ActionServlet delega il controllo del flusso al “RequestProcessor” che ha il compito di coordinare gli oggetti responsabili della gestione del modello e della Business Logic secondo quanto configurato nel file di controllo “struts-config.xml”.

Lo schema seguente illustra il ruolo svolto dalle varie componenti di Struts nel coordinamento del flusso:

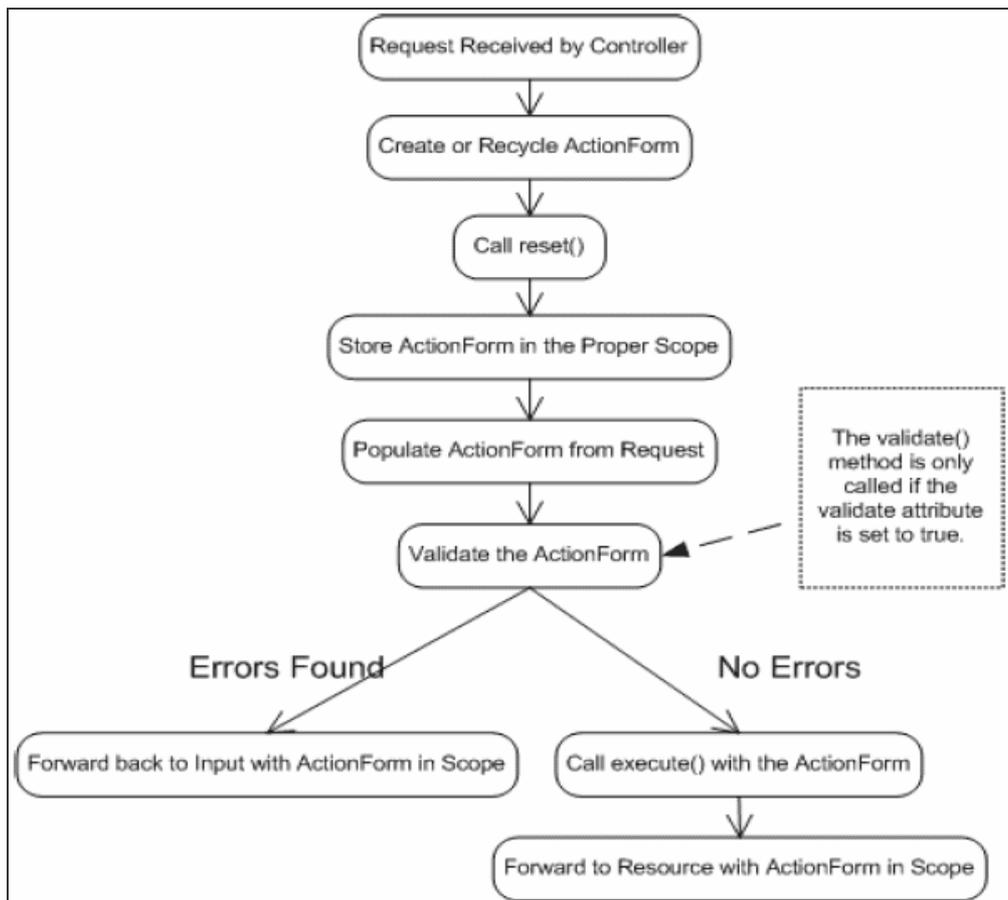


Figura 6- Activity Diagram delle principali componenti di Struts

Sono le varie Action che contengono la business logic adeguata ad invocare (*invoke()*) i vari metodi degli oggetti (*Service*) dell'applicativo (classi di accesso ai dati, ecc.) e a caricare il contenuto (*storeData()*) nel modello che sarà utilizzato per la rappresentazione degli stessi.

Struttura del layout di navigazione

Per mantenere il più alto grado di compatibilità con i browser attualmente sul mercato, senza pregiudicare la fruibilità del prodotto, si è reso necessario l'utilizzo di "Frames HTML".

Le versioni dei browser di riferimento per garantire la compatibilità con il maggior numero di utenti sono Netscape (dalla versione 4.51 sino al recente Mozilla) e Internet Explorer (dalla versione 4.01 alla versione 6.x).

Il layout delle pagine di SIDI-TOP è riconducibile ai seguenti:

<p>Solo Menu e Body</p> <p>Il Menù compare a sinistra ed è dinamico, ovvero varia il suo contenuto in base al contesto ed alle scelte effettuate durante la navigazione del sito</p>	<table border="1" data-bbox="810 206 1310 271"> <tr> <td data-bbox="810 206 954 271">Menu</td> <td data-bbox="954 206 1310 271">Body</td> </tr> </table>	Menu	Body		
Menu	Body				
<p>Menu, Header e Body</p> <p>La zona “header” è utilizzata per visualizzare informazioni di intestazione fisse (senza scorrimento / scrolling) o form di ricerca i cui risultati sono visualizzati nella zona “body”</p>	<table border="1" data-bbox="810 510 1310 741"> <tr> <td data-bbox="810 510 954 741" rowspan="2">Menu</td> <td data-bbox="954 510 1310 575">Header</td> </tr> <tr> <td data-bbox="954 575 1310 741">Body</td> </tr> </table>	Menu	Header	Body	
Menu	Header				
	Body				
<p>Menu, Footer e Body</p> <p>Il Footer è usato tipicamente per contenere informazioni di riepilogo (ad esempio l’elenco indagini selezionate per il “confronto indagini”)</p>	<table border="1" data-bbox="810 880 1310 1122"> <tr> <td data-bbox="810 880 954 1122" rowspan="2">Menu</td> <td data-bbox="954 880 1310 1059">Body</td> </tr> <tr> <td data-bbox="954 1059 1310 1122">Footer</td> </tr> </table>	Menu	Body	Footer	
Menu	Body				
	Footer				
<p>Menu, Header, Footer e Body</p>	<table border="1" data-bbox="810 1184 1310 1473"> <tr> <td data-bbox="810 1184 954 1473" rowspan="3">Menu</td> <td data-bbox="954 1184 1310 1249">Header</td> </tr> <tr> <td data-bbox="954 1249 1310 1417">Body</td> </tr> <tr> <td data-bbox="954 1417 1310 1473">Footer</td> </tr> </table>	Menu	Header	Body	Footer
Menu	Header				
	Body				
	Footer				
<p>Popup</p> <p>Le finestre di “popup” compaiono quando sono seguiti dei link che portano a “zone di navigazione indipendenti” quali ad esempio la pagina di Help o dei Tesauri</p>	<table border="1" data-bbox="810 1594 1310 1809"> <tr> <td data-bbox="810 1594 1310 1809">Non definito a priori; generalmente schermo pieno o con navigazione “a Tab” indipendente dall’applicazione</td> </tr> </table>	Non definito a priori; generalmente schermo pieno o con navigazione “a Tab” indipendente dall’applicazione			
Non definito a priori; generalmente schermo pieno o con navigazione “a Tab” indipendente dall’applicazione					

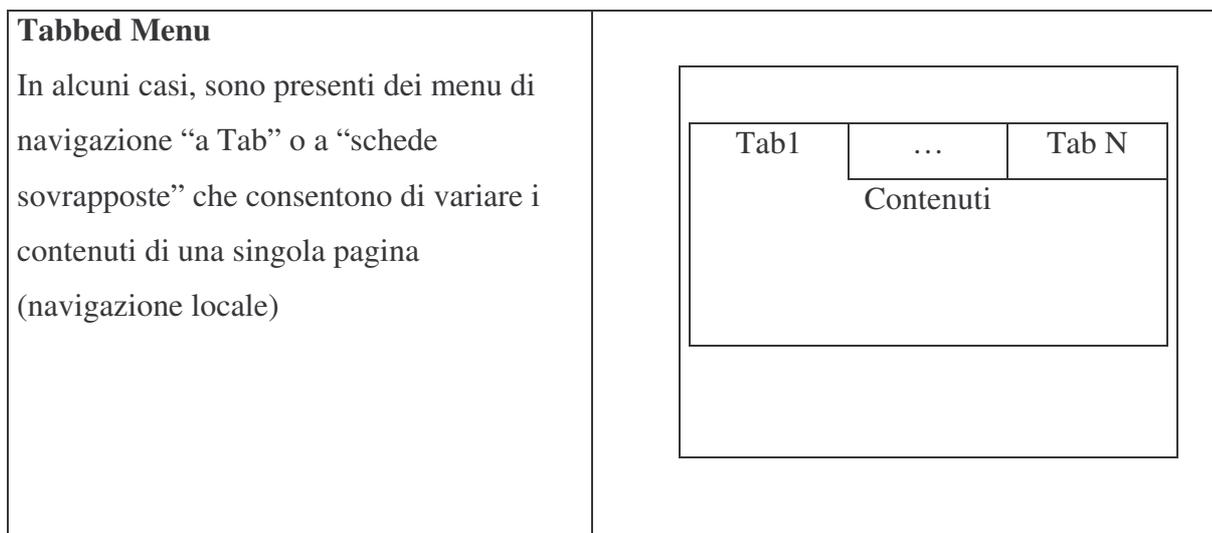


Figura 7 - Schemi dei layout dei frame

Utilizzo di un menu dinamico

Al fine di guidare l'utente nella navigazione del sistema è stato realizzato un menu dinamico, che cambia e si adatta alla navigazione di ogni singolo utente.

Per ottenere un'elevata apertura è stato scelto di utilizzare una tecnologia estremamente flessibile facile da modificare ed adattare alle diverse esigenze: **XML**.

Infatti, sia che si tratti di aggiungere, cancellare o modificare voci di menu, sia che si tratti di modificare i vincoli che regolano la visibilità del menu stesso sarà possibile intervenire direttamente sul file menu.xml che contiene la struttura del menu.

Partendo dagli Schemi XML definiti per i menu sono stati generati due package per ogni menu: il primo contiene le interfacce che definiscono quali metodi sono necessari per accedere al documento XML ed agli elementi che lo compongono; il secondo contiene le classi che implementano le interfacce.

Il framework di Accesso ai Dati

Il framework per l'accesso ai dati si suddivide in due package:

1. Il primo contiene le classi di tipo DAO (Data Access Object) che contengono i DAO Business Methods che caricano i dati dal DB.
2. Il secondo contiene i Value Object generati a partire dalle strutture delle tabelle o query effettuate dai DAO Business Methods.

I Value Objects sono dei Java Bean che mappano le colonne delle tabelle o query con le rispettive proprietà. Il loro scopo è quello di contenere le informazioni relative ad un record.

Per consentire una facile identificazione del record quando questo viene rappresentato su HTML è

stato prevista una classe astratta **SelectableBean** che contiene una proprietà aggiuntiva – **selected** – per memorizzare se il record in questione risulta selezionato dall'utente o meno e un metodo astratto **getIdentifier()** che sarà di volta in volta implementato per restituire un identificativo univoco del record in formato *java.lang.String*.

Utilizzo di package per la realizzazione dei grafici

I grafici sono stati creati utilizzando la libreria **JfreeChart**. Data la varietà di grafici necessari è stato creato un motore generico per la creazione dei grafici.

Le informazioni che il generatore si aspetta per poter visualizzare il grafico sono contenute nel **ChartDefinition**, un bean che contiene informazioni generali, quali titolo grafico, nome assi, ecc. e i dati veri e propri da rappresentare.

Il singolo dato viene rappresentato tramite il **chartItem** che contiene il valore dell'asse X, dell'asse Y e della Categoria. Questa terna di informazioni è sufficiente per la rappresentazione del dato su qualsiasi grafico.

L'elenco dei dati da rappresentare è memorizzato in un Array di **chartItem** detto **chartItemCollection** che farà poi parte a sua volta del **ChartDefinition**.

L'elaborazione vera e propria del grafico è delegata ad una serie di classi specializzate per il singolo grafico. Proprietà comune a tutte le classi specializzate è quella di mostrare al generatore dei grafici la stessa interfaccia così da garantire un'omogeneità di funzionamento.

Il reperimento dei dati è affidato ad una struttura esterna alla classe dei grafici, in modo da separare completamente la logica di accesso ai dati da quella di generazione e presentazione degli stessi.

E' richiesta la compatibilità del dato da rappresentare con la struttura elementare, rappresentante il punto, ovvero il **ChartItem**.

Una volta associate al **ChartDefinition** le informazioni generali sul grafico e l'elenco di dati (collezione di **ChartItem**) da rappresentare, non resta che invocare il motore che genera i grafici specificando il tipo di grafico desiderato. Il risultato viene restituito come file **.png** utilizzabile nel contesto della web application.

La creazione del generico grafico prevede una serie azioni che vengono illustrate nel seguito:

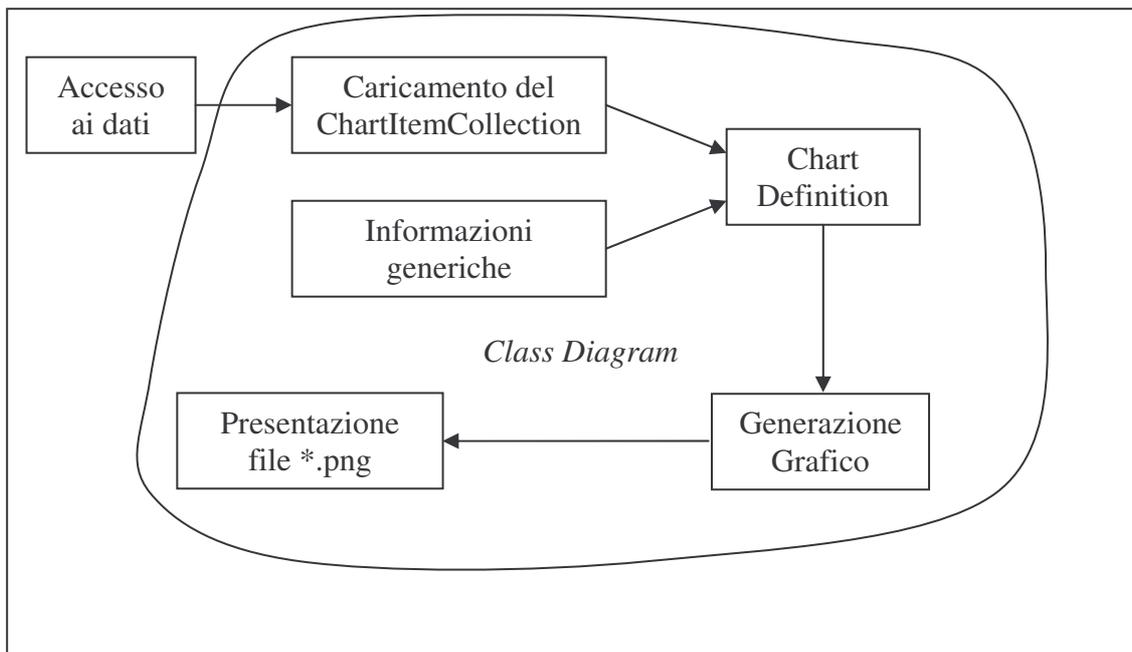


Figura 8- Funzionamento per la produzione del grafico

Tool di sviluppo adottati

Per lo sviluppo è stato utilizzato il tool visuale **Eclipse**⁵ molto utile per lo sviluppo in team di applicativi web, che si avvia a diventare lo standard dello sviluppo del software dei prossimi anni.

Eclipse è una piattaforma di sviluppo ideata da un consorzio di grandi società, chiamato Eclipse Foundation, e creata da una comunità strutturata sullo stile dell'open source. **Eclipse** è un ambiente di programmazione, multi piattaforma e multi linguaggio, molto versatile anche grazie alla sua natura open source. Questo tool visuale è stato utilizzato per lo sviluppo, la compilazione, il versioning, il debug ed il deploy dell'applicazione.

Architettura di Eclipse:

⁵ (<http://www.eclipse.org/>)

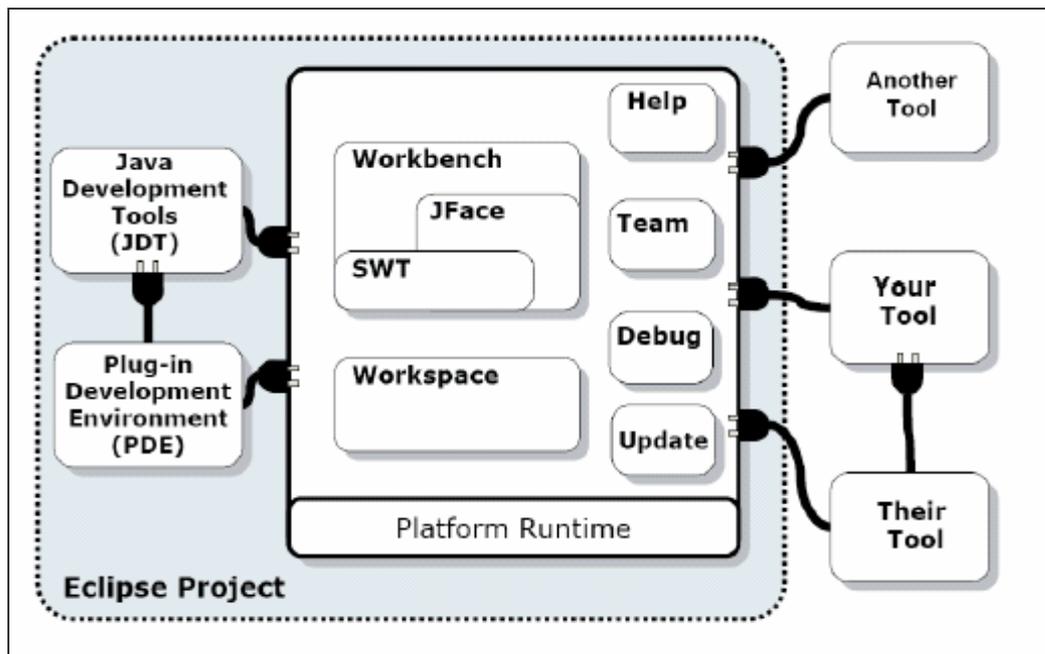


Figura 9 - Architettura di Eclipse

Pur essendo orientata allo sviluppo del progetto stesso, questo IDE (sistema di sviluppo integrato) è utilizzato anche per la produzione di software di vario genere. Il programma è scritto in linguaggio Java, ma utilizza delle librerie grafiche diverse da quelle della Sun Microsystems in modo da rendere più veloce la visualizzazione. Il sistema è incentrato sull'uso di plug-in, delle componenti software ideate per uno specifico scopo, per esempio la generazione di diagrammi UML. Nella versione base è possibile programmare in Java, con funzioni di aiuto come completamento automatico o suggerimento dei tipi di parametri e dei metodi, possibilità di accesso diretto a CVS e riscrittura automatica del codice in caso di cambiamenti nelle classi.

2.3 ASIMET

2.3.1 Obiettivi

Attraverso l'informazione documentata nel sistema SIDI è stato possibile realizzare il sistema informativo ASIMET⁶ per la predisposizione delle Note Metodologiche dell'Annuario Statistico Italiano. ASIMET permette di produrre le note metodologiche partendo dall'informazione validata di SIDI in un formato compatibile con il formato di stampa del volume. Questo progetto ha portato ad una standardizzazione dell'informazione diffusa e ad un maggiore controllo della qualità della stessa oltre che ad una diminuzione dei tempi nella predisposizione delle note.

L'alto livello di standardizzazione ha reso possibile l'integrazione del sistema con il sottosistema ASIMET automatizzando e armonizzando l'acquisizione dei metadati necessari a produrre la Nota Metodologica dell'Annuario Statistico Italiano. In previsione della pubblicazione dell'Annuario Statistico Italiano, l'Istat ha costituito un gruppo di lavoro avente il compito di studiare l'attuabilità di un sottosistema, che si integrasse con il sistema SIDI, per la produzione delle Note Metodologiche relative alle indagini che contribuiscono all'Annuario.

L'attuazione di questa integrazione non ha richiesto un elevato carico di lavoro perché il sistema SIDI è stato sviluppato operando una forte standardizzazione sia dal punto di vista metodologico che tecnico, infatti anche nello sviluppo dell'interfaccia utente ha permesso di distinguere le diverse funzionalità in modo uniforme, quindi il software aggiuntivo sviluppato per la gestione del sistema ASIMET, non ha compromesso l'applicativo già realizzato.

Per agevolare il compito dei Responsabili di Indagine, nel reperimento delle informazioni atte alla composizione della Nota Metodologica, partendo dal patrimonio informativo presente nel sistema SIDI, sono state realizzate delle tabelle di trascodifica che permettono di confrontare la corrispondenza tra le informazioni più sintetiche da fornire per l'ASI con quelle più dettagliate di SIDI.

Questa informatizzazione ha permesso di poter consultare on-line sui sistemi SIDI-TOP e SIQUAL (vedi par. 2.4) le note metodologiche prodotte per i singoli processi produttivi d'Istituto.

2.3.2. Architettura del sistema

Il sottosistema ASIMET è stato inserito come una nuova attività nel pacchetto applicativo ORACLE/Forms realizzato per SIDI gestione delle indagini. Tale scelta ha permesso di realizzare un

⁶ *Il Sistema ASIMET per la gestione delle Note Metodologiche dell'Annuario Statistico Italiano* -. Autori: G. Brancato, A. Cardacino, D. Camol, M. Angelucci, C. Pellegrini.

software perfettamente integrato con il sistema di gestione di SIDI effettuando un'efficiente interazione fra le basi dati dei due sistemi.

La predisposizione del file di stampa, delle note metodologiche, è stata realizzata con il software generalizzato CREANOTE sviluppato in Visual Basic che accede al DB Oracle tramite ODBC. L'output di stampa, realizzato in RTF, è stato implementato tramite l'utilizzo di librerie VBA di Microsoft Word.

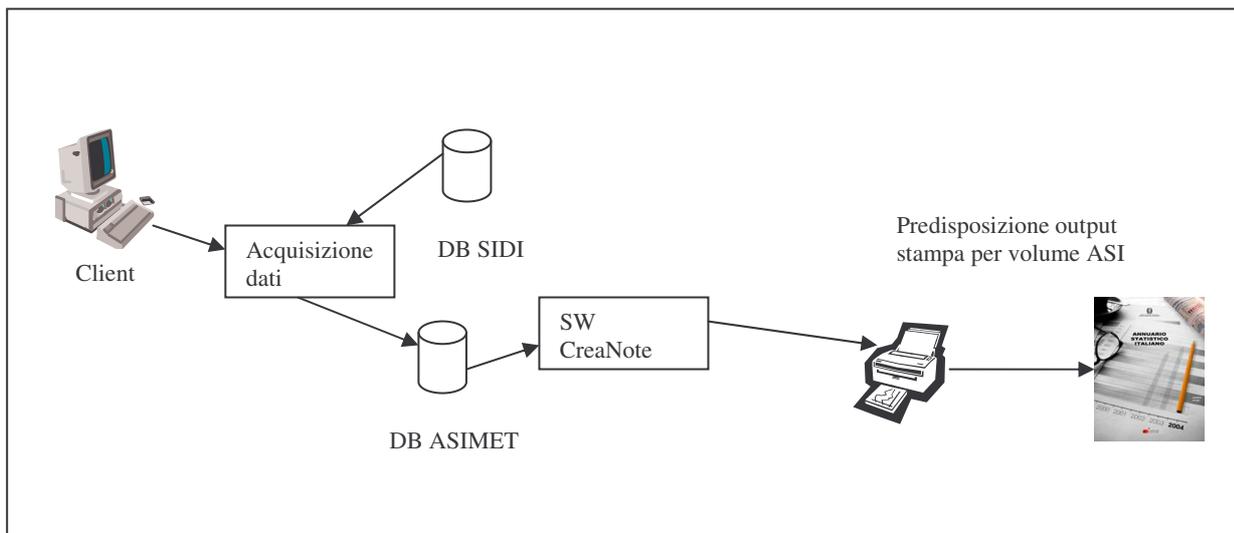


Figura 10 – Schema architetturale Asimet

2.4. SIQUAL

2.4.1 Obiettivi

Il sistema informativo sulla qualità (SIQUAL) è un sistema informativo di interrogazione: di metadati, di informazioni sulla qualità e di dati statistici accessibile attraverso Internet da parte di utenti esterni, , ottenuto a partire dai sistemi informativi SIDI e ASIMET.

Il sistema SIQUAL consente, agli utenti collegati via Internet, di interrogare i processi statistici dell'Istat: accedere alle informazioni relative ai processi produttivi statistici (rilevazioni ed elaborazioni) e di arrivare ai dati on line, quando disponibili. Le informazioni sono principalmente tratte dal sistema informativo di documentazione delle indagini SIDI. Particolarmente SIQUAL fornisce informazioni relative al contenuto informativo del processo, alle sue modalità di conduzione (scomposizione del processo in fasi e operazioni) e alla qualità intesa sia come qualità dell'informazione statistica diffusa (scheda qualità) che come insieme delle attività di prevenzione, controllo e valutazione dell'errore durante il processo (azioni di controllo). E' inoltre presente un'area di documentazione dove sono disponibili documenti sia di carattere generale che di interesse specifico per una rilevazione o elaborazione.

Tramite un processo di ETL è stato possibile, partendo dall'informazione di SIDI, realizzare una base dati denormalizzata atta ad effettuare ricerche per aree tematiche che non portino mai a dei risultati nulli.

L'interfaccia del sistema SIQUAL è di tipo GUI per Web Application conforme allo standard ISO. Il sistema SIQUAL è stato progettato, dal punto di vista dell'interfaccia, intorno a delle aree divise e dedicate, che sono state raggruppate per caratteristiche funzionali per rendere la navigazione agevole all'utente seguendo le direttive imposte dai responsabili del portale d'Istituto.

2.4.2. Architettura del sistema

La soluzione architetturale adottata è quella three-tiers che prevede l'utilizzo dell'RDBMS Oracle come database server di Apache/Tomcat e della tecnologia Jsp come application server e di un web browser come interfaccia utente.

La scelta effettuata è stata quella di creare un'applicazione web based seguendo quelle che sono le direttive dettate dal framework Struts versione 1.2, pertanto si è cercato di demandare la logica applicativa alla classe di business e alle varie action di controllo.

E' stata utilizzata la tecnologia Model View Controller del framework Struts 1.2, relativamente a:

- **Java Server Pages:** pagine scritte in linguaggio HTML con contenuti dinamici in Java
- **Data Beans :** classi java utilizzate per la conservazione dei dati presentati dalle jsp.

- **Action:** classi per l'intercettazione delle richieste del browser.

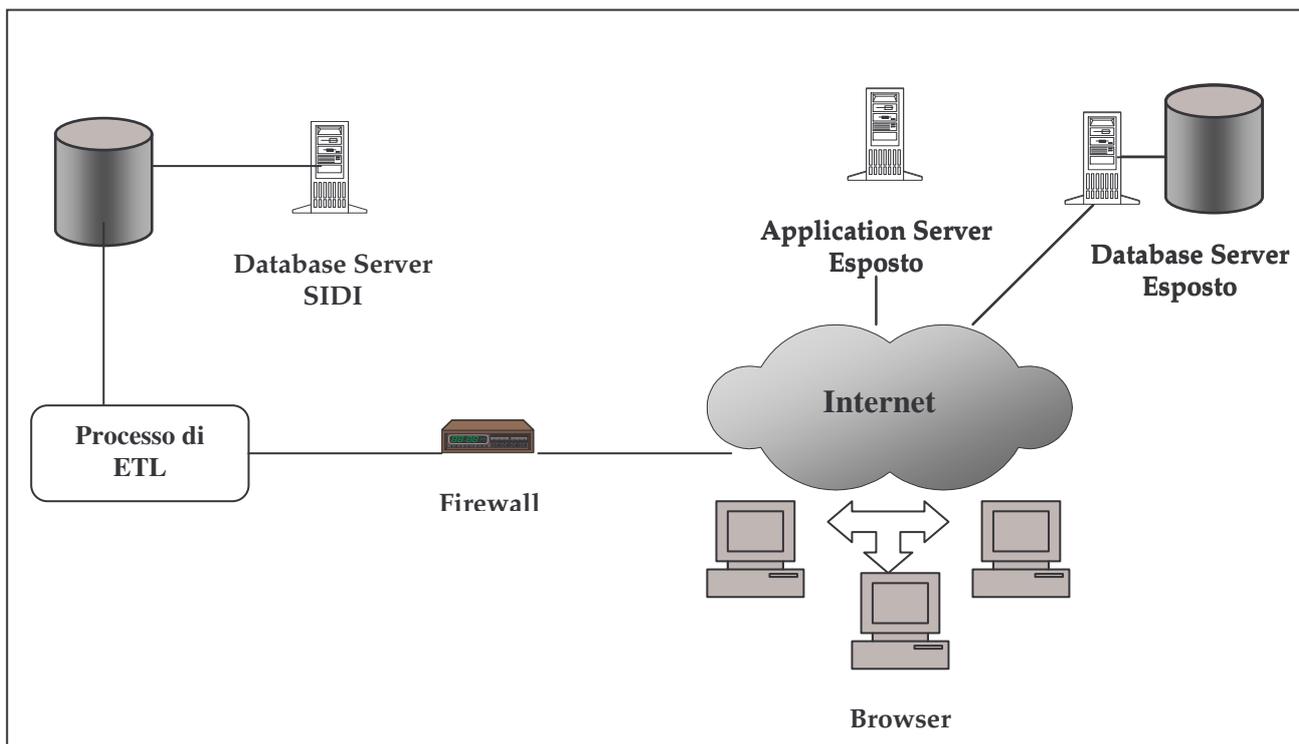


Figura 11 - Architettura SQUAL

L'applicazione risiede su due server Linux esposti (cioè visibile da Internet) su uno è installato il Web container Tomcat e l'applicazione, che viene rilasciata come .war, sull'altro è il database server Oracle.

Utilizzo del motore di ricerca ORACLE

Molto importante per un sistema di navigazione che tratta metainformazione avere un potente motore di ricerca che permetta di ricercare parole all'interno del sistema sia su tabelle di dati che all'interno di documenti. A tal fine è stata effettuata una sperimentazione che ha permesso di ottimizzare l'utilizzo del tool di Oracle interMedia.

Oracle interMedia è il sistema di gestione dei dati multimediali più potente ed utilizzato al mondo, arricchisce le feature di un normale database fornendo funzionalità per ricerche testuali avanzate (ricerche tematica, catalogazione automatica, highlighting, ecc.) e tecniche per il confronto di immagini, video e suoni.

Oracle Text, all'interno di questo tool, è il modulo dedicato alla gestione dei dati testuali, che è stato utilizzato all'interno del sistema. Molto approfondita è stata l'analisi delle tecniche di indicizzazione e interrogazione dei dati. Per poter utilizzare Intermedia su una colonna testuale o di BLOB è necessario creare un indice sulla colonna specificando la clausola "indextype is

`ctxsys.context`” ed eventualmente inserire parametri per un trattamento particolare. La creazione di un indice di tipo `ctxsys.context` genera tabelle aggiuntive contenenti i token e le informazioni necessarie all’identificazione delle tuple che contengono il testo ricercato. La modifica dei dati indicizzati non influisce sul contenuto degli indici, che devono pertanto essere sincronizzati ogni volta che i dati cambiano. Alla creazione dell’indice viene sempre associata una STOPLIST per eliminare parole poco significative al fine di una ricerca. Le parole escluse dall’indicizzazione non vengono trovate se ricercate da sole e ignorate se associate ad altre. La stoplist di default è in italiano e contiene gli articoli determinativi e indeterminativi, le preposizioni semplici ed articolate, alcuni avverbi, congiunzioni, ecc. La lista fornita da Oracle non è però completa, mancano infatti alcune variazioni di preposizioni e congiunzioni, alcuni avverbi e altre parole di uso comune ma non significative. Oracle fornisce anche una stoplist vuota, che consentirebbe di non escludere alcuna parola dall’indicizzazione.

E’ stata, quindi, creata una nuova stoplist basata su quella di default con l’aggiunta delle parole mancanti (ad esempio NON, AD, OD, ED, DELLA, ...). Inoltre, per la natura dei documenti interessati dalla ricerca in SIQUAL, è stato necessario aggiungere alla stoplist ulteriori parole, la classe “numbers” e le singole lettere dell’alfabeto.

Riferimenti Bibliografici

Apache. <http://www.apache.org/>

Apache Tomcat. <http://tomcat.apache.org/>

Apache Struts <http://struts.apache.org/>

Oracle. <http://www.oracle.com/>

Eclipse. <http://www.eclipse.org/>

Java: <http://java.sun.com>

Brancato G. (2004). *Il Sistema ASIMET per la gestione delle Note Metodologiche dell'Annuario Statistico Italiano*. Istat, Documenti, n.3, Anno 2004.

http://www.istat.it/dati/pubbsci/documenti/Documenti/doc_2004/2004_3.doc

Brancato G., Simeoni G. (2004) *Tesauri del Sistema Informativo di Documentazione delle Indagini (SIDI)*. Istat, Documenti n.1, Anno 2004.

http://www.istat.it/dati/pubbsci/documenti/Documenti/doc_2004/2004_1.doc

Brancato G, Pellegrini C, Simeoni G. e Signore M. (2004) *Standardising, Evaluating and Documenting quality: the implementation of Istat information system for survey documentation - SIDI*. European Conference on Quality and Methodology in Official Statistics. Mainz, 24-26 May 2004. CD ROM

Brancato G., Pellegrini C, D'Amore G. (2004) *IT solutions and information systems in support to quality*. Conferinta Nationala: Statistica Oficiala în Slujba Societatii, Bucuresti, 12-14 Iulie 2004.

http://www.insse.ro/evenimente/conferinta/Rezumate/ISTAT/En/ENAbstract_Brancato_Damore.pdf