

## **CHI SIAMO**

La CASPI 2000 Soc. Coop a.r.l. nasce nel 1996 rilevando alcune attività ed il personale della CASPI S.r.l., azienda indirizzata alla progettazione e sviluppo applicativi finalizzati alla realizzazione di sistemi informativi dal 1985. Sin dal primo momento l'ambito d'attività privilegiato è stata l'organizzazione informatica contabile, a cui si sono affiancati progetti per lo sviluppo di siti e servizi WEB, attività nella quale ha indirizzato gran parte degli investimenti nella reengineering del parco Software applicativo a listino.

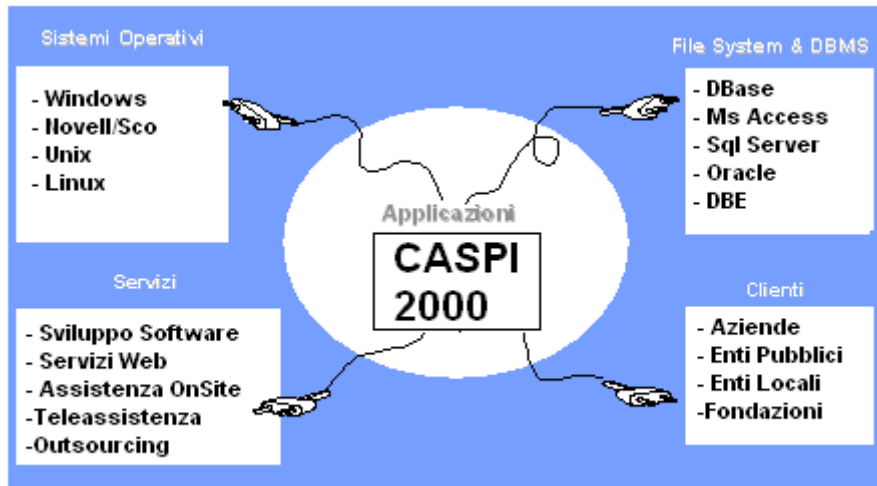
***CASPI 2000 Soc Coop A.r.l. è produttrice e distributrice del software applicativo che verrà descritto in questo documento.***



La caratteristica principale dei pacchetti è **l'integrazione** ottenuta come **risultato nativo** insito nelle applicazioni in quanto i gruppi di ricerca e sviluppo, responsabili dell'implementazione e della successiva evoluzione dei progetti software, operano tra loro in stretta collaborazione con l'obiettivo comune di produrre applicazioni che riusano, senza duplicarle, le informazioni già presenti nel sistema. Si ottiene, in questo modo, un notevole aumento del valore intrinseco della base informativa a disposizione del nostro cliente per il supporto alle decisioni strategiche.

I vantaggi che ne derivano sono essenzialmente riconducibili ad una maggiore sicurezza sull'attendibilità delle informazioni immesse in quanto si evitano le duplicazioni, conseguentemente si riducono gli errori e si ottiene un notevole risparmio di tempo; in una parola, i nostri applicativi si affiancano agli amministratori e cercano, con loro, di aumentare l'efficienza e l'efficacia della gestione.

La base di dati può essere adattata alle specifiche esigenze del cliente, rendendo così molto elastiche le nostre soluzioni. Infatti accade frequentemente che il cliente disponga già sia di hardware che di software di base su cui installare i nuovi applicativi oppure capita che vi siano delle preferenze su determinati prodotti per salvaguardare investimenti antecedenti o Know How interni.



Le applicazioni gestionali di tipo contabile hanno evidenziato come tutti i processi siano fortemente collegati ed integrati: ne consegue che il miglior strumento metodologico da adottare per elaborare una soluzione informatica integrata ed integrabile "nativamente" con altri sottosistemi, sia quello di ricorrere ad una progettazione effettuata con l'ausilio di un CASE. La tecnologia CASE (Computer Aided Software Engineering) svolge ormai un ruolo determinante in tutti i settori, industriali, amministrativi, gestionali. Il successo di questa tecnologia può essere misurato nel numero di applicazioni complesse ed altamente qualitative implementate. L'implementazione include l'ideazione, la stesura dell'analisi della base dati e delle relazioni tra essi ed infine la redazione dell'analisi visuale dei processi.

## **2. ARCHITETTURA DEL PACCHETTO**

Gli elementi di produttività rilevati hanno raggiunto elevate percentuali rispetto ai metodi di sviluppo tradizionali. Comunque ancora più significativi sono i guadagni di produttività in fase di macroanalisi o analisi dei progetti, senza sacrificare sia le prestazioni sia i tempi di realizzo.

La CASPI 2000 Soc Coop dalla sua costituzione utilizza piattaforme di sviluppo che comprendono sia strumenti CASE utilizzati per la parte di analisi e progettazione (upper case), sia quelli relativi alla generazione di codice (lower case). Gli strumenti vengono utilizzati coerentemente con quanto previsto dalle metodologie di controllo della qualità e per la gestione del progetto globale. I linguaggi di programmazione utilizzati sono di tipo Object Oriented, con codice nativo a 32 bit

### **2.1 Utilizzo di strumenti CASE**

Per ogni fase metodologica del progetto è previsto l'uso dello strumento automatico, dalla fase di pianificazione fino alla redazione delle procedure applicative attraverso un apposito linguaggio di programmazione; l'approccio alla realizzazione delle applicazioni è di tipo What You Think Is What You Get (WYTIWYG) ovvero "quello che pensi è quello che ottieni". Di seguito viene descritto in che modo potrà essere utilizzato il CASE nella progettazione del sistema informativo al fine di censire le funzioni ed automatizzare le necessarie attività per realizzare la macroanalisi dei processi e l'analisi visuale delle applicazioni da realizzare.

In dettaglio:

- Pianificazione strategica dell'informazione
- Costruzione e verifica di un modello dati e di un modello processi eseguiti
- Produzione di un prototipo completo di tutto quanto è richiesto dagli Utenti più esigenti. Help contestuale, manuale tecnico, manuale Utente e perfino l'autodimostrativo ed il corso tutorial per far apprendere all'Utente l'utilizzo dell'applicazione.

## 2.2 IL C.A.S.E. UTILIZZATO

La realizzazione del modello aziendale è fondamentale per lo sviluppo di un pacchetto applicativo e per la sua successiva installazione presso il sistema informativo del cliente. Il CASE dBsee ed i suoi TOOLS, offrono questa possibilità, generando dei supporti allo sviluppo che costituiscono un eccellente mezzo di comunicazione tra gli esperti di sistemi informativi e gli utenti dell'Ente.

- Definizione dell'applicazione ad alto livello (*upper CASE*), utilizzando formalismi grafici semplici ed intuitivi, per ottenere una visione globale del progetto. Per la definizione della base dati è disponibile un editor grafico per modelli *Entità-Relazioni*. La definizione dell'applicazione viene effettuata tramite *Progettazione Visuale*.
- Definizione dei dettagli dell'applicazione in modo interattivo, attraverso un insieme di strumenti WYSIWYG per l'impostazione di tabelle, menu, data entry, reports ecc.. (*lower CASE*).
- Disponibilità di un insieme di *oggetti predefiniti* (dialoghi, menu, data entry, browse, controls, reports ecc..) che possono essere utilizzati all'interno della propria applicazione e personalizzati o modificati a piacere.
- Possibilità di definire *nuove classi di oggetti* da utilizzare all'interno delle proprie applicazioni.
- Possibilità di generare automaticamente il codice sorgente utilizzando il repository di progetto creato con le fasi visuali di progettazione dell'Applicazione.
- *Gestione automatica della Multiutenza* nell'accesso ai dati, con controlli diretti e risoluzione automatica degli eventuali problemi su operazioni di dead-lock o di stallo informativo .
- *Modificabilità* del codice generato, con la possibilità di includere le modifiche effettuate nelle successive operazioni di generazione.
- Gestione di *dialoghi attivi*, in cui é possibile cambiare durante l'esecuzione lo stato del dialogo, modificandone l'aspetto o le azioni associate ai vari eventi.
- *Presentazione grafica dei dati*, tramite grafici o animazioni associati ai dati presentati a video.
- Interrogazioni tramite esempi (*Query By Example*), per massimizzare la facilità d'uso da parte dell'utente finale.

### ***2.2.1 Modellazione DATI (data modeling)***

Tool del CASE dBsee utilizzato per la costruzione delle aree dati e dei diagrammi entità-relazioni. Si possono inoltre rappresentare diversi livelli concettuali (come aree, dati, entità e sottotipi) in modo da rendere più facile la comprensione dei modelli dati agli utenti. E' possibile inoltre creare il layout del data-base fisico e tutte le sue vedute logiche (tabelle), utilizzato dai progettisti per ottimizzare la struttura della base dati ottenuta dal modello concettuale dei dati.

### ***2.2.2 Modellazione funzionale***

Tool del CASE dBsee utilizzato per rappresentare la scomposizione delle attività fino ad arrivare, partendo dal modello concettuale dei dati, all'individuazione delle funzioni primitive del sistema, con relativa creazione del layout del video. La peculiarità di questo processo risiede nel fatto che le varie funzioni sono direttamente collegate al repository delle informazioni gestite. Con tale tools, oltre a definire l'interfaccia screen delle transazioni on-line da realizzare, è possibile ipotizzare il disegno di report. Questo potente e flessibile strumento garantirà in sede di realizzazione delle procedure, l'integrità tra gli oggetti già individuati e presenti nel dizionario dati (moduli funzioni e dati), con le varie interfacce create (screen e report). L'output prodotto da questa fase, permette di simulare conversazioni on-line a livello di workstation prima di sviluppare gli applicativi o prima di dettagliare la definizione della logica dei controlli associati alle informazioni medesime.

### ***2.2.3. Generazione del Codice sorgente***

Tool del CASE dBsee utilizzato per generare il codice sorgente del progetto così costruito. dBsee fonda la sua architettura di generazione sul concetto di template; il template è il programma che dBsee esegue per generare a sua volta i programmi. Nel template sono infatti contenute tutte le istruzioni per indicare a dBsee come tradurre in sorgenti XBASE le definizioni del database, gli oggetti di data entry e tutte le altre componenti dell'applicazione memorizzate nel REPOSITORY di progetto. Quello che fa di dBsee un generatore di livello superiore è la possibilità che offre al programmatore di intervenire manualmente sui programmi generati senza che gli interventi manuali vengano persi nelle rigenerazioni successive. Il programmatore può infatti aggiungere parti di codice sorgente XBASE, controlli e routines personalizzate agendo con il proprio editor in punti ben precisi dei programmi generati chiamati "PUNTI DI INIEZIONE" e posizionati in punti tali da consentire al programmatore il pieno controllo del flusso del programma. È questa la prerogativa che rende dBsee particolarmente gradito ai programmatori esperti che si sentono così liberi di aggiungere proprie routine per svolgere funzionalità non previste dal sistema.

Inoltre nella fase di progettazione dell'applicazione per lo sviluppatore sono abilitate le seguenti funzioni:

- **Possibilità di definire controlli a livello di ogni Oggetto e campo di input:** Valore di default, controllo formale di validazione del dato immesso, campo calcolato, visualizzazione condizionale, edit condizionale, display automatico al variare di altri campi, controlli di congruenza fra i campi.
- **Sofisticato sistema di gestione del data entry.** L'editing dei dati viene effettuato mediante oggetti che possono essere composti fino ad un massimo di 99 pagine. In questo modo è possibile, con estrema semplicità, creare prospetti di data entry anche molto complessi e tali da soddisfare le esigenze dell'analista più esigente; il flusso fra le pagine è continuo o pilotabile via mouse o tastiera.
- **Gestisce un numero illimitato di relazioni:** Uno a Molti, Molti a Uno, Uno a Uno e Molti a Molti. Ad ogni ramo relazionale è possibile associare un attributo di INTEGRITÀ REFERENZIALE in modo da controllare il comportamento del sistema quando vengono creati o cancellati record di file in relazione con altri. Le modalità supportate sono: CASCADE, RESTRICT, NULLIFY, ORPHAN e PREVENT.
- **Librerie esterne** dBsee genera le applicazioni in formato sorgente XBASE, per cui è del tutto normale che si possa fare uso di tutte le librerie prodotte dal mercato delle terze parti che hanno fatto di XBASE un prodotto di successo. È inoltre possibile richiamare programmi XBASE già realizzati in precedenza.
- **Generazione e Template Funzionalità in rete locale** Le applicazioni generate da dBsee sono perfettamente funzionanti in rete locale. Tutti i controlli di integrità referenziale, gestione delle chiavi primarie univoche, creazione di codici automatici progressivi e ricostruzione indici sono generate per funzionare correttamente in un contesto di rete locale. Una sofisticata gestione di semafori consente inoltre di risolvere particolari problemi di conflitti in multiutenza. Sono inoltre previste speciali funzionalità di time out che consentono di sbloccare automaticamente un record dopo un determinato tempo di inattività della stazione e di aggiornare automaticamente la visualizzazione quando viene modificato da un'altra stazione di lavoro.