

**Standard datawarehouse**  
**Indicazioni per lo sviluppo e realizzazione di progetti**

## INDICE

<b>1. INTRODUZIONE.....</b>	<b>3</b>
1.1 DESTINATARI DEL DOCUMENTO.....	4
1.2 SCOPO DEL DOCUMENTO.....	4
1.3 FONTI.....	4
<b>2. INDICAZIONI PER LO SVILUPPO E LA REALIZZAZIONE DI PROGETTI IN ARCHITETTURA DI DATA WAREHOUSE.....</b>	<b>5</b>
2.1 IL CONTESTO.....	5
2.2 LA MODELLAZIONE DEI DATI IN AMBIENTE DECISIONALE.....	5
2.3 PROCESSO DI COSTRUZIONE DI UN DATA MART.....	11
2.4 GLI STRUMENTI.....	12
2.5 DELIVERY DI PROGETTO .....	12
<b>3. ARCHITETTURA SOFTWARE PER LO SVILUPPO DI PROCESSI ETL CON INFORMATICA POWER CENTER.....</b>	<b>16</b>
3.1 HARDWARE E SOFTWARE.....	16
3.2 SCHEMA GUIDA PER L'ASSEGNAZIONE DEI PROFILI UTENTE .....	17
3.3 MIGRAZIONE DEI PROCESSI ETL NEI VARI AMBIENTI .....	18
3.4 SETUP DELLE WORKSTATION CLIENT .....	20
3.5 MANUTENZIONE DEI REPOSITORY .....	20
<b>4. ARCHITETTURA BUSINESS OBJECTS/WEBINTELLIGENCE.....</b>	<b>22</b>
4.1 SISTEMA CLIENT/SERVER .....	22
4.2 SISTEMA WEBINTELLIGENCE.....	23
4.2.1 Azioni utente .....	24
4.2.2 Dimensionamento.....	25
4.2.3 Partizionamento .....	25
4.2.4 Il Clustering di WebIntelligence.....	26
4.3 BROADCAST AGENT .....	27
4.4 DIMENSIONAMENTO DEI SERVER WEBINTELLIGENCE .....	28
4.5 CONFIGURAZIONE DEL REPOSITORY BUSINESSOBJECTS/WEBINTELLIGENCE .....	30
4.6 SCHEMA GUIDA PER L'ASSEGNAZIONE DEI PROFILI UTENTE .....	31
4.7 MIGRAZIONE NEI VARI AMBIENTI.....	32
4.8 SETUP DELLE WORKSTATION FULL CLIENT.....	33
4.9 GESTIONE DELLE VERSIONI.....	33
4.10 MANUTENZIONE .....	34
4.11 REQUISITI DI VERSIONI E COMPATIBILITÀ DEL SW .....	34
<b>5. SCHEDULAZIONE TRAMITE OPC .....</b>	<b>36</b>
5.1 PROCEDURE ETL CON SORGENTE BASE DATI DB2.....	36

## 1. Introduzione

Il presente documento fa parte della collezione di documenti “data warehouse”, che raccoglie le indicazioni specifiche delle varie componenti sotto un unico indirizzo, separando al contempo le varie tematiche, al fine di proporre una più agevole lettura e meglio indirizzare le necessità di aggiornamento dei singoli documenti.

Tale collezione è composta dai seguenti documenti:

### Architettura Data warehouse

Descrive i concetti base del Data Warehouse e sottolinea le differenze di obiettivi e caratteristiche rispetto ai sistemi di tipo gestionale. E' è orientato soprattutto alla definizione di standard tecnologici, pur delineando brevemente anche gli aspetti di approccio metodologico legati allo sviluppo di progetti di Data Warehouse.

Per le tre componenti acquisizione di dati, gestione e distribuzione sono descritti i requisiti e le funzionalità richieste, e vengono inoltre indicati anche i prodotti di mercato che meglio rispondono alle esigenze e si integrano con il sistema informativo del Ministero.

### Indicazioni per lo sviluppo e realizzazione di progetti di Data warehouse

E' il presente documento.

### Il disegno degli ambienti

Il documento descrive gli ambienti tecnologici previsti per ospitare sia l'Enterprise Data Warehouse sia i singoli progetti di Data Mart.

### **1.1 *Destinatari del documento***

Il documento è rivolto in modo particolare ai responsabili dei progetti di Data Mart ed intende rappresentare un ausilio concreto per coloro che partecipano alle singole iniziative progettuali del Ministero e non una linea guida teorica

### **1.2 *Scopo del documento***

Lo scopo del presente documento è quello fornire indicazioni per affrontare lo sviluppo e la realizzazione delle iniziative di natura informativa in corso presso il Ministero in modo integrato e uniforme, in coerenza con le scelte architettureali e con gli ambienti esistenti presso il Ministero.

### **1.3 *Fonti***

Le scelte tecnologiche e metodologiche indicate nel presente documento tengono conto dei sistemi esistenti nei diversi Dipartimenti, che rappresentano dei vincoli in termini di operatività.

Si sono valutate inoltre le indicazioni tecnologiche e metodologiche dell'AIPA, di osservatori del mercato dell'Information Technology (Gartner Group, Ovum, Dataquest) e di società di consulenza (Technology Transfer).

## **2. Indicazioni per lo sviluppo e la realizzazione di progetti in architettura di Data Warehouse**

### **2.1 Il contesto**

La presenza delle diverse iniziative in essere nell'ambito dei Dipartimenti porta a considerare opportuno l'approccio "incrementale". Tale approccio consiste nella definizione di un Enterprise Data Warehouse come risultato di un percorso iterativo che prevede la realizzazione di Data Mart, secondo le priorità che verranno dettate dall'Amministrazione.

La integrazione dei diversi progetti è garantita dalla definizione, a cura Consip, di un modello informativo comune, che assicura la condivisione della semantica, la consistenza e la qualità dei dati.

Il modello informativo comune si arricchirà in modo iterativo dei modelli dati dei singoli progetti di data mart centralizzandoli in un unico repository.

Nel caso in cui il fabbisogno informativo di un nuovo progetto di data mart, esaminato in dettaglio nella fase di analisi, sia già stato oggetto di acquisizione dai sistemi legacy da parte di altri progetti, se ne verificherà, assieme alle persone Consip, il riutilizzo direttamente a partire dall'Enterprise Data Warehouse o dalle aree di staging. Per la parte di fabbisogno non ancora disponibile, l'analisi porterà all'acquisizione del nuovo dato dai sistemi sorgente e rappresenterà un delta nel modello dell'Enterprise Data Warehouse.

Ogni singola iniziativa di data mart avrà quindi un collegamento al progetto di costruzione del Data Warehouse Integrato del Ministero, contribuendo a renderlo completo in modo iterativo.

### **2.2 La modellazione dei dati in ambiente decisionale**

La tipicità dei progetti di Data Warehouse, rispetto alle tradizionali applicazioni gestionali, si riflette anche in una diversa tecnica di modellazione dei dati.

In ambiente decisionale i dati vengono organizzati in modo da rispondere direttamente alle interrogazioni tipiche dell'utente finale.

Ipotizziamo, ad esempio, che l'obiettivo dell'utente sia quello di monitorare nel tempo la pianificazione e la realizzazione dei progetti di investimento che utilizzano finanziamenti pubblici, localizzati in varie aree del territorio nazionale, legati a diversi settori di attività.

In maniera semplificata, si può immaginare di rappresentare questa attività attraverso un cubo, i cui lati riproducono le variabili tempo, località e classificazione di attività,

relativamente ai progetti. Ogni punto all'interno del cubo è l'intersezione delle coordinate definite dai lati del cubo stesso e rappresenta la misura del fatto di interesse (il progetto e il relativo finanziamento ) per quella particolare combinazione di località, settore di attività, tempo, che rappresentano le dimensioni dell'analisi.

In ambito Data Warehouse, si parla quindi di modellazione dimensionale dei dati.

Il modello dimensionale e quello tradizionale entity-relationship sono potenzialmente in grado di memorizzare le stesse informazioni. La differenza è legata alla ricerca di prestazioni ottimali sulle due tipologie di ambienti.

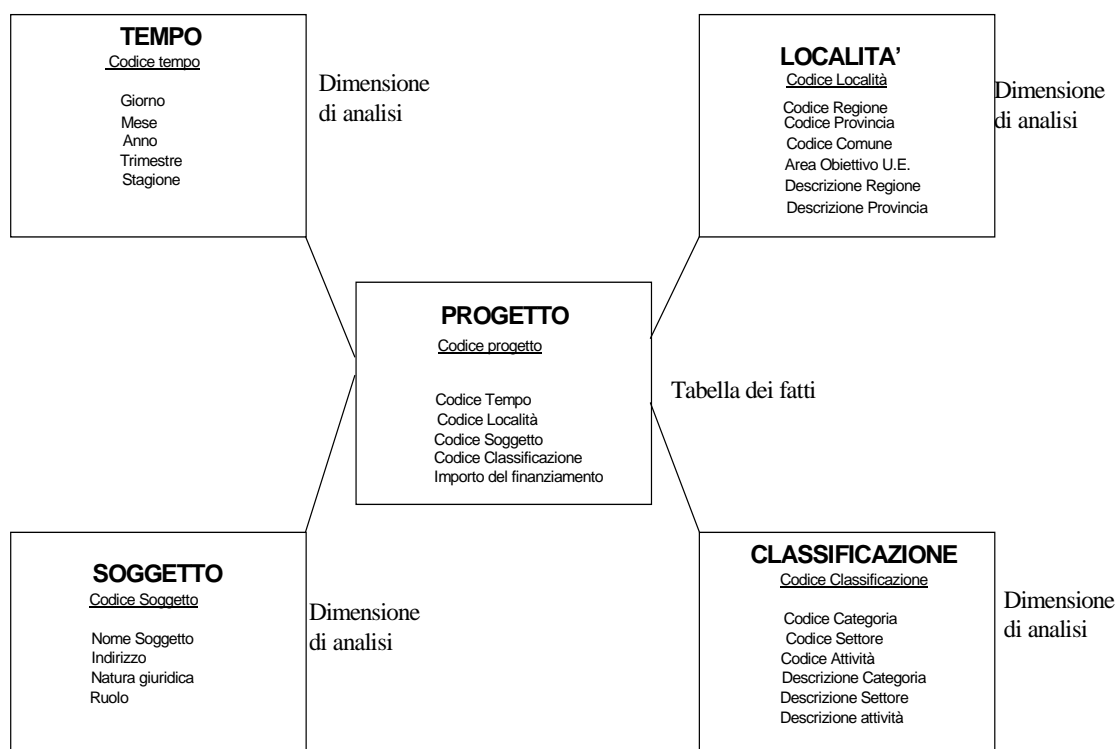
Negli ambienti operazionali è fondamentale adottare tecniche di normalizzazione che, eliminando le ridondanze, permettono ad un elevato numero di transazioni ripetitive di interrogare ed aggiornare le informazioni in un singolo punto del database.

I sistemi di supporto alle decisioni, invece, sono caratterizzati da un ridotto numero di transazioni, che operano però in maniera trasversale sulla base dati, richiedendo l'accesso in lettura su un elevato numero di dati da mettere in join. L'uso di tecniche di normalizzazione appare quindi controproducente nella progettazione di un sistema di Data Warehouse. In ambiente decisionale è più corretto disegnare il database in modo da soddisfare in maniera diretta e semplificata le esigenze specifiche degli utenti finali, anche attraverso dati denormalizzati e aggregati.

E' importante dedicare molta attenzione all'analisi del modello concettuale, prevedendo tutte le dimensioni di analisi e per ciascuna dimensione i relativi attributi. Gli attributi delle tabelle dimensionali vengono usati direttamente come condizioni di ricerca nelle interrogazioni sul Data Warehouse e come intestazioni delle colonne nei report degli utenti finali. L'uso di dati preaggregati e di sintesi limita il numero di operazioni necessarie a ricostruire l'informazione richiesta.

L'implementazione tipica del modello dimensionale è quella che risulta nello schema dati a stella o star schema.

## STAR SCHEMA



Lo star schema, oltre a essere ottimizzato per le interrogazioni, presenta l'ulteriore vantaggio di consentire una rappresentazione grafica del business più comprensibile per l'utente finale.

A differenza dello schema entità-relazioni, in cui tutte le entità sono rappresentate nello stesso modo, lo star schema è asimmetrico. Al centro del diagramma c'è una tabella dominante di grandi dimensioni, che è l'unica a possedere collegamenti multipli, che si concretizzano in campi chiave in essa contenuti, alle altre tabelle. Le altre tabelle rappresentano le dimensioni di analisi attraverso cui si esamina il "fatto" di interesse.

Tecnicamente si definiscono "fact table" la tabella centrale e "dimension table" le altre.

La fact table è la tabella in cui vengono memorizzate le misure numeriche del fatto (nell'esempio l'importo del finanziamento del progetto di investimento), ognuna delle quali rappresenta l'intersezione di tutte le dimensioni e quantifica una o più variabili. Normalmente è l'unica tabella normalizzata.

Le altre tabelle rappresentano le dimensioni di analisi attraverso cui si esamina il "fatto" e contengono attributi di tipo descrittivo. La cardinalità di queste tabelle, normalmente denormalizzate, è limitata.

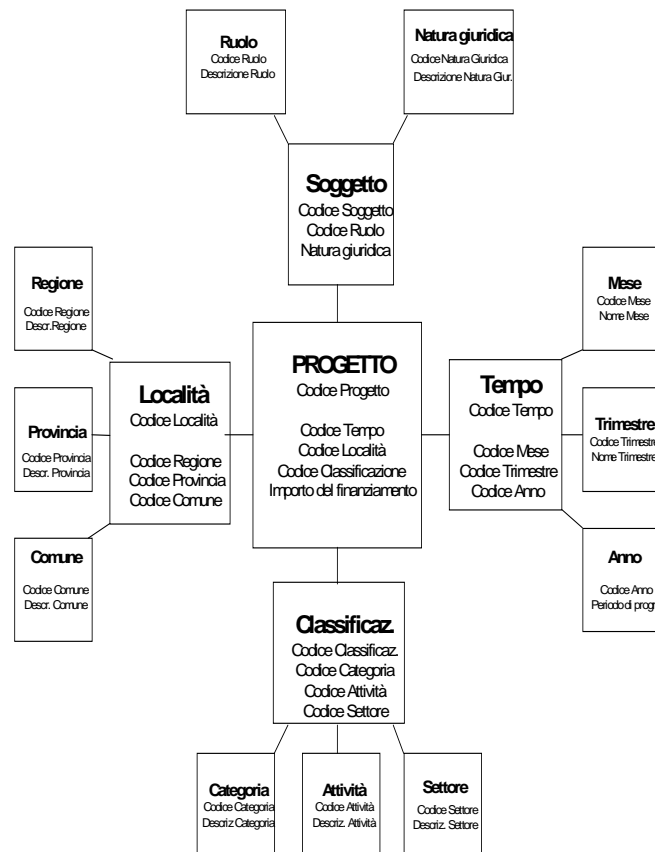
Tra le tabelle dimensionali, la tabella della dimensione tempo è sempre presente nei modelli dimensionali, quale che sia il business da rappresentare. Questo deriva dalla caratteristica dei sistemi di Data Warehouse che gestiscono anche le informazioni storiche.

A volte, per migliorare ulteriormente la interpretabilità del modello, anche le tabelle dimensionali vengono normalizzate, esplicitando le gerarchie. Questo dà luogo ad uno schema detto a "fiocco di neve" o Snowflake Schema.

+



## SNOW FLAKE SCHEMA



La navigazione dei dati attraverso il modello snowflake è meno efficiente che nello Star Schema perchè aumenta il numero di tabelle su cui effettuare le join.

Va considerato che, se da un lato la denormalizzazione garantisce migliori prestazioni nelle operazioni di interrogazione, la stessa rende più oneroso l'aggiornamento e il caricamento dei dati. Più in generale, le operazioni di manutenzione del modello dati multidimensionale, quali l'estensione a nuove dimensioni d'analisi, richiede il caricamento "ex novo" della base

dati. Per questo, di volta in volta, occorre valutare il trade-off tra le due esigenze e adottare a seconda dei casi lo Star Schema o lo Snowflake Schema.

Va inoltre aggiunto che, per modelli molto complessi, con diverse fact table che condividono numerose dimensioni, diventa difficile adottare la tecnica dimensionale, anche perchè non è sempre chiaro quali siano le tabelle dei fatti e quali quelle delle dimensioni.

Per il disegno concettuale dell'Enterprise Data Warehouse appare preferibile, quindi, un modello dati Entity-Relationship opportunamente denormalizzato.

La modellazione dimensionale è invece senz'altro più idonea per la rappresentazione dei dati dei Data Mart.

I modelli dimensionali possono essere implementati sia su DBMS relazionali sia su specifici DBMS multidimensionali.

Le indicazioni fornite in questo paragrafo, lungi dall'essere esaustive, hanno il solo scopo di sottolineare le specificità dei progetti di Data Warehouse anche rispetto alle tecniche di modellazione dei dati.

Per maggiori approfondimenti, relativi alla modellazione dei dati, sia a livello di modello Enterprise sia a livello di Data Mart, si rimanda ai noti trattati di modellazione dimensionale (Ralph Kimball - "The Data Warehouse Lifecycle Toolkit" - Wiley- 1998; W.H. Inmon - "Building the Data Warehouse" - Wiley- 1996)

Per il disegno dei dati è indispensabile utilizzare un tool che consenta di rappresentare e documentare in modo standard i modelli dati sia a livello concettuale sia a livello logico-fisico.

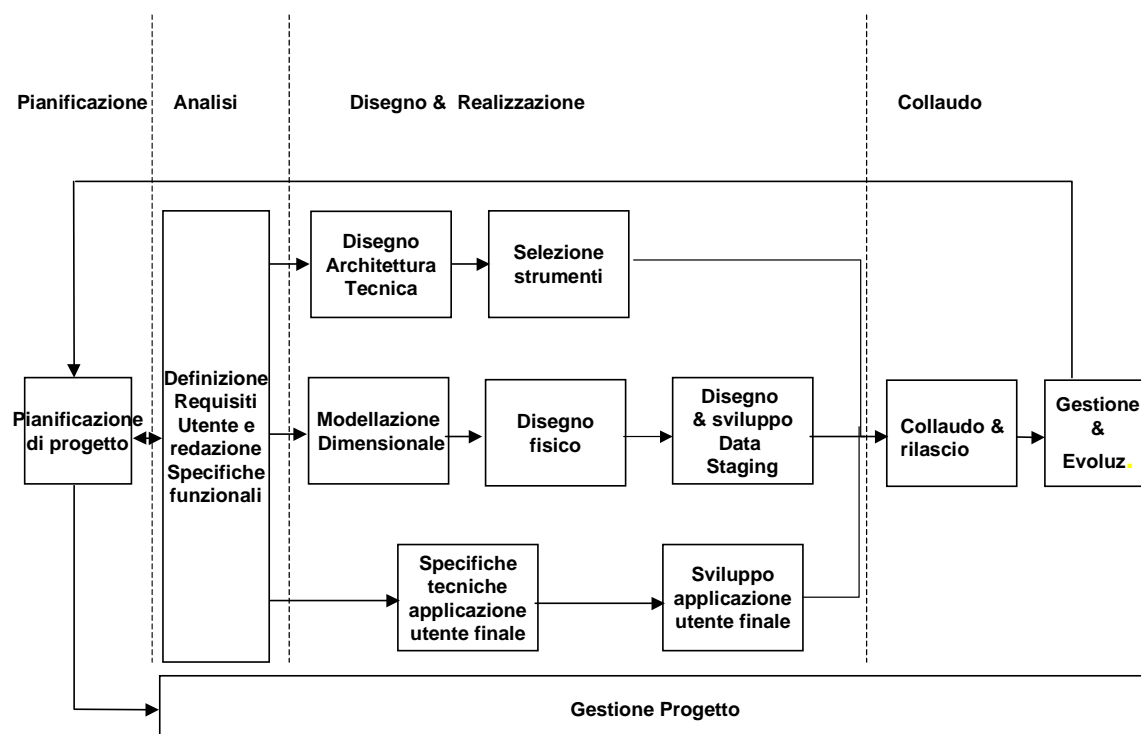
I tool consentono di generare automaticamente le definizioni delle strutture fisiche (DDL), in modo ottimizzato per il DBMS target. I tool garantiscono infine la possibilità di recuperare il patrimonio informativo esistente a partire dalle strutture fisiche attraverso l'attività di reverse engineering.

I tool attualmente selezionati per il Ministero, tra quelli disponibili sul mercato, sono: Erwin (C.A.), Cool:Biz (Sterling Software), Cool: Business Team + Cool:DBA (Sterling Software), Designer 2000(Oracle) solo se seguito dallo sviluppo con la suite Oracle.

## 2.3 Processo di costruzione di un Data Mart

Il processo produttivo per la costruzione di un Data Mart non differisce, nelle fasi, da un qualunque progetto di sviluppo, avendo alcune particolarità solamente per contenuti specifici di alcuni documenti (ad esempio star schema del Data Mart, ecc.), utilizzo di strumenti specializzati (ETL e così via), utilizzo frequente di prototipi. Pertanto le modalità di gestione del progetto, (pianificazione, controlli di qualità, verifiche ed approvazioni,...) saranno le stesse normalmente utilizzate.

Lo schema seguente è una vista “grafica” del ciclo di sviluppo ricavata da Ralph Kimball - “The Data Warehouse Lifecycle Toolkit” - Wiley- 1998; fig.2.1:



che peraltro è facilmente mappabile sulle fasi previste dagli attuali contratti in essere.

In tale schema le fasi e le attività individuate per il processo di costruzione di un Data Mart non procedono in maniera strettamente sequenziale bensì sono possibili ricicli tra esse.

Ad esempio la fase di “Analisi dei requisiti utente” consiste nell’individuazione delle esigenze dell’utente e viene svolta attraverso interviste e contatti con l’utente e l’esame della documentazione già esistente relativa all’area di interesse.

Le informazioni utili all’utente e i relativi modi di visualizzazione individuati in questa fase sono di input alla successiva fase di disegno e realizzazione.

Una stretta sequenzialità fra le due fasi non può sussistere, in quanto, nello svolgimento della progettazione, possono verificarsi le seguenti situazioni:

- ulteriori contatti con l’utente che chiariscono i requisiti;

- individuazione di nuove o diverse informazioni proponibili all'utente, scaturite dall'analisi dei dati disponibili in azienda;
- definizione di diverse modalità di visualizzazione, in dipendenza dagli strumenti tecnologici disponibili.

La raccolta dei requisiti, quindi, può continuare a verificarsi anche nelle fasi successive. Tuttavia, quando è consolidato anche se non definitivamente, l'insieme di requisiti, il ciclo progettuale prosegue con le fasi successive.

In tale fase può essere molto utile produrre un **prototipo**, al fine di mostrare la tipologia di funzioni e dati e le modalità di navigazione che si intendono implementare. Attraverso un ciclo di revisioni del prototipo si giunge a stabilire e a validare in maniera precisa i requisiti dell'utente.

A fronte di variazioni dei requisiti già concordati, si incrementa di volta in volta un unico documento contenente la proposta di automazione aggiornata, emettendo versioni successive per ciascuna iterazione. In caso di modifiche ai requisiti iniziali, infatti, è necessaria una nuova fase di verifica da effettuarsi sulla documentazione aggiornata.

In fase di Analisi dei Requisiti si aggiorna e completa il Piano di progetto, in particolare pianificando le sedute di "*Riesame della Progettazione*", da svolgersi nella successiva fase di "Disegno".

## **2.4 Gli strumenti**

Per semplicità vengono qui riassunti gli strumenti adottati in ambito data warehouse. Per una completa trattazione si rimanda al documento di architettura.

Sistema operativo EDW: Unix

Sistema operativo DM: Unix o Windows NT

Data base : Oracle

Acquisizione dati (ETL): Informatica Power Center

Accesso ai dati: preferibilmente Business Objects e Oracle Discoverer

Modellazione dati: Erwin, Cool:Biz, Cool:DBA + Business Team, Designer(solo se seguito da catena di sviluppo Oracle)

## **2.5 Delivery di progetto**

Di seguito sono indicate le informazioni, relative solamente alla parte dati, che devono essere prodotte nell'ambito dei documenti di progetto e viene esposto uno schema che focalizza gli aspetti di Staging area, Enterprise Data Warehouse, Data Mart.

Nell'esposizione viene utilizzato il termine "schema concettuale", la cui definizione è riportata più avanti.

	ANALISI	DISEGNO & REALIZZAZIONE
Staging Area	<ul style="list-style-type: none"> <li>• Bozze dei prospetti</li> <li>• Modello Concettuale, in forma di schema E/R, contenente tutte le entità e gli attributi necessari per elaborare le misure richieste,</li> <li>• Descrizioni delle entità e degli attributi contenuti nello schema E/R di progetto.</li> <li>• Modalità di storicizzazione dei dati.</li> <li>• Individuazione della granularità necessaria.</li> <li>• Individuazione dei sistemi sorgente e delle relative modalità/volumi di alimentazione.</li> <li>• Individuazione delle porzioni degli schemi logico/fisico dei dati dei sistemi sorgente che sono riconducibili ad oggetti contenuti nello schema E/R di progetto.</li> <li>• Descrizione di tabelle/colonne/relazioni ove mancanti.</li> <li>• Mappatura tra gli schemi logico/fisico e lo schema E/R di progetto (tutte le entità/attributi devono essere mappate a uno o più tabelle/colonne).</li> <li>• Risoluzione delle problematiche di integrazione (semantiche, sintattiche e di contenuti) tra gli schemi logico/fisici di origine</li> <li>• Schema logico unico di origine.</li> </ul>	<ul style="list-style-type: none"> <li>• Definizione delle caratteristiche tecniche (tabelle, file) dello schema fisico dei dati da implementare nella Staging Area e generazione di dette strutture.</li> <li>• Configurazione del tool ETL per il caricamento delle strutture di cui sopra.</li> <li>• Preparazione dei necessari ambienti operativi.</li> </ul>
Enterprise Data Warehouse	<ul style="list-style-type: none"> <li>• Schema E/R del progetto (da integrare nell' EDWH)</li> <li>• Glossario dei dati di progetto (da integrare nell'EDWH).</li> <li>• Schema logico dell' EDWH.</li> <li>• Mappatura tra le strutture dello schema logico unico di origine e quelle contenuti nello schema logico del EDWH.</li> <li>• Definizione delle strategie di estrazione, trasformazione e caricamento dell' EDWH a partire dallo schema logico unico di origine.</li> <li>• Definizione delle modalità di trattamento della dimensione tempo.</li> </ul>	<ul style="list-style-type: none"> <li>• Definizione delle caratteristiche tecniche dello schema fisico dei dati da implementare come EDWH e generazione di dette strutture.</li> <li>• Configurazione del tool ETL per il caricamento delle strutture di cui sopra a partire dall' Area di Staging.</li> <li>• Preparazione dei necessari ambienti operativi.</li> </ul>

Data Mart	<ul style="list-style-type: none"> <li>• Individuazione di Fatti e Dimensioni di interesse per il progetto.</li> <li>• Modello Concettuale del DM sotto forma di Star Schema o Snowflake.</li> <li>• Schema logico</li> <li>• Regole di calcolo delle misure.</li> <li>• Modalità di trattamento della dimensione tempo.</li> <li>• Glossario del Data Mart.</li> <li>• Strategie di estrazione, trasformazione e caricamento del DM</li> <li>• Modalità di interfacciamento col DM.</li> </ul>	<ul style="list-style-type: none"> <li>• Definizione delle caratteristiche tecniche dello schema fisico dei dati da implementare come DM e generazione di dette strutture.</li> <li>• Realizzazione funzionalità di accesso al DM</li> <li>• Configurazione del tool ETL per il caricamento delle strutture di cui sopra a partire dall' EDWH</li> <li>• Preparazione dei necessari ambienti operativi..</li> </ul>
-----------	---	---

Uno **schema concettuale** deve avere i seguenti requisiti affinché possa garantire un livello di qualità adeguato (C.Batini, G. De Petra, M. Lenzerini, G. Santucci - "La progettazione concettuale dei dati" - Franco Angeli - 1999).

- 1) **Completezza**, lo schema deve rappresentare tutte e sole le proprietà dei concetti della realtà di interesse descritte nei requisiti Utente;
- 2) **Correttezza**, le strutture del modello devono essere utilizzate nello schema in modo conforme al significato con cui sono definite nel modello;
- 3) **Minimalità**, ogni aspetto della realtà d'interesse deve essere rappresentato una sola volta nello schema;
- 4) **Leggibilità**, lo schema deve essere rappresentativo della realtà di interesse in modo comprensibile ed autoesplicativo;
- 5) **Consistenza**, lo schema concettuale deve verificare tutti i vincoli definiti su di esso.

Parte integrante dello schema concettuale è il **glossario**, inteso come raccolta delle definizioni dell'applicazione, delle entità coinvolte nello schema concettuale e dei loro attributi. Il glossario deve considerare almeno le informazioni riguardanti nome esplicito breve, descrizione e sinonimi associati.

La fase di raccolta e formalizzazione delle esigenze utente, per brevità compresa nella fase di analisi dello schema precedente, viene considerata d'importanza primaria per eseguire il controllo formale e sostanziale di tutte le fasi successive, in particolare per l'approvazione delle specifiche funzionali. Tale formalizzazione dei requisiti, che può essere un autonomo documento o può essere inserito nell'ambito delle "specifiche funzionali", prodotte durante la fase di analisi, dovrà contenere un primo schema concettuale nel quale sarà sufficiente individuare solo le principali informazioni, ossia fermarsi all'individuazione delle "classi di informazioni", e non sarà necessario descrivere in maniera completa gli attributi, limitandosi a segnalare quelli rilevanti nella realtà d'interesse.

Nella fase di analisi relativamente all'Enterprise Data Warehouse (che deve essere intesa come EDW specifico del singolo progetto) deve essere presentato uno schema concettuale ottenuto per approssimazioni successive; partendo da quello iniziale (redatto durante l'analisi dei requisiti) si potranno realizzare schemi sempre più accurati e completi fino allo schema conclusivo esplicativo della realtà di interesse (presentato nella specifica funzionale). Lo schema concettuale dovrà essere integrativo delle eventuali sotto-aree individuate o, relativamente alla complessità in esso espressa, potrà essere costituito da una raccolta di schemi, purché sia leggibile la realtà interpretata. Allo schema concettuale sarà sempre associato un glossario dati in cui verranno raccolte tutte le descrizioni delle entità e degli attributi espressi nelle modalità suddette.

Per il disegno concettuale dell'Enterprise Data Warehouse appare preferibile un modello dati Entity-Relationship.

Nella fase di progettazione del Data Mart lo schema concettuale sarà in formato star schema e il glossario associato, strutturato come il precedente, sarà caratterizzato dal contenere le definizioni delle "dimensioni" e delle "misure". Queste saranno costituite da entità già trattate dove anche le nuove entità e i nuovi attributi saranno ottenuti da queste per derivazione o aggregazione; in questi casi si dovranno descrivere brevemente le operazioni fatte.

### 3. Architettura software per lo sviluppo di processi ETL con Informatica Power Center

Le informazioni di seguito descritte dettagliano sia attività che coinvolgono i gruppi di sviluppo, sia attività riservate all'amministratore del sistema. Si è comunque scelto di descrivere anche le attività non di competenza dei gruppi di sviluppo al fine di consentire una migliore comprensione dell'intero flusso operativo.

#### 3.1 Hardware e software

Di seguito viene descritta l'architettura prevista sul lato hardware, software ed inoltre viene approfondita l'organizzazione dei Repository di PowerCenter per i vari ambienti previsti.

##### Hardware

Sono stati previsti due server UNIX distinti, uno per l'ambiente di sviluppo e collaudo dei processi ETL, ed uno per l'ambiente di esercizio, entrambi collegati in rete con gli sviluppatori e i responsabili dei progetti.

##### Software

- *Server di Sviluppo/Collaudo*

**Dati Target:** Dovranno essere previsti 2 schema Oracle per i Data Mart di ogni progetto, uno per l'ambiente di sviluppo e uno per l'ambiente di collaudo; dovranno essere previsti inoltre 2 schema Oracle per gli ambienti di sviluppo e collaudo per l'Enterprise Data Warehouse.

**Repository:** Saranno creati 1 schema differente per ogni repository di progetto, sia esso per la realizzazione di un Data Mart, sia per l'EDW.

Gli schema creati per i repository ed i target dovranno essere acceduti tramite un singolo utente Oracle che non avrà nessuna visibilità sugli altri schema.

I vari schema dovranno essere organizzati in tre istanze Oracle differenti: una per tutti i repository, una per i dati target dei Data Mart dei progetti, ed uno per i dati target dell'EDW, che preveda inoltre uno schema per il Data Staging

**PowerCenter Server:** dovrà essere installato un PowerCenter Server per ogni progetto, avendo cura di utilizzare differenti utenti UNIX, di effettuare l'installazione del software su differenti directory (da mantenere non visibili agli altri utenti) ed infine di configurare il server perché possa operare su porte TCP/IP differenti (tramite personalizzazione del parametro **ServerPort** del file di configurazione **pmserver.cfg**) onde assicurarsi che ogni repository abbia a disposizione il suo specifico processo server per l'esecuzione delle sessioni ETL.

**Organizzazione interna dei repository:** Ogni repository relativo all'ambiente di sviluppo conterrà due folder differenti, uno per lo sviluppo vero e proprio dei mapping, ed il secondo utilizzato per il passaggio in ambiente di collaudo, dove verranno copiati i singoli mapping precedentemente testati. In ambiente di collaudo, ogni repository conterrà un solo folder.



All'interno dei repository di sviluppo e di collaudo, per ogni progetto, dovranno essere codificate le opportune **Database Connection**, che dovranno avere stesso nome (per agevolare il passaggio di ambiente delle sessioni ETL) ma diverse proprietà in quanto riferimenti a schema target e sorgenti differenti.

- *Server di Esercizio*

**Dati Target:** Dovrà essere previsto uno schema Oracle per i Data Mart di ogni progetto ed uno per l'EDW.

**Repository:** Sarà creato uno singolo schema per il repository di esercizio di PowerCenter. Gli schema creati per il repository ed i target dovranno essere acceduti tramite un singolo utente Oracle che non avrà nessuna visibilità sugli altri schema.

**PowerCenter Server:** l'ambiente di esercizio prevede un solo repository di PowerCenter, quindi sarà installato un singolo server di PowerCenter.

**Organizzazione interna dei repository:** Ogni repository conterrà un solo folder per progetto di Data Mart per l'esecuzione dei mapping in esercizio.

Le Database Connection, siano esse applicate come source o come target all'interno delle sessioni ETL, dovranno avere lo stesso nome di quelle codificate sugli ambienti di sviluppo e collaudo.

### **3.2 Schema Guida per l'assegnazione dei profili utente**

Sono stati individuati i seguenti profili utenti per gli ambienti di sviluppo, collaudo ed esercizio:

- **Sviluppatore**

Lo sviluppatore avrà i tutti i permessi sui folder negli ambienti di sviluppo e collaudo, ma non avrà nessun privilegio di amministrazione del repository

- **Utente Produzione**

Avrà gli stessi permessi e privilegi di uno sviluppatore, ma in più dovrà avere il permesso di scrittura sul folder del repository di esercizio; l'utente sarà sostanzialmente l'unico sviluppatore del progetto ad avere l'autorità per portare in esercizio le sessioni ETL.

- **Operatore**

L'operatore sarà il profilo utilizzato per mantenere, schedare, eseguire e, eventualmente, modificare le sessioni ETL in ambiente di esercizio. Esso avrà quindi pieni permessi sul folder del repository di esercizio, ma non avrà i privilegi per poter modificare i mapping o amministrare il repository.

- **Super User**

Il Super User è il profilo di amministrazione di tutti i repository; esso avrà il privilegio di super user, che automaticamente dà pieni privilegi e permessi sul repository.

Il primo passo da compiere è quindi quello di individuare gli utenti di progetto e codificarli, con uguale nome utente e password, su ciascuno dei tre repository di progetto (sviluppo, collaudo, esercizio); fatto questo occorrerà definire i profili (user group) per ogni repository di progetto ed assegnare i permessi relativi ai folder ed eventuali privilegi di amministrazione.

La tabella di seguito riassume permessi sui folder e privilegi per ogni profilo utente codificato:

Profilo	Folder di Sviluppo	Folder di Collaudo	Folder di Esercizio	Privilegi
Sviluppatore	RWX	RWX	R	Use Designer Browse Repository Create Sessions and Batches
Utente Produzione	RWX	RWX	RW	Use Designer Browse Repository Create Sessions and Batches
Operatore			RWX	Create Sessions and Batches
Super User	RWX	RWX	RWX	Tutti

Verificata l'assegnazione dei privilegi ai profili, si può procedere all'assegnazione dei profili agli utenti definiti.

### 3.3 Migrazione dei processi ETL nei vari ambienti

La struttura degli ambienti ripartiti su diversi repository, precedentemente illustrata, rende già di per sé le operative di migrazione dei processi ETL ampiamente semplificata; di seguito vediamo le operative per i passaggi tra i vari ambienti:

- Passaggio da sviluppo a collaudo**  
Una volta conclusa la parte di sviluppo vera e propria, la quale avviene sul folder predefinito per lo sviluppo, i mapping pronti per il passaggio a collaudo vengono copiati **manualmente** all'interno del secondo folder, che dovrà contenere solo e solamente mapping, strutture sorgenti e strutture target pronti per il collaudo. A questo punto, il passaggio all'ambiente di collaudo avviene tramite la semplice operativa di **copia** del folder dal repository di sviluppo a quello di collaudo.

Tale operativa può essere effettuata da un qualsiasi membro del gruppo degli sviluppatori.

- **Passaggio da collaudo ad esercizio**

Il passaggio all'ambiente di esercizio viene effettuato tramite **copia** del folder dal repository di collaudo a quello di esercizio; tale operativa è effettuabile in questo caso solamente dall'utente di progetto al quale sia stato assegnato il profilo di Utente Produzione, unico profilo che abbia i permessi di scrittura sul folder del repository di esercizio.

### **3.4 Setup delle Workstation Client**

Di seguito le operative di setup da eseguire per ogni Workstation client di sviluppo di un progetto:

- **Installazione di PowerCenter client**
- **Installazione dei driver ODBC forniti da Informatica**
- **Installazione di PowerConnect for DB2 client**
- **Import dei registry del repository**

All'installazione della prima workstation client, bisognerà creare, o censire, se già creati, i repository di progetto per gli ambienti di sviluppo, collaudo ed esercizio; le informazioni codificate in questa prima installazione (Nome del repository, utenza/password dello schema database, utenza/password per il repository, nome del Data Source ODBC) possono venire facilmente esportate tramite le funzionalità di import dei registry del repository del Repository Manager di PowerCenter, nonché riutilizzate nelle successive installazioni, settando così automaticamente tutte le informazioni relative alla connettività ai repository.

**N.B.** Preventivamente a questa operativa bisognerà creare i Data Source ODBC per il collegamento ai vari repository, i quali dovranno avere lo stesso nome per tutte le Workstation dello stesso progetto.
- **Impostazione del Data Source ODBC per il collegamento a DB2**

Preventivamente a questa operativa bisognerà codificare un utente DB2 per ogni progetto che possa accedere al catalogo contenente le strutture dati sorgente (in genere una replica del catalogo SYSIBM o di parte di esso), da codificare all'interno del Data Source.
- **Impostazione dei Data Source ODBC per altri DB sorgenti/target**

Creare ed impostare i Data Source ODBC per le ulteriori strutture sorgenti e per i target, avendo cura di mantenere gli stessi nomi di Data Source per tutte le Workstation di uno stesso progetto (eventualmente eseguire un export/import dei registri ODBC).

### **3.5 Manutenzione dei repository**

Onde assicurare agli utenti il miglior livello di servizio possibile e un minor tempo di restore degli ambienti in caso di crash di sistema, si consiglia di operare a due livelli: Tuning delle performance dei DB dei repository e backup degli ambienti server.

- **Tuning delle performance dei DB dei repository**

Un tuning per mantenere efficiente la lettura/scrittura dei metadati del repository, sia in fase di sviluppo dei mapping che in fase di esecuzione delle sessioni ETL, è sicuramente quello di far eseguire il calcolo delle statistiche sulle tabelle e sugli indici dello schema del repository.

Ciò può essere fatto seguendo questi passi:

- Connettersi al DB del repository tramite SQL/Plus
- Lanciare i seguenti comandi:  

```
select 'analyze table ', table_name, ' compute statistics;' from user_tables where  
table_name like 'OPB_%'  
select 'analyze index ', index_name, ' compute statistics;' from user_indexes where  
index_name like 'OPB_%'
```
- Salvare l'output ottenuto come script SQL.
- Eseguire lo script tramite SQL/Plus.

Tale operativa è da eseguirsi possibilmente di notte con cadenza **settimanale** per i repository di sviluppo e collaudo, mentre per l'ambiente di esercizio è consigliabile eseguirla manualmente immediatamente dopo i passaggi ad ambiente di produzione di nuovi metadati relativi a sessioni ETL.

- **Backup degli ambienti server**

Il backup degli ambienti server è da intendersi a tre livelli distinti: backup fisico del server, backup degli schema Oracle e backup proprietario del repository di PowerCenter. È consigliabile l'esecuzione di tutti e tre i backup, con cadenze differenti:

- **Backup fisico: Settimanale** per gli ambienti di sviluppo e collaudo; **Dipendente dalla frequenza di esecuzione delle sessioni ETL** per l'ambiente di esercizio.
- **Backup degli schema Oracle: Giornaliero** per gli ambienti di sviluppo e collaudo; **Dipendente dalla frequenza di esecuzione delle sessioni ETL** per l'ambiente di esercizio.
- **Backup proprietario del repository: Prima di ogni passaggio di ambiente.**

## 4. Architettura Business Objects/WebIntelligence

Vengono nel seguito descritti le caratteristiche e i requisiti tecnici hardware/software per l'utilizzo di un'architettura cluster WEBINTELLIGENCE (WebI). Trattandosi di una configurazione mista (sistemi operativi UNIX e NT) viene specificato il numero e le caratteristiche dei server di ciascun sistema operativo su cui vengono installati i prodotti WebI e BroadCast Agent.

I requisiti con cui è stata disegnata la soluzione sono di seguito elencati:

- **Sicurezza dei dati**  
Devono essere definiti diversi profili di utenza per consentire ai vari utenti l'accesso solo ai dati di propria competenza.
- **Scalabilità dell'architettura**  
L'architettura proposta dovrà essere scalabile all'aumentare del numero di progetti e di utenti.
- **Semplicità e razionalità nelle operative di passaggio di ambiente**  
Gli universi e i documenti BUSINESSOBJECTS e WEBINTELLIGENCE nonché i profili di utenza dovranno essere migrati nei vari ambienti (Sviluppo, Collaudo, Esercizio) con operative semplici e che assicurino il minor rischio possibile di incoerenza tra i metadata sviluppati nei suddetti ambienti.
- **Schematicità nell'assegnazione dei profili utente**  
Data la numerosità dei progetti di Data Mart e il numero crescente di utenti, è necessario uno schema guida per l'assegnazione dei profili utente che possa essere applicabile in maniera standard a qualsiasi nuovo progetto e al crescere del numero di utenti che accedono ai vari Data Mart.

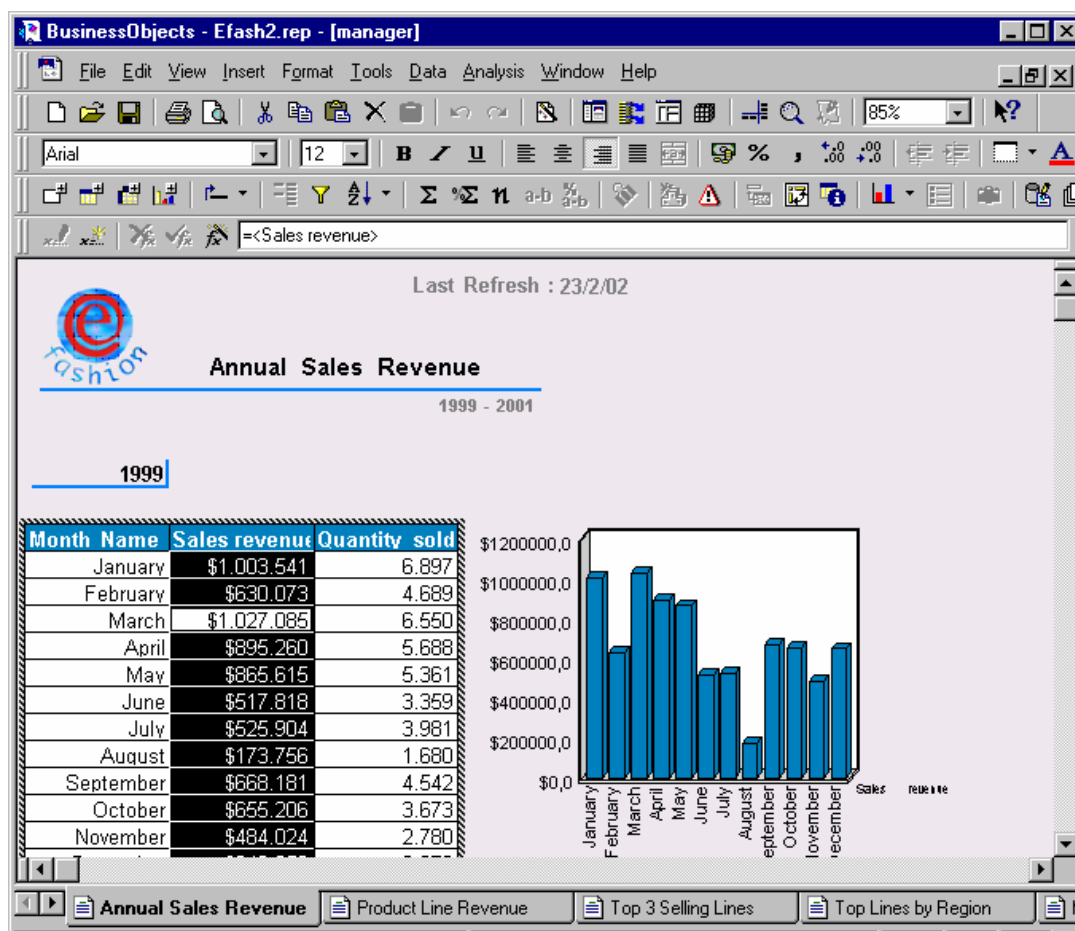
BUSINESSOBJECTS sarà utilizzato sia in versione Client/Server sia in architettura a 3 livelli (WEBINTELLIGENCE). Si ritiene più opportuno, nella maggior parte dei casi, l'utilizzo di Web Intelligence per consentire agli utenti di accedere ai documenti da qualsiasi punto utilizzando un browser; tuttavia, quando il numero di utenti è limitato e a carattere dipartimentale, la versione client/server risponde meglio alle esigenze dell'utente finale se sono richieste una completa integrazione di BUSINESSOBJECTS con Excel e alcune delle funzionalità più evolute di analisi non ancora integrate in WEBINTELLIGENCE.

Di seguito viene descritta l'architettura sul lato Hardware, Software per entrambe le soluzioni e viene inoltre approfondita l'organizzazione dei Repository di BUSINESSOBJECTS per i vari ambienti previsti. Verranno anche descritti i criteri di dimensionamento dei server Web Intelligence.

### 4.1 Sistema client/server

BUSINESSOBJECTS è un prodotto client/server che consente agli utenti non tecnici di realizzare interrogazioni estemporanee (ad hoc), preparare dei report ed analizzare le informazioni memorizzate nei datawarehouse aziendali, data mart e nei pacchetti applicativi. BUSINESSOBJECTS viene installato su PC dotati di una connessione di rete e di middleware (es. per Oracle 8, Net8) verso la base di dati aziendali. Fornisce un'interfaccia

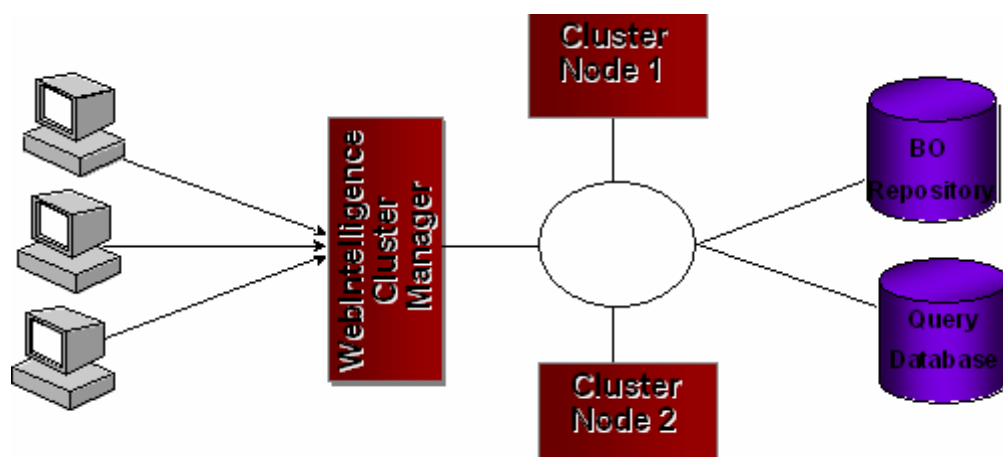
compatibile Office (vedi figura seguente) con tutte le funzionalità di creazione query/analisi multidimensionale, oltre ad una serie numerosissima di funzioni di analisi avanzata personalizzabile. Fornisce un pannello di query mediante il quale gli utenti possono creare/modificare le interrogazioni a varie sorgenti di dati quali database relazionali attraverso gli Universi BUSINESSOBJECTS, fogli Excel, stored procedures, procedure Visual Basic o SQL a mano libera.



## 4.2 Sistema WebIntelligence

WEBINTELLIGENCE è uno strumento *multi-tier* di tipo *thin-client* che consente agli utenti non tecnici di realizzare da un browser Web interrogazioni estemporanee (ad hoc), preparare dei report ed analizzare le informazioni memorizzate nei datawarehouse aziendali, data mart e nei pacchetti applicativi.

Il sistema è composto di un web browser sul primo livello, un Application Server sul medio livello, ed un database come terzo livello. Ciascun livello nel sistema comunica utilizzando una rete: LAN, WAN, Internet. La figura seguente mostra un grafico schematico della soluzione cluster WEBINTELLIGENCE.



WEBINTELLIGENCE può essere installato in modalità Stand-Alone (un'unica macchina) oppure in modalità Cluster nodes (installazione distribuita su più macchine). Nella nostra analisi considereremo una configurazione Cluster nodes.

Nel dimensionamento di un sistema WebIntelligence devono essere considerate molte variabili. In particolare, la velocità della rete tra i client ed il secondo livello, e tra il secondo livello e la banca dati. Altre variabili da tenere opportunamente in considerazione sono sicuramente le prestazioni del database dove viene di norma inserito il repository BUSINESSOBJECTS e del database dove sono memorizzate le informazioni aziendali su cui effettuare le query per l'analisi.

Un altro elemento importante ai fini del dimensionamento è sicuramente rappresentato dall'esigenza di determinate performance che si vogliono ottenere nell'aggiornamento dei documenti.

Per le analisi che seguiranno, non essendo noti i dati sulle prestazioni della rete e dei DB implicati, verranno appositamente trascurate le variabili di cui sopra e ci si focalizzerà invece sul secondo livello dell'architettura ossia le macchine destinate ad ospitare le applicazioni WEBINTELLIGENCE e BROADCAST AGENT.

#### 4.2.1 Azioni utente

Le tipiche attività di un utente WebIntelligence sono: query ad hoc, lettura e refresh di documenti memorizzati sul secondo livello (documenti personali), lettura e refresh di documenti memorizzati sul repository BUSINESSOBJECTS.

Inoltre è da considerare che la filosofia della versione 2.5 di WebI consiste nell'implementare un ambiente multi-formato di consultazione/fruizione delle informazioni aziendali conformemente alla metafora del Web. WEBINTELLIGENCE 2.5 permette, quindi, all'utente di leggere e rieseguire report generati direttamente con la piattaforma WEBINTELLIGENCE sia di leggere e aggiornare dei documenti generati dalla piattaforma Client/Server e distribuiti (via Repository) per gli utenti WEBINTELLIGENCE. Permette, inoltre, di navigare tra gli stessi



tramite hyperlink, di collegare immagini, suoni ecc. nonché di attivare pagine html anche di altri ambienti.

#### **4.2.2 Dimensionamento**

Come assunzioni generali per l'analisi del dimensionamento si definiscono:

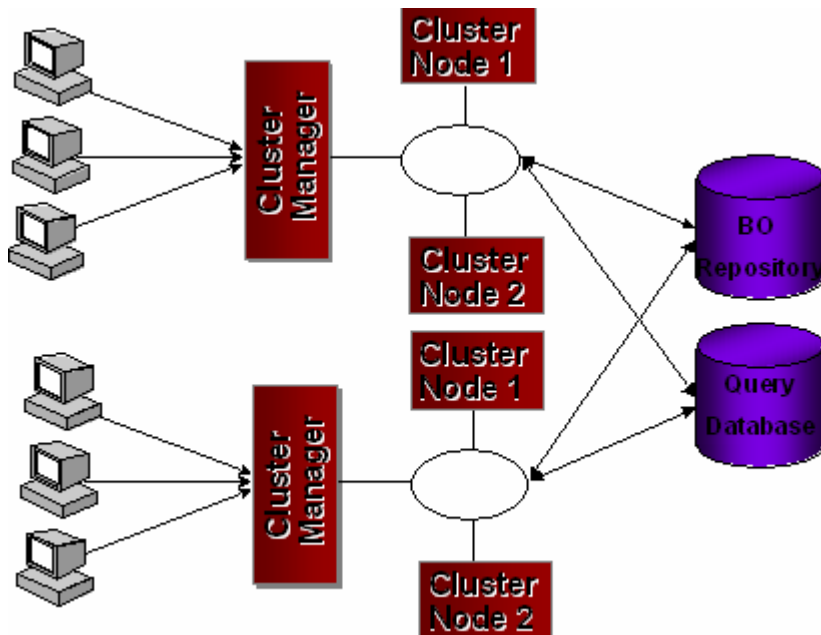
- **Utenti Connessi**  
Un utente connesso è collegato al sistema WEBINTELLIGENCE ed ha un corrispondente processo WIQT.exe attivo ma non sta correntemente consumando CPU. L'utente connesso consuma risorse di memoria ma non occupa risorse del processore. Tipicamente potrebbe trattarsi dell'utente che ha effettuato un'operazione di Refresh e che ora sta leggendo il documento.
- **Utenti Attivi**  
Si definisce utente attivo un utente connesso che svolge operazioni che coinvolgono la CPU del server, come refresh di un report o generazione di un documento. L'utente attivo consuma sia risorse di memoria che di CPU.
- **Utenti potenziali**  
Sono gli utenti del sistema che non sono in un particolare momento collegati a WebIntelligence ma che sono comunque presenti nel repository BusinessObjects.

Da analisi precedenti condotte su prodotti server è possibile determinare che in base alla totalità della popolazione di utenti potenziali generalmente solo un **25%** degli utenti in un dato istante possono essere considerati connessi e di questi solo un **20%** sono attivi.

#### **4.2.3 Partizionamento**

Il partizionamento consiste nell'avere due installazioni distribuite WEBINTELLIGENCE al fine di suddividere ulteriormente il carico di lavoro sulle macchine e permettere in questa maniera un aumento incrementale delle utenze del sistema.

Entrambe le installazioni si riferiscono allo stesso Repository ed allo stesso database di dati aziendali. Nella figura seguente è mostrato un esempio di installazione partizionata.



#### 4.2.4 Il Clustering di WebIntelligence

Vediamo più in dettaglio il funzionamento di WEBINTELLIGENCE in configurazione Cluster. I Cluster WEBINTELLIGENCE possono essere di due tipologie:

##### 1. Cluster Manager

E' la macchina su cui viene installato:

il WEBSERVER, l'OAD (ORB Activation Daemon), l'OSAGENT (utility Visigenic che WEBINTELLIGENCE utilizza come Broker Corba), i moduli WebIntelligence. In una installazione Cluster può esserci una ed una sola macchina Cluster Manager.

Il Cluster Manager è il punto centrale di coordinamento del sistema WebIntelligence e realizza i seguenti servizi :

- Tiene traccia dei processi sul sistema WebIntelligence utilizzando il WIClusterManager.exe
- Mantiene le informazioni sulle sessioni degli utenti attivi sul sistema (Session Manager)

Il Cluster manager tiene traccia delle sessioni utente e delle macchine del sistema WebIntelligence dove sono stati attivati i processi per l'utente in questione.

##### 2. Cluster Node

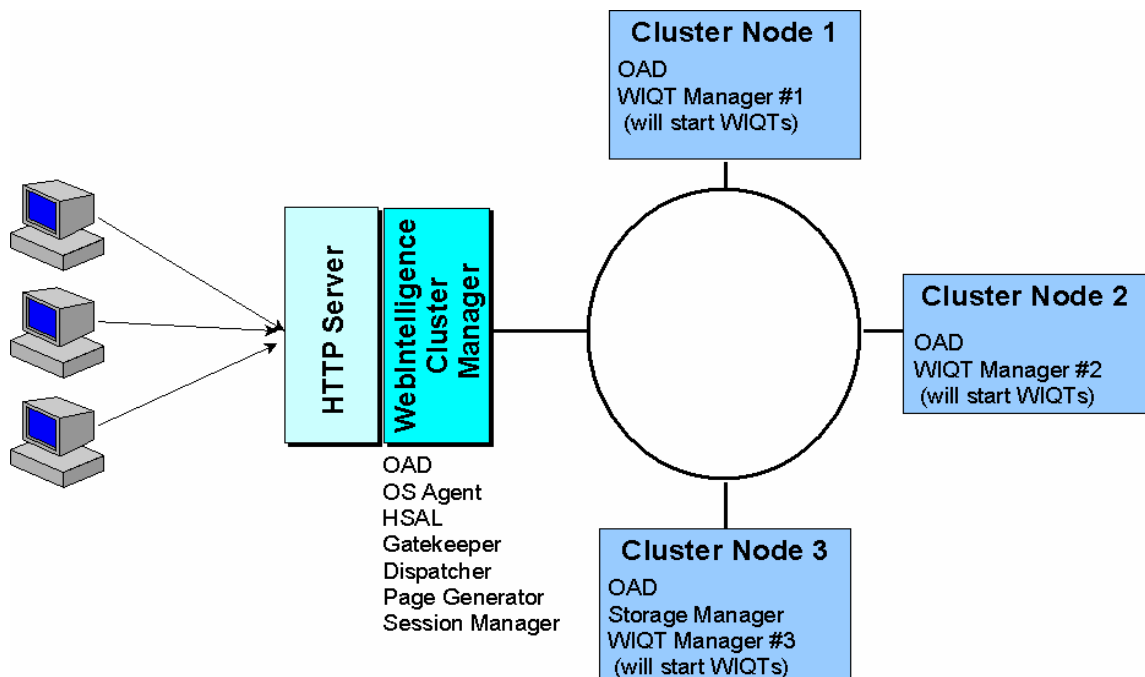
E' possibile avere uno o più installazioni Cluster Node.

Ognuna di esse esegue l'OAD per comunicare con il Cluster Manager e per effettuare lo start dei processi WebIntelligence sul relativo nodo.

E' possibile configurare un Cluster node in modo da attivare solo i moduli del sistema che si vogliono distribuire. Tale operazione può essere fatta direttamente attraverso il modulo

Administrator che consente l'attivazione/disattivazione dei vari servizi sulle varie macchine della rete WebIntelligence.

Ai fini di una corretta distribuzione del carico di lavoro è opportuno configurare i vari nodi con i moduli WIManager e Report Server al fine di poter suddividere i processi WIQT e BusObj. Questa figura mostra la distribuzione dei processi sui cluster.



### 4.3 BROADCAST AGENT

Il BroadCast Agent (BCA) è un prodotto Server Enterprise che abilita gli utenti BUSINESSOBJECTS o WebIntelligence a elaborare automaticamente e a pubblicare i documenti tramite il repository, una intranet, il file system del server e il World Wide Web. Il BCA può essere installato sia su server UNIX che su Windows NT

E' composto da due moduli principali, che possono essere installati su più server in una rete per un uso distribuito. I due moduli sono:

lo *Scheduler*, il quale controlla la programmazione dei task e li avvia come previsto;

la *Console*, utilizzata dall'amministratore per controllare lo stato dei task elaborati e ancora da elaborare, e per eseguire le azioni quali l'esecuzione immediata di un task o il suo annullamento.

#### Requisiti tecnici:

Di seguito vengono specificati i requisiti tecnici del/dei server per ottenere l'aggiornamento automatico dei documenti. Il server che ospita il BCA deve essere un server dedicato, cioè non dovrebbe avere altre applicazioni server installate. Sebbene il BCA possa essere

installato sia su UNIX che su NT, Viene considerato il caso in cui il server che ospita il BCA sia un server NT 4.0.

### Hardware

Per l'utilizzo del BCA è necessario un server con Pentium II 200 Mhz. Si consiglia però l'utilizzo di un server bi-processore a 450 Mhz con cache di tipo L2. La RAM di base necessaria è 256 MB, ed occorrerà avere 16 MB di RAM aggiuntive per ogni documento in più che si desidera aggiornare contemporaneamente. Se la finestra temporale in cui il BCA deve effettuare l'aggiornamento è limitata, e soprattutto se i documenti da aggiornare sono pesanti in termini di refresh e di ricalcolo delle eventuali variabili all'interno di documenti full-client, si consiglia di dotare il server di 512 MB di RAM.

Bisogna inoltre configurare uno spazio di *swap* sufficiente a gestire situazioni di massimo carico perché se un task non può essere elaborato nella RAM disponibile si verifica uno "swap" (cioè parte della memoria disco fisso viene utilizzata come RAM) e il processo rallenta notevolmente. Inoltre, se lo spazio di swap viene esaurito si possono avere problemi di affidabilità (uno o più aggiornamenti possono fallire). Lo spazio su disco necessario è di circa **300 MB**.

### Software

Windows NT Server 4.0 Service Pack 5. Il server non deve essere un Backup Domain Controller. E' consigliabile, inoltre, installare NT Server come 'stand-alone'.

## **4.4 Dimensionamento dei Server WebIntelligence**

L'architettura proposta è di tipo misto UNIX (IBM AIX) e Windows NT. A questo proposito, occorre tenere presente che tutti i processi BUSINESSOBJECTS girano su server NT mentre i processi WIQT girano su server UNIX.

Non essendo possibile determinare, in questa fase, il numero esatto di utenti presenti sul repository BusinessObjects, si evidenziano le caratteristiche hardware dei server scalate in fasce di utenza potenziale (da 100 a 100.000) considerando le seguenti assunzioni:

- ❑ Percentuale di picco di utenti connessi : 20%
- ❑ Percentuale di picco di utenti concorrenti : 5%
- ❑ Percentuale di documenti full-client BusinessObjects : 50%

Le fasce di utenza ipotizzabili e considerate sono le seguenti:

- 100 utenti potenziali
- 500 utenti potenziali
- 1.000 utenti potenziali
- 2.000 utenti potenziali
- 5.000 utenti potenziali
- 10.000 utenti potenziali
- 15.000 utenti potenziali
- 50.000 utenti potenziali
- 75.000 utenti potenziali
- 100.000 utenti potenziali

Per ciascuna fascia di utenza si fornisce il dimensionamento sia dei nodi NT sia dei nodi UNIX in termini di:

- Numero minimo di nodi;
- Dimensione minima della RAM per ogni nodo;
- Numero di processori per nodo;
- Eventuale nodo di controllo richiesto. Il nodo di controllo, quando necessario, ospita sia il server HTTP che il cluster manager e deve essere un server più potente;

# Scenario	Utenti Attivi	Utenti Connessi	Utenti Potenziati	Configurazione minima
1	5	25	100	1 nodo NT RAM 256 MB 1 processore; 1 nodo UNIX RAM 400 MB 1 processore.
2	25	125	500	1 nodo NT RAM 725 MB 2 processori; 1 nodo UNIX RAM 1600 MB 1 processore.
3	50	250	1.000	1 nodo NT RAM 1350 MB 4 processori; 1 nodo UNIX RAM 3100 MB 1 processore.
4	100	500	2.000	2 nodi NT RAM 1350 MB 4 processori; 2 nodi UNIX RAM 6 GB 2 processori. (*)
5	250	1.250	5.000	4 nodi NT RAM 1600 MB 4 processori; 2 nodi UNIX RAM 15 GB 4 processori. (*) (**)
6	500	2.500	10.000	7 nodi NT RAM 1800 MB 4 processori; 2 nodi UNIX RAM 30 GB 8 processori. (*) (**)
7	750	3.750	15.000	10 nodi NT RAM 1975 MB 4 processori; 3 nodi UNIX RAM 22 GB 8 processori. (*) (**)
8	2.500	12.500	50.000	32 nodi NT RAM 2GB 4 processori; 6 nodi UNIX RAM 30 GB 8 processori. (*) (**)
9	3.750	18.750	75.000	47 nodi NT RAM 2GB 4 processori; 8 nodi UNIX RAM 28 GB 8 processori. (*) (**)
10	5.000	25.000	100.000	63 nodi NT RAM 2GB 4 processori; 11 nodi UNIX RAM 30 GB 8 processori. (*) (**)

Qui di seguito si riporta una stima dell'hardware per alcuni scenari, quelli di maggior interesse, ipotizzando una percentuale di documenti BusinessObjects del 20%:

# Scenario	Utenti Attivi	Utenti Connessi	Utenti Potenziali	Configurazione minima
4	100	500	2.000	1 nodo NT RAM 1100 MB 2 processori; 1 nodo UNIX RAM 5700 MB 2 processori.
5	250	1.250	5.000	2 nodi NT 1350 MB RAM 4 processori; 2 nodi UNIX RAM 14 GB 8 processori. (*) (**)
6	500	2.500	10.000	3 nodi NT RAM 1766 MB 4 processori; 2 nodi UNIX RAM 28 GB 12 processori. (*) (**)
8	2.500	12.500	50.000	13 nodi NT RAM 2 GB 4 processori; 5 nodi UNIX RAM 28 GB 12 processori. (*) (**)

(\*) Uno dei server UNIX è il nodo di controllo

(\*\*) Utilizzo del partizionamento

#### 4.5 Configurazione del repository BUSINESSOBJECTS/WebIntelligence

E' previsto l'utilizzo di due repository BUSINESSOBJECTS : uno per l'ambiente di sviluppo e di collaudo, uno per l'ambiente di esercizio.

Ciascun repository sarà organizzato secondo un'architettura distribuita caratterizzata da un singolo dominio di sicurezza e dai domini degli universi e dei documenti distribuiti. Più precisamente saranno creati un dominio degli universi e un dominio dei documenti per ogni progetto di Data Mart. Dovranno essere previsti, pertanto, uno schema Oracle per il dominio di sicurezza e per ogni progetto uno schema Oracle che contenga il relativo dominio degli universi e dei documenti.

Poiché gli ambienti di sviluppo e collaudo coesistono sullo stesso server Unix e sulla stessa istanza Oracle, dovranno essere previsti all'interno di essa schemi Oracle diversi per il repository di sviluppo e quello di collaudo.

La scelta dell'architettura distribuita per il repository BUSINESSOBJECTS consente di utilizzare per più progetti di Data Mart un unico server Web Intelligence (un server WebIntelligence punta a un unico repository) e al tempo stesso risponde ai requisiti di sicurezza e riservatezza visto che è possibile definire dei profili di utenza che possono accedere a ben determinati domini di universi e di documenti.

Durante la creazione del/i repository, BUSINESSOBJECTS provvede anche alla creazione di vincoli di integrità referenziale nonché di indici con lo scopo di ottimizzare le prestazioni durante il login degli utenti sia da client/server che da WEBINTELLIGENCE. Naturalmente, in presenza di un repository contenente decine di migliaia di utenti, occorrerà misurare preventivamente il tempo medio di connessione, ed eventualmente aggiungere spazio ai suddetti indici.

#### **4.6 Schema guida per l'assegnazione dei profili utente**

Si individuano i seguenti profili utente per gli ambienti di sviluppo, collaudo ed esercizio:

- **Amministratore del repository**  
L'amministratore del repository è colui che ha il profilo di supervisore generale ed è l'amministratore principale del sistema. Ha la visibilità di tutta la struttura gerarchica degli utenti presenti nel repository e può effettuare qualsiasi operazione di assegnazione profili/funzionalità a tutti gli utenti nonché costruire gruppi e stabilire l'appartenenza degli utenti a questi ultimi.
- **Supervisore di gruppo**  
Come il precedente, con la differenza che la visibilità che ha delle utenze è limitata al gruppo a cui appartiene e le operazioni che svolge sono relative solamente agli utenti del suo gruppo.
- **Utente Sviluppatore**  
Avrà il profilo di Designer e potrà creare/modificare universi e documenti Full Client / WebI. Per ogni progetto di Data Mart dovranno essere definiti uno o più utenti con questo profilo, aventi privilegi solo sui domini degli universi e dei documenti del progetto.
- **Utente finale**  
E' il profilo dell'utente che crea ed esegue query, crea documenti e salva documenti nella cartella personale (ma non può pubblicarli sul repository) e analizza i dati. E' soggetto alla restrizione dei dati accedibili (attraverso gli universi) e alle funzionalità di BUSINESSOBJECTS e WEBINTELLIGENCE che può eseguire assegnategli dal supervisore del gruppo di appartenenza.

Una volta individuati gli utenti del progetto, essi dovranno essere definiti con uguale nome utente e password su ciascuno dei tre repository (sviluppo, collaudo ed esercizio). Tramite l'utilizzo di gruppi di utenza sarà possibile semplificare l'amministrazione del repository: ogni utente appartiene ad almeno un gruppo e ne eredita i diritti. All'interno del gruppo, i profili dei singoli utenti potranno essere personalizzati.

La tabella di seguito riassume i permessi (R=in sola lettura, RW= in lettura/modifica) sui vari repository e quali prodotti sono utilizzabili per ogni profilo utente individuato:

Profilo	Repository di Sviluppo	Repository di Collaudo	Repository di Esercizio	4.6.1.1.1 Domini di universi	4.6.1.1.2 Domini di documenti	4.6.1.1.3 Prodotti
Amministratore del repository	RW	RW	RW	RW	RW	Tutti
Supervisore di gruppo	–	RW (sul gruppo di appartenenza)	RW (sul gruppo di appartenenza)	RW (sul gruppo di appartenenza)	RW (sul gruppo di appartenenza)	Tutti
Utente Sviluppatore	RW	RW	RW	RW	RW	Designer, BusinessObjects, WebIntelligence
Utente finale	–	–	R	R (solo quelli abilitati)	R (solo quelli abilitati)	BusinessObjects, WebIntelligence

#### 4.7 Migrazione nei vari ambienti

La migrazione tra i vari ambienti, ossia il trasferimento delle informazioni da un repository ad un altro comprende essenzialmente il trasferimento di

- Utenti con profili di accesso e connessioni (dominio di sicurezza)
- Documenti BusinessObjects e WebIntelligence
- Universi.

Queste informazioni sono contenute nelle tabelle di database del repository e il loro trasferimento può essere effettuato sia in modo automatico, operando un “export” delle tabelle di repository da uno schema all’altro, oppure in modo manuale, avvalendosi delle funzioni di import/export messe a disposizione dallo strumento stesso, permettendo in tal modo di analizzare le informazioni da migrare e stabilire ciò che deve essere migrato. Per quanto riguarda gli utenti, il modulo SUPERVISOR fornisce la funzionalità di Import/Export su file testo delle informazioni riguardanti i nomi utente, le password nonché i profili utente. Per i documenti e gli universi si può effettuare l’importazione dal vecchio repository e la pubblicazione sul nuovo del singolo o tutti i documenti / universi sui domini desiderati utilizzando i moduli BUSINESSOBJECTS e DESIGNER rispettivamente.

- Passaggio da sviluppo a collaudo (manuale)

Molti dei documenti e degli universi presenti nel repository di sviluppo non saranno migrabili in collaudo, per cui questo passaggio sarà effettuato in modo manuale. Il processo di creazione e aggiornamento dei gruppi di utenti e dei singoli utenti può essere parzialmente automatizzata tramite l’importazione/esportazione di tali



informazioni da/in un file di testo preformattato tramite appositi comandi. Pertanto la configurazione degli utenti e dei gruppi dell'ambiente di sviluppo può essere esportata in un file di testo, che poi dovrà essere importato nel repository di collaudo per ricreare la configurazione degli utenti. Tuttavia le operazioni di importazione e di esportazione non possono essere utilizzate per l'assegnazione di risorse (universi, documenti, ecc.), per cui i diritti di accesso alle risorse BUSINESSOBJECTS dovranno essere in parte ridefiniti manualmente nel repository di esercizio.

- Passaggio da collaudo ad esercizio (automatico)

Il repository di esercizio dovrà rispecchiare esattamente la struttura di quello di collaudo, e la migrazione può, quindi, essere effettuata in modo automatico operando un "export" del database.

La migrazione dall'ambiente di collaudo a quello di esercizio comporta la copia del dominio degli universi, del dominio dei documenti, dei gruppi di utenti e dei singoli utenti (dominio di sicurezza) dal repository di collaudo a quello di esercizio. Per copiare i domini degli universi e dei documenti nonché le categorie è sufficiente fare l'export degli utenti Oracle ad essi relativi e l'import verso i corrispondenti utenti Oracle dell'ambiente di esercizio.

#### **4.8 Setup delle Workstation Full Client**

Per i progetti che utilizzano l'architettura client/server devono essere eseguite, per ogni Workstation client di progetto, le seguenti operative di setup:

- **Installazione di BUSINESSOBJECTS client**
- **Installazione del middleware di connessione al database Oracle**
- **Creazione dell'alias Oracle per il collegamento al repository**  
E' opportuno che gli alias Oracle per il collegamento ai repository abbiano lo stesso nome per tutte le workstation dello stesso progetto.
- **Copia del/i file di chiave per il/i repository (Bomain.key)**  
E' opportuno che i file Bomain.key per l'accesso ai repository mantengano gli stessi nomi per tutte le Workstation dello stesso progetto.

#### **4.9 Gestione delle versioni**

BUSINESSOBJECTS non ha al suo interno un modulo di "versioning" dei file. Tuttavia, nel modulo Designer è presente un meccanismo di "lock" degli universi mediante il quale ciascun designer può "bloccare" lo sviluppo del/degli universi su cui sta lavorando, impedendo ad altri designer di effettuare l'export dello stesso universo sul repository (senza però impedirne l'utilizzo da parte degli utenti e degli altri designer). Per quanto riguarda i documenti occorrerà tenere traccia delle diverse versioni dei documenti inserendo nel nome del file un suffisso che ne indica la versione.

#### 4.10 **Manutenzione**

- **Repository**

Periodicamente si farà uso delle funzionalità interne a Supervisor di scan/compact dei vari domini di documenti e di universi nonché del dominio di sicurezza. Per quanto riguarda il backup, si potrà procedere ad un semplice backup delle tabelle di repository utilizzando le funzioni messe a disposizione dal database stesso.

- **Sistema WebIntelligence**

Dopo l'installazione e l'impostazione iniziale/tuning dei parametri di funzionamento, si potrà verificare periodicamente la validità dei parametri in presenza di modifiche sostanziali all'ambiente operativo (numero di utenti, numero e capacità dei nodi nel cluster, numero di documenti presenti nel repository, durata delle query, ecc).

- **BroadCast Agent**

Anche per il BroadCast Agent occorrerà verificare, al sopraggiungere di modifiche importanti all'ambiente operativo, alcuni parametri-chiave impostati durante l'installazione, come, ad esempio, il numero di documenti aggiornabili contemporaneamente, la durata della sessione, ecc.

#### 4.11 **Requisiti di versioni e compatibilità del SW**

Sistemi operativi / database / Web Server supportati con WEBINTELLIGENCE 2.6 e BUSINESSOBJECTS 5.0 :

- ✓ **IBM AIX 4.3**

Middleware: Oracle Net 8.0.x      Server: Oracle 8.0.x  
                  Oracle Net 8.1.x      Server: Oracle 8.1.x  
HTTP Server: Apache 1.3.9 (solo inglese)  
                  Netscape Enterprise Server 3.6, 3.5.1, 4.0.2 (solo inglese)

- ✓ **Sun Solaris 2.6**

Middleware: Oracle Net 8.0.x      Server: Oracle 8.0.x  
                  Oracle Net 8.1.x      Server: Oracle 8.1.x  
  
HTTP Server: Apache 1.3.6 (solo inglese)  
                  Netscape Enterprise Server 3.6, 3.5.1 (solo inglese)

✓ **Microsoft Windows NT4 SP6a/2000**

Middleware: Oracle Net 8.0.x      Server: Oracle 8.0.x  
                 Oracle Net 8.1.5      Server: Oracle 8.1.5  
                 Oracle Sql\*Net 2.3.4      Server: Oracle 7.3.4

HTTP Server : Internet Information Server 4.0 (con programmabilità ASP)

                 Netscape Enterprise Server 3.6

                 Netscape Enterprise Server 4.0.3 (solo inglese)

                 Apache 1.3.6 (solo inglese)

## 5. Schedulazione tramite OPC

### 5.1 Procedure ETL con sorgente base dati DB2

Per consentire l'automazione dell'esecuzione delle procedure di ETL, che tramite il processo Unix denominato "sessione" trasportano i dati dal DB2 su sistema centrale a tabelle Oracle su sistema dipartimentale Unix, è stato deciso di utilizzare il prodotto OPC.

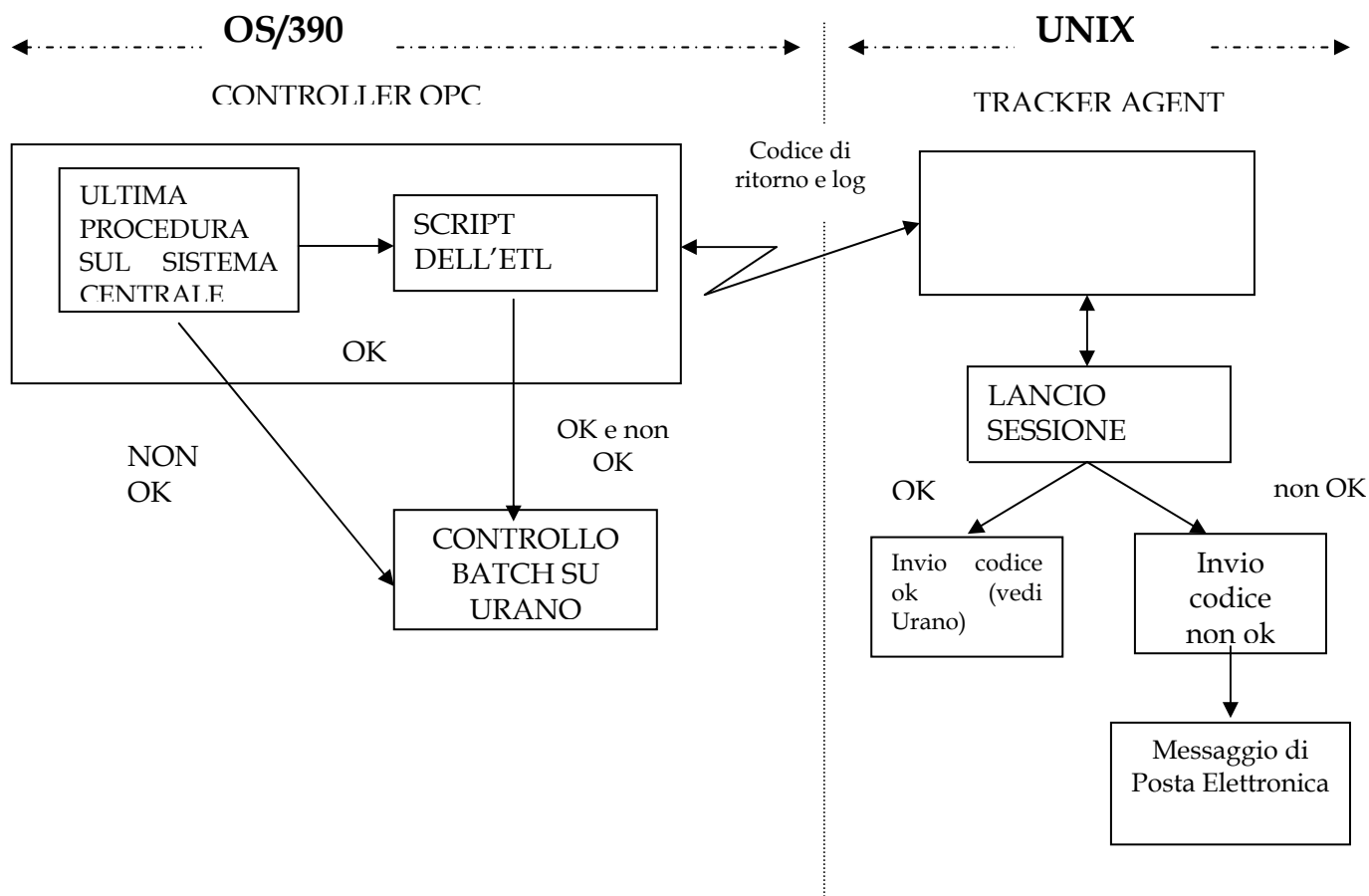
In particolare, per realizzare il colloquio da sistema centrale OS/390 a sistema dipartimentale Unix, viene utilizzato il componente dell'OPC, denominato "Tracker Agent", che si installa e si personalizza su Unix; il componente in questione colloquia con l'Engine di OPC situato sul sistema OS/390, a sua volta denominato "Controller". L'architettura descritta consente di eseguire processi Unix sulla base di script che sono sottoposti al controller su OS/390. In questo modo è possibile inserire processi Unix (le sessioni di ETL) come Job su sistema centrale, condizionati fisicamente o logicamente alla corretta esecuzione di job su mainframe.

Le procedure di ETL che hanno come fonte dati le tabelle del DB2 su sistema centrale possono essere di due tipi:

- procedure che ad ogni caricamento ricoprono integralmente la base dati target su Oracle;
- procedure che ad ogni caricamento aggiornano i dati pregressi mediante operazioni di inserimento o di replace.

In questo contesto è possibile schematizzare due prototipi di "trenini batch" atti a garantire l'automatismo, il controllo e la sicurezza delle procedure di ETL.

Nel primo caso:



In questo caso i gruppi applicativi dovranno predisporre un trenino costituito da un unico job contenente lo script Unix; in generale è un comando batch del tipo:  
/percorso/comandobatch  
dove comandobatch è così strutturato:

```
su - u_etl_cl
/home/u_etl_cl/Informatica/PowerCenter/pmcmd start nome_sessione utente password
nome_folder : nome_batch 0 1 exit
```

I comandi sono espressi in grassetto e hanno il seguente significato:

- **su** è un comando Unix che mi consente di cambiare utenza (viene settata quella dell'ETL)
- **pmcmd start** è il comando del prodotto di ETL che mi consente di lanciare in modalità batch le sessioni
- **0** e **1** significano rispettivamente che sto lanciando un batch e che il processo Unix deve attendere il prompt prima di segnalare che il comando è completato.

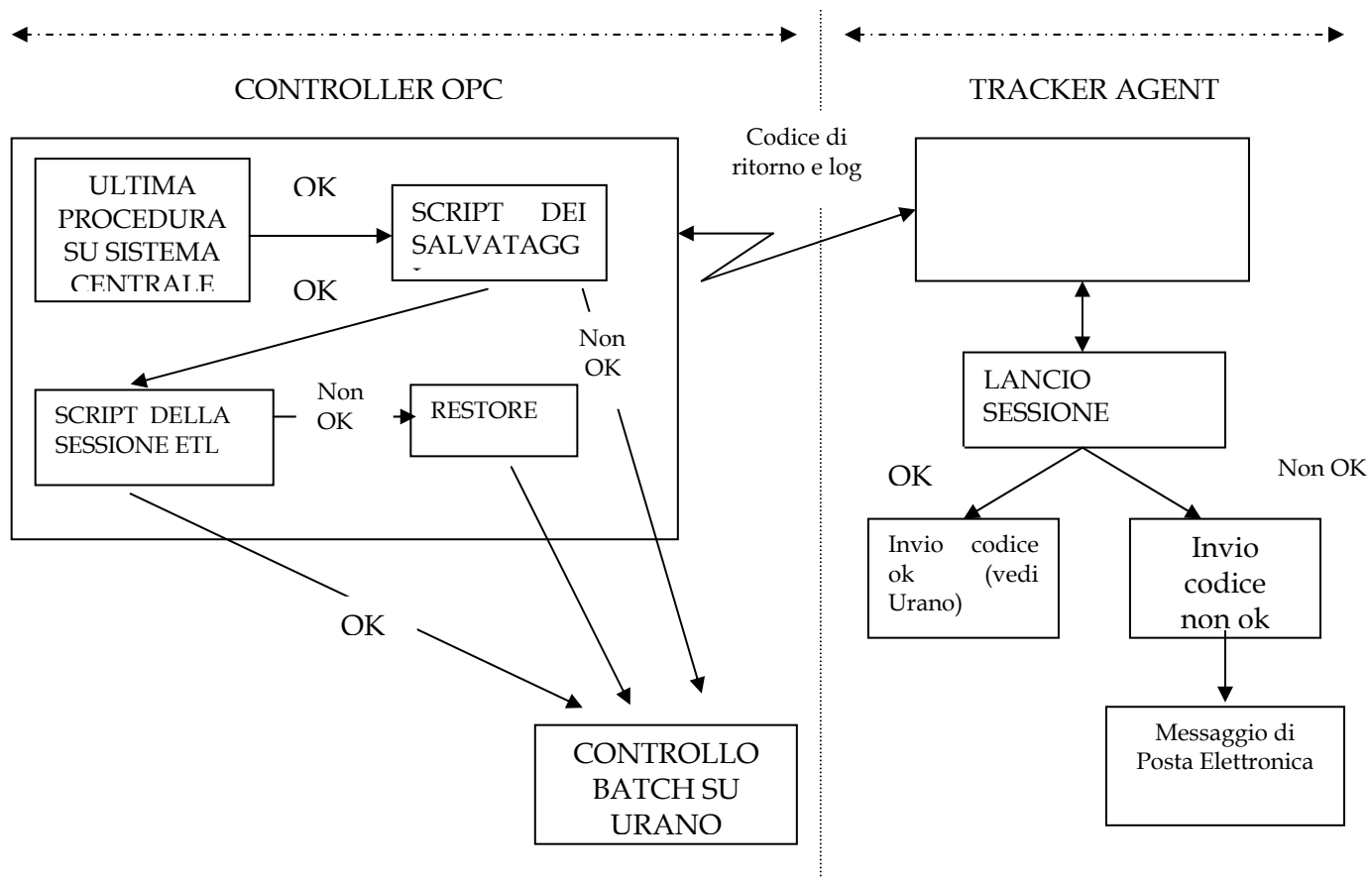
I parametri sono espressi in corsivo e vanno personalizzati dall'utente.

Il comando batch viene inviato dal Controller OPC su OS/390 al componente Tracker Agent su Unix. Il Tracker sottopone il file allo Unix che rimane in attesa del completamento del comando (che può essere costituito anche da più sessioni in cascata). Al completamento del lavoro ETL il log con i codici di ritorno viene raccolto dal Tracker e inviato al Controller. L'esecuzione del trenino, indipendentemente dal suo esito può essere controllato mediante un browser sul server "Urano" nel menù relativo alle schedulazioni e identificato dall'URL :

<http://urano/Applications/Batch/Schedulazione.html>.

Inoltre, nel caso che la sessione ETL non termini correttamente il prodotto di ETL consente di personalizzare la sessione in modo che sia inviato un messaggio di posta elettronica (personalizzazione a carico dei gruppi applicativi).

Nel secondo caso:



In questo caso l'integrità dei dati deve essere salvaguardata con una procedura più complessa. Infatti, poiché i dati possono essere inseriti ma anche aggiornati, è possibile ottenere come situazione finale del caricamento una fotografia della base dati diversa dalla versione precedente e che (questo è il punto critico) non necessariamente può essere ricostruita a partire dai dati gestionali.

In questo contesto i gruppi applicativi dovranno predisporre un trenino batch così costituito:

- uno script di salvataggio della base dati Oracle esistente prima dell'aggiornamento,
- uno script della sessione di ETL,
- un eventuale script di restore della base dati su Oracle alla versione precedente, nel caso che la sessione di ETL non termini correttamente.

La procedura è completamente automatizzabile mediante la gestione dei codici di ritorno; è possibile verificare sul sito URANO l'esito finale della procedura e anche gli step intermedi. Analogamente al caso precedente, nel caso che la sessione non termini correttamente può essere automatizzata la partenza di un messaggio di posta elettronica.

Il punto critico di questa procedura è rappresentato dal job di restore; infatti, prima di qualunque altra azione, è strettamente necessario che termini correttamente il job di restore al fine di ripristinare una situazione di partenza congruente. Quindi nel caso che lo script di restore non termini correttamente, per garantire la sicurezza e l'integrità dei Data Mart, (magari è opportuno che l'OPC sia personalizzato in modo da rieseguire una seconda volta il restore terminato male) è opportuno che sia implementato un meccanismo (ad esempio un flag di controllo) che inibisca il giorno successivo (o la volta successiva, in funzione della frequenza di schedulazione) l'esecuzione del trenino. Solamente alla fine corretta dello script di restore verrà riattivato il normale processo di alimentazione delle basi dati conoscitive.