



**INTESA SANPAOLO
GROUP SERVICES**

DIREZIONE SISTEMI INFORMATIVI

DC SIST SISTEMI APPLICATIVI DATI RISCHIO

DC SIST DATA TECHNOLOGY

Reporting B(F)D Qlik – Linee guida di sviluppo

"Modifica Proprietà Documento"

Fare Doppio Clic sul bottone per Modificare le proprietà del Documento.

SSA Acronimo	2C REPQ0		
Codice Progetto	PRJ336732-01		
Classificazione	-- Classificazione --		
Autorizzati	-- Autorizzati --		
Autore	Qlik		
Nome file	Reporting B(F)D-Linee guida di sviluppo.doc		
Versione	1.0	Stato	-- Stato --
Approvato da	-- Approvato da --		
Data creazione	01/08/2017	Data ultimo aggiornamento	18/12/2017

Si rammentano le seguenti Note Operative:

1. I documenti classificati "ad uso interno" non possono essere divulgati all'esterno di Intesa Sanpaolo S.p.A.
2. La persona che venisse in possesso di un documento classificato "riservato" e non trovi il suo nominativo nella lista del personale autorizzato all'utilizzo è invitato a consegnare il documento ad una delle persone citate nella lista di cui sopra.

Cronologia

Nella seguente tabella è riportata la cronologia delle modifiche al presente documento

Data	Autore	Motivazione
18/12/2017	Qlik	

Approvazione Documento

Nella seguente tabella è riportata la lista degli stakeholder con cui il documento è stato condiviso e da cui è stato approvato.

Stakeholder	Ruolo (Rispetto al Progetto)	Data Condivisione	Data Approvazione

Indice Generale:

CRONOLOGIA	2
APPROVAZIONE DOCUMENTO	3
INDICE GENERALE:	4
1 SCOPO E AMBITO	5
2 ARCHITETTURE APPLICATIVE	6
2.1 ARCHITETTURE APPLICATIVE PROGETTUALI	6
2.1.1 Architettura ad un solo livello (Diretta)	6
2.1.2 Architettura a 2 livelli con QVD	7
2.1.3 Architettura a 2 livelli con QVF (QlikMart)	8
2.1.4 Architettura a 3 livelli con QVF (Base App e QlikMart)	9
2.1.5 Architettura a 3 livelli mista QVD e QVF	10
2.2 ARCHITETTURE DI PROGETTO DI RIFERIMENTO	11
2.2.1 Architettura ad un solo livello (Diretta)	11
2.2.2 Architettura a 3 livelli mista QVD e QVF	12
3 ARCHITETTURA DI RIFERIMENTO	14
3.1 CARTELLA APPLICATIVA DI RIFERIMENTO	14
3.2 GESTIONE CARTELLA QVD COMUNI (CROSS PROGETTO)	18
3.3 DEVELOPMENT FRAMEWORK	18
3.3.1 Applicazioni Qlik naming convention	18
3.3.2 Back-end	19
3.3.3 Front-end	20
3.4 ARCHITETTURE APPLICATIVE	21
3.4.1 livello "0" - data source	22
3.4.2 Extraction level	24
3.4.3 Specifiche di setup e uso	25
3.4.4 Impostazione delle variabili di base	25
3.4.4.1.1 Microsoft Excel file "ETL01_Settings.xls"	26
3.4.4.1.2 ETL01.qvf Application	33
3.4.5 Transformation level	36
3.4.6 Specifiche di setup e uso	37
3.4.6.1 SetUp Base Variables	37
3.4.7 Livello presentation	38
3.5 ARCHITETTURA AD UN SOLO LIVELLO (DIRETTA)	39
3.6 BEST PRACTICE PER LA CREAZIONE DI APP DA UTILIZZARE IN SELF SERVICE	40
3.6.1.1 Master Library (Voci Principali)	41
3.6.1.2 Variabili	43
4 DATA VISUALIZATION	44
5 GESTIONE EXTENSIONS	47
6 CREAZIONE CONNESSIONI ODBC	48
6.1 GESTIONE CARTELLE GENERICHE PER UTENTE	48
7 MAPPATURA ESIGENZA DI REPORTING E ARCHITETTURA CONSIGLIATA	49
8 CONI DI VISIBILITÀ DEI DATI	50
8.1 CAMPI DI SISTEMA DELLA SEZIONE RELATIVA AL CONTROLLO DEGLI ACCESSI	51
8.2 RIDUZIONE DINAMICA DEI DATI	52
9 ALLEGATI	56

1 Scopo e Ambito

Intesa - San Paolo ha definito e adottato una serie di linee guida per i progetti di Business Intelligence basati sulla piattaforma Qlik. La conformità a queste linee guida servirà a:

- Fornire una serie di principi e regole generali per gli sviluppatori.
- Facilitare le operazioni di manutenzione delle applicazioni.
- Migliorare le capacità della piattaforma tecnologica.

Queste linee guida non sono obbligatorie; ogni sviluppatore può adottarle in modo flessibile durante lo sviluppo del progetto Qlik.

Tuttavia, la ragione per non conformarsi alle linee guida deve essere valutata e giustificata.

Queste linee guida saranno valide per tutti i nuovi progetti Qlik sviluppati in Intesa - San Paolo dal rilascio di queste linee guida.

Per tutte le applicazioni Qlik già esistenti, in caso di necessità di ripristino o di manutenzione delle stesse, dovrebbe essere presa in considerazione l'adozione di alcuni o di tutti i principi definiti nelle linee guida, alla luce dell'utilità e dei vantaggi che questo porterà alla qualità dell'applicazione.

Al fine di facilitare l'adozione di queste linee guida, Intesa - San Paolo fornisce un 'kit' di applicazioni standard parametriche che:

- Aiuteranno gli sviluppatori.
- Ridurranno il tempo necessario per lo sviluppo del progetto durante la sua implementazione.
- Faciliteranno l'attività di manutenzione dei progetti.

Questo documento è costituito dalla documentazione che definisce lo standard metodologico per lo sviluppo di applicazioni BI sulla piattaforma Qlik e fornisce una descrizione dettagliata di come utilizzare le applicazioni parametriche introdotte.

Gli standard metodologici per lo sviluppo sono:

- Definizione dell'architettura di riferimento:
 - livelli, struttura delle cartelle dei progetti e le diverse componenti dell'applicazione.
- Definizione dello standard per lo sviluppo:
 - strutturazione e organizzazione dello script, linee guida per l'utilizzo delle variabili in Presentation, gestione dei metadati, convenzioni di denominazione per variabili e parametri e procedure di controllo qualità.
- Linee guida per la progettazione del layout: principi di visual design e delle best practice

Questo documento si riferisce alla definizione dell'architettura, alla descrizione dettagliata dei diversi livelli, dell'organizzazione di directory dei progetti e dei componenti dell'applicazione.

2 Architetture applicative

Le seguenti pagine contengono, in primo luogo, diverse architetture di dati che possono essere implementate come parte delle soluzioni Qlik.

Queste pagine non rappresentano tutte le possibili architetture. Infatti, rappresentano solo una piccola parte di quelle possibili.

Infatti rappresentano i concetti più utilizzati (ad eccezione dell'architettura ad un solo livello) per promuovere la scalabilità, il riutilizzo e la coerenza nello sviluppo di applicazioni Qlik.

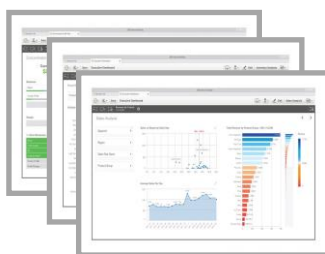
Di seguito viene indicata l'architettura applicativa di riferimento che Intesa - San Paolo adotta come standard per lo sviluppo delle soluzioni Qlik.

Viene, infine, delineata l'architettura applicativa a cui Intesa - San Paolo intende evolvere.

2.1 Architetture applicative progettuali

2.1.1 Architettura ad un solo livello (Diretta)

Presentation Layer



Utilizzato per l'accesso diretto alle fonti alimentanti.
Particolarmente indicato per la parte di self service su base dati sorgenti strutturate e su ambiti big data.

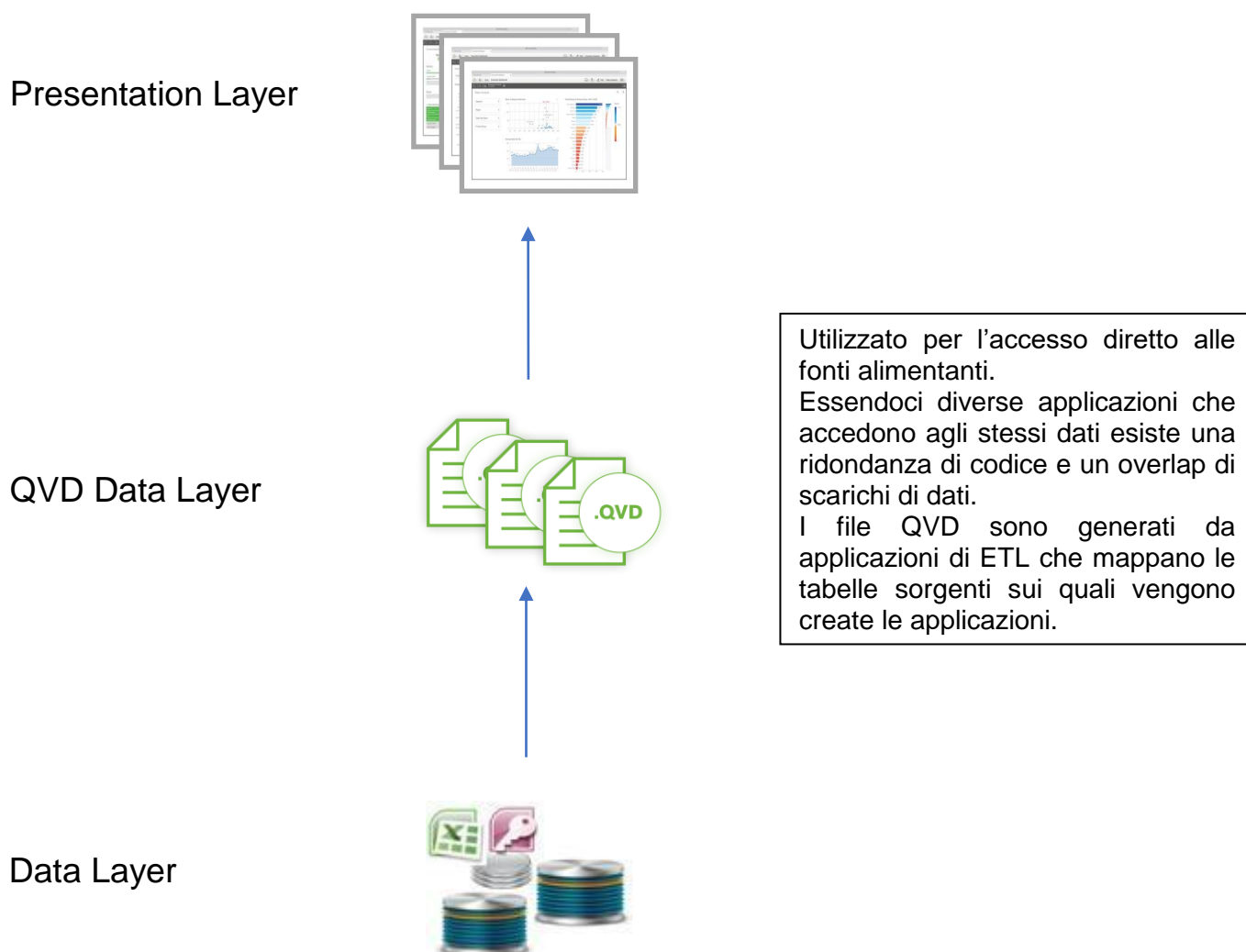
Data Layer



Questa architettura coinvolge solo query dirette verso i database di origine. Questa è la soluzione indicata per tutte le analisi di self service che consentono agli utenti, a seconda del rispettivo livello di accesso alle piattaforme, di accedere alle fonti dati attraverso le connessioni disponibili.

Questo approccio è quindi adatto per progetti "Big Data" con funzionalità «Accesso diretto» o con generazione di «App on Demand».

2.1.2 Architettura a 2 livelli con QVD



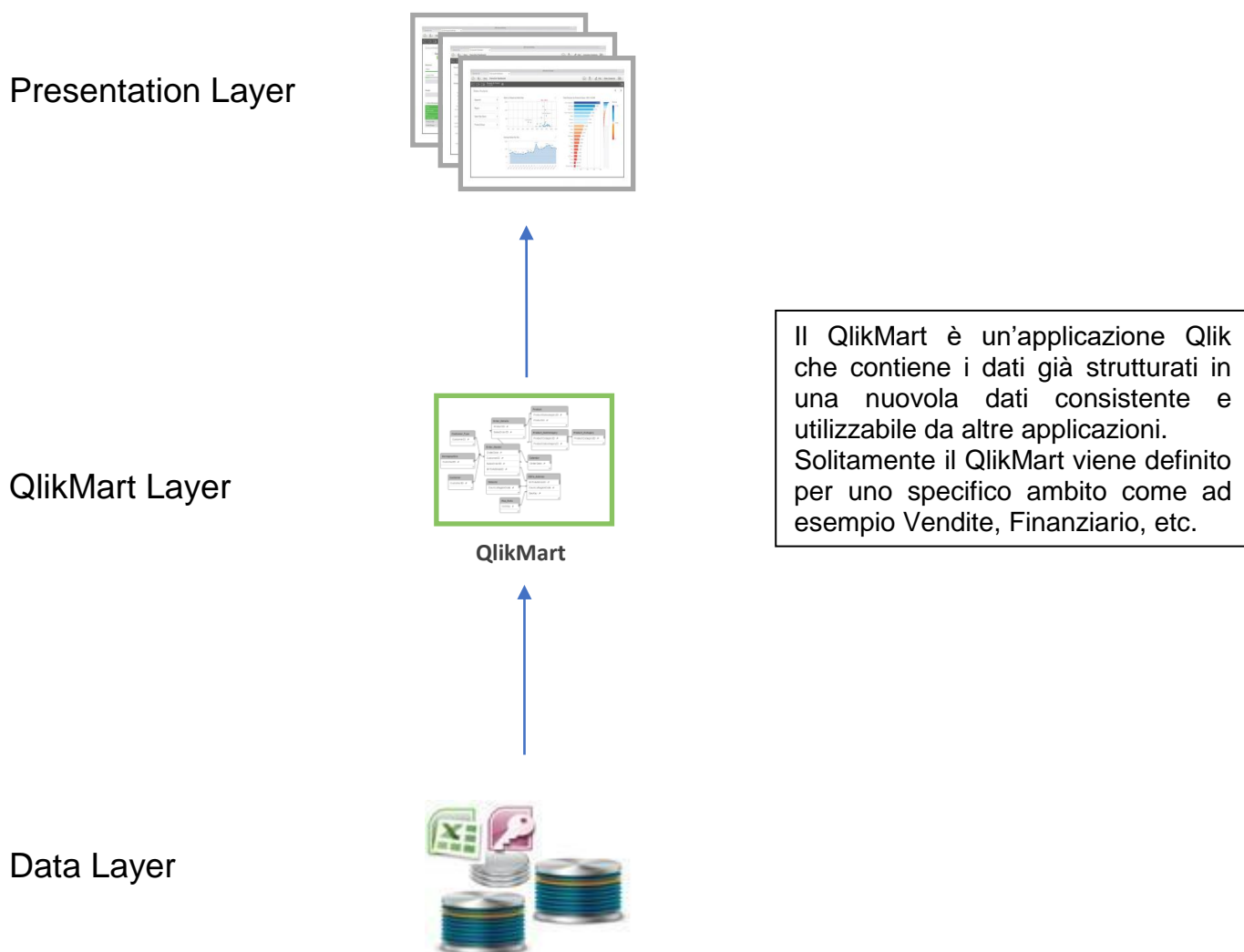
Questa architettura include l'uso di file QVD per un secondo livello di dati. A livello di presentation le applicazioni genereranno modelli di dati da diversi file QVD, ma non avranno bisogno di estrarre direttamente dai database di origine. Ciò aiuta a promuovere il riutilizzo e la coerenza tra le diverse app.

Questo tipo di architettura consente di separare lo sviluppo dell'applicazione di presentazione dai dati dei database di origine.

Uno svantaggio di questa architettura è la definizione del cloud associativo nelle applicazioni di presentazione: questo è fondamentale per la manutenzione applicativa, in particolare per il modello di dati ad alta complessità.

Questa architettura potrebbe essere utilizzata per lo sviluppo di progetti con requisiti di ricondizionamento intraday (quasi in tempo reale). Si consiglia quando le origini dati sono normalizzate e ottimizzate per il layer di presentation.

2.1.3 Architettura a 2 livelli con QVF (QlikMart)



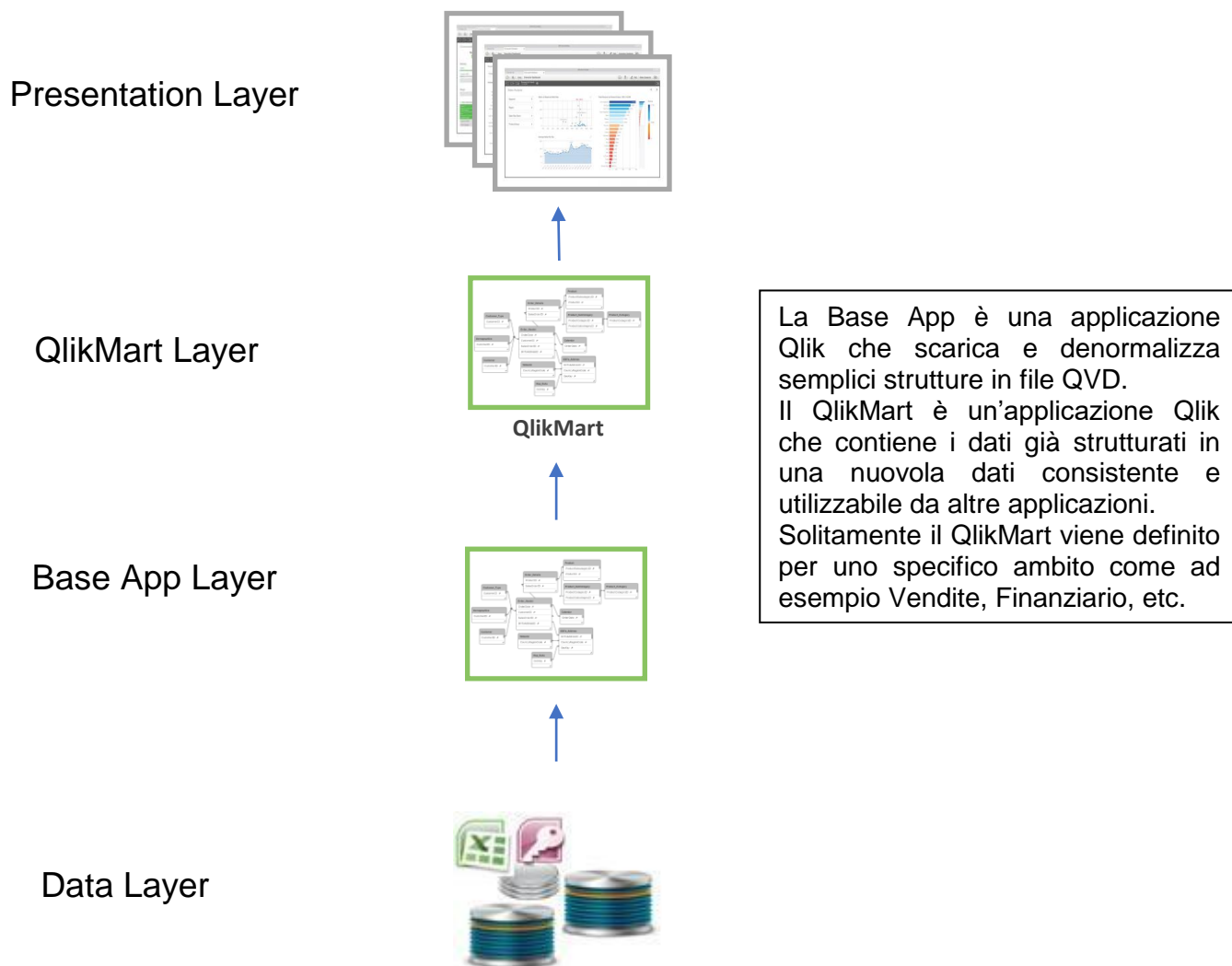
Questa architettura include l'uso di file di applicazioni "QlikMart" come secondo livello di dati. Questi QlikMarts sono applicazioni Qlik senza il layer di presentation.

Funzionano come modelli di dati che possono essere caricati in un'applicazione di presentazione con una singola riga di codice (chiamata carico binario).

Il principale handicap di questa architettura è rappresentato dalla "rigidità" della funzionalità "Carico binario": questa funzionalità carica l'intero modello di dati; se è necessario integrare il modello con altre tabelle, potrebbe essere difficile e non flessibile. Questa architettura non gestisce il caricamento incrementale da origini dati e la conservazione della cronologia o riutilizzo in molte applicazioni.

Questa architettura può essere utilizzata per lo sviluppo di progetti che comportano un volume elevato di dati con frequenti aggiornamenti. E' consigliata quando più di un'applicazione di presentazione è basata sullo stesso modello di dati.

2.1.4 Architettura a 3 livelli con QVF (Base App e QlikMart)



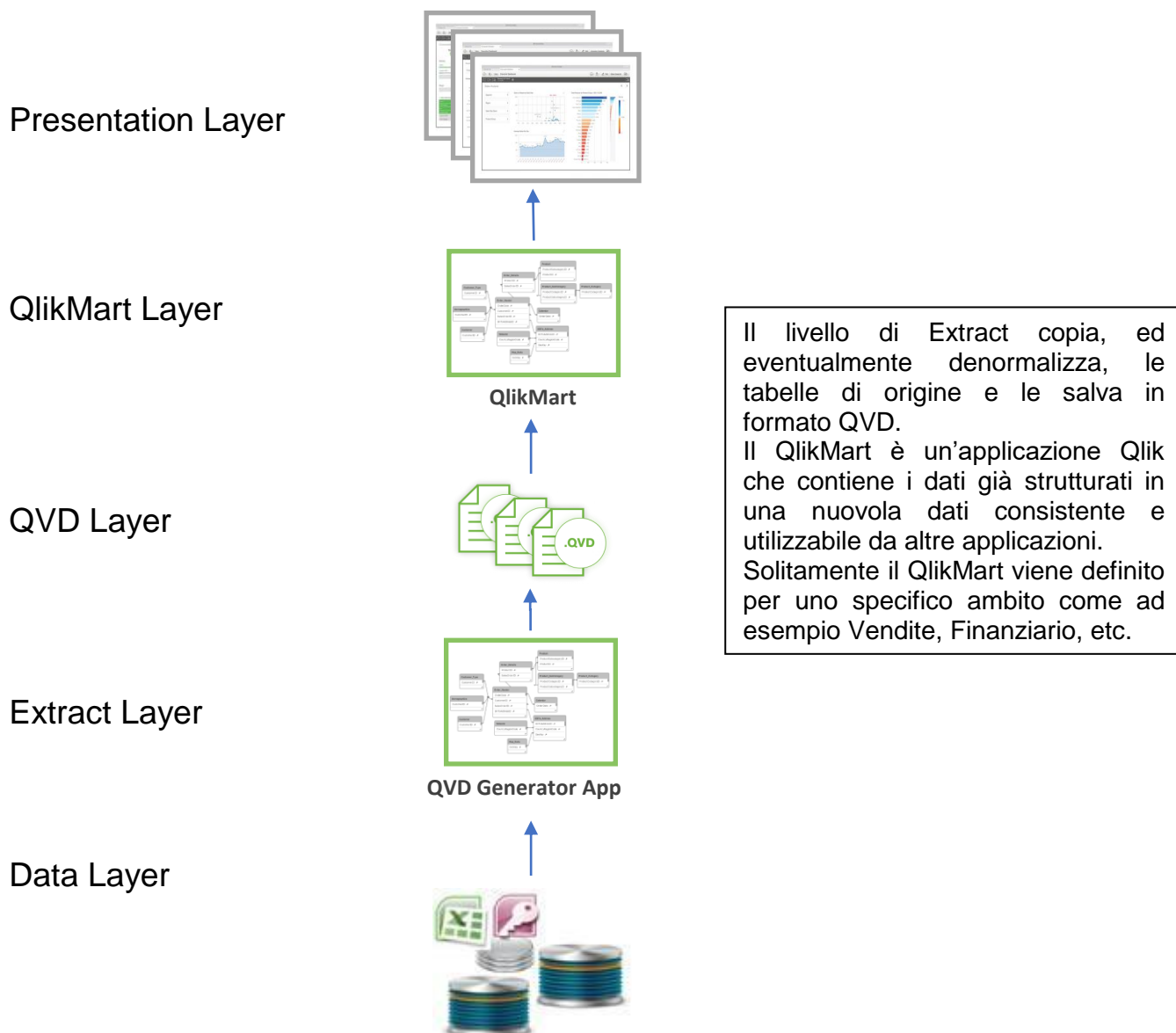
Questa architettura include l'uso di file QVF come secondo e terzo livello di dati. Ciò consente al primo strato QVF di contenere modelli di dati relativamente semplici con un livello più normalizzato del database di origine, con poche attività di aggregazione o trasformazione.

Nel livello QlikMart avvengono le trasformazioni e le aggregazioni. Questa architettura avrà lo scripting più complesso a livello di QlikMart poiché un intero modello di dati viene caricato dal livello precedente prima di attivare le trasformazioni sui dati.

Questa architettura può essere utilizzata per lo sviluppo di progetti che comportano un volume elevato di dati con frequenti aggiornamenti. Si consiglia quando più di un'applicazione di presentazione è basata sullo stesso modello di dati. Rispetto al singolo modello QlikMart, in alcuni

casi, il primo livello di QVF consente di gestire l'accesso alle fonti di dati utilizzando la segregazione.

2.1.5 Architettura a 3 livelli mista QVD e QVF



Questa architettura include l'utilizzo dei file QVD come secondo livello dati, basato sul quale viene generato il livello QlikMart. Ciò consente alle applicazioni del livello di presentation di essere dei «Binary Load» del livello precedente. Rispetto al precedente modello QlikMart, il primo strato di QVD consente la gestione dei caricamenti incrementali e la conservazione della cronologia delle informazioni. Rimane la rigidità della funzionalità «Binary Load» nelle applicazioni di presentazione; Questa architettura allunga la catena del processo limitando il vantaggio del QlikMart in termini di tempo di ricarica. Resta però valida, ad esempio, per lo sviluppo di progetti con più di una presentazione con lo stesso modello di dati.

2.2 Architetture di progetto di riferimento

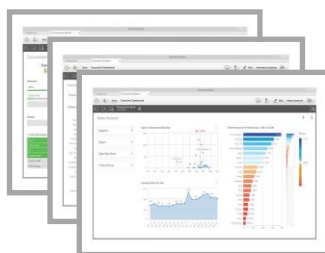
L'architettura applicativa di riferimento per i progetti che Intesa adotta, all'interno dell'architettura Lambda, per lo sviluppo standard dei progetti Qlik sono due:

- Architettura ad un solo livello (Diretta)
- Architettura a due livelli (QVD e QVF)

Vediamo in dettaglio entrambe le architetture.

2.2.1 Architettura ad un solo livello (Diretta)

Presentation Layer



Data Layer



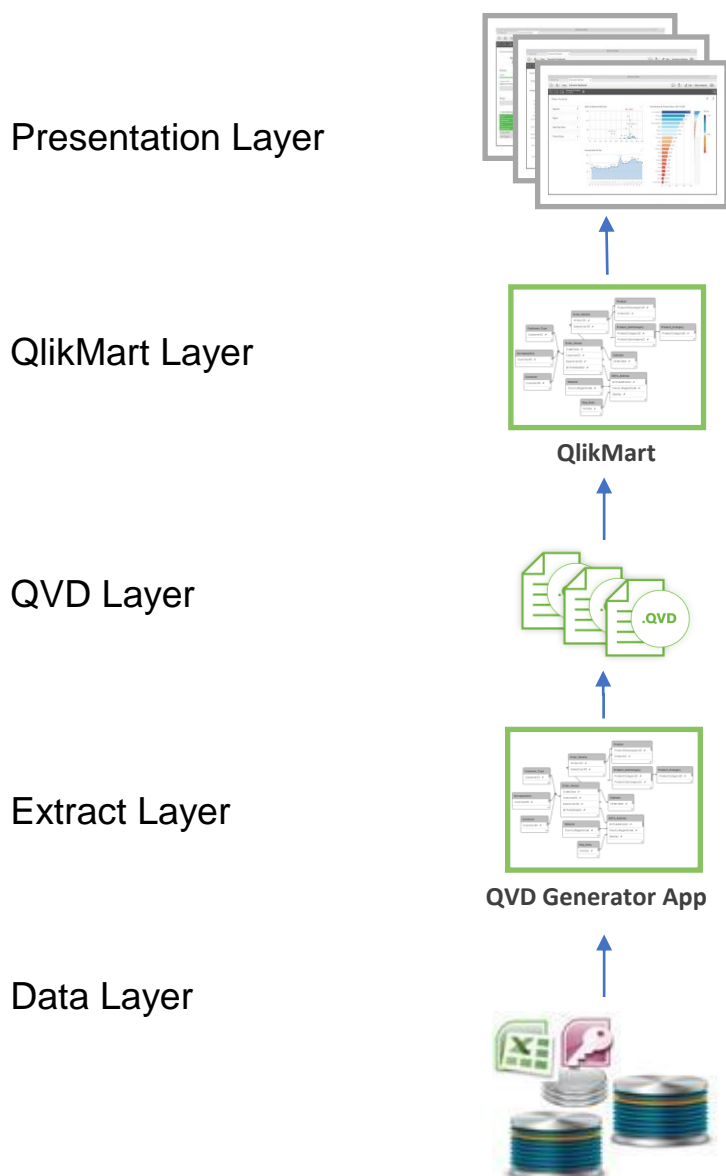
Questa struttura risponde alle esigenze di:

- Accesso al repository Big Data
- Flessibilità per l'analisi degli utenti
- BI self-service
- Analisi ad hoc e mashup di dati

A partire dalle origini dati, il livello Presentation legge direttamente i dati dalla sorgente all'applicazione Qlik. L'origine dati può essere uno o più database o file QVD o una combinazione di entrambi.

All'interno della infrastruttura Intesa – San Paolo i dati saranno disponibili attraverso il DSL (Data Service Layer) che esporrà, applicando gli opportuni criteri di sicurezza agli utenti, le diverse basi dati disponibili. Principalmente saranno disponibili dei datamart che copriranno le aree previste di analisi.

2.2.2 Architettura a 3 livelli mista QVD e QVF



A partire dalle origini dati il livello **Estrazione** genera un repository di file QVD che sono corrispondenza one-to-one delle tabelle DB di origine.

Il livello di **QlikMart** opera con la normalizzazione e l'aggregazione dei dati per definire il modello di dati associativo (cloud associativo), questo consente il riutilizzo (a volte parziale) in applicazioni di presentazione ottimizzando le prestazioni di reload.

Il livello di **Presentation** include solo lo sviluppo della visualizzazione dei dati e le funzionalità di data discovery.

Questa architettura offre questi vantaggi:

- Livello di **Estrazione**:
 - Permettere di separare lo sviluppo del progetto (modello di dati e presentazione) dalle fonti dati
 - Consentire la gestione del caricamento incrementale e il congelamento delle informazioni
 -
- Livello **QlikMart**:
 - le trasformazioni e le aggregazioni vengono definite a questo livello. Tutta la logica e le regole necessarie sono presenti in un unico punto.
 - Riutilizzo dello stesso modello per diverse applicazioni
 - Gestione di un volume elevato di dati con reload frequenti
- Livello di **Presentation**:
 - Applicazioni di front-end con il solo sviluppo della visualizzazione se ci si basa sull'architettura a 3 livelli
 - Applicazioni di front-end con trasformazione e aggregazione di dati e sviluppo della visualizzazione se ci si basa su un'architettura a livello singolo
 - Consente di separare lo sviluppo delle applicazioni di Presentation dal back end che possono essere gestiti da sviluppatori diversi.

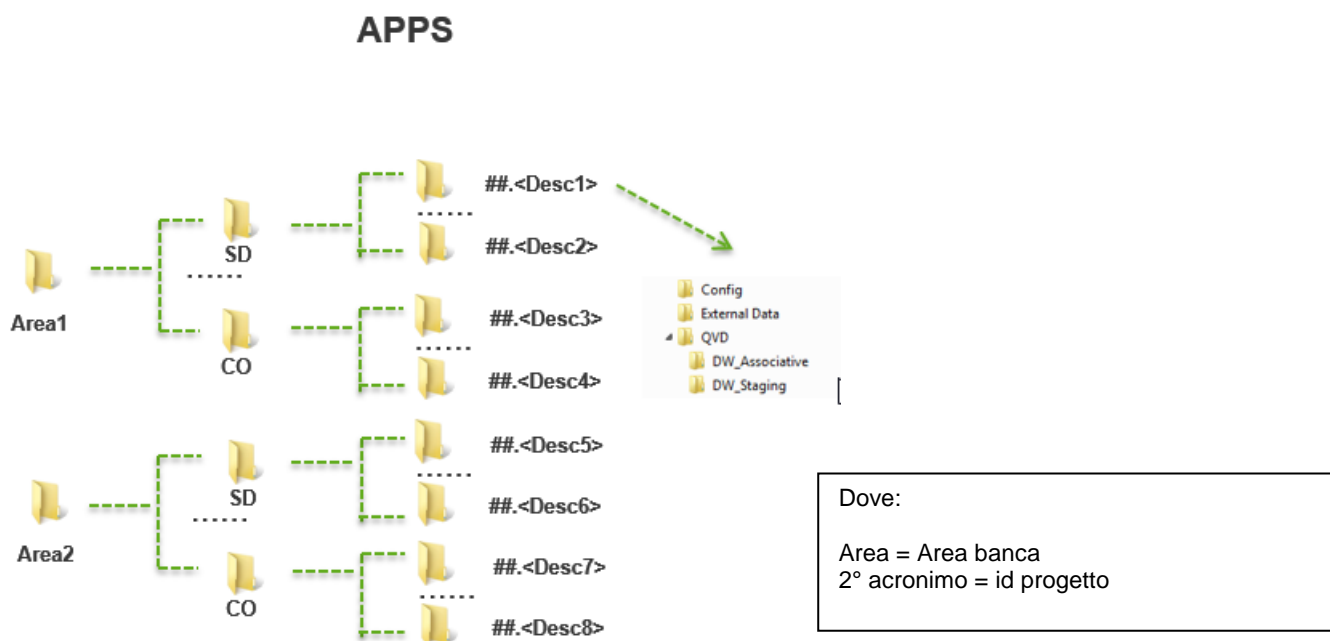
3 Architettura di riferimento

Le pagine seguenti descrivono come implementare l'architettura di riferimento definita precedentemente in termini di:

- Scopo dei componenti dell'applicazione
- Struttura della cartella di riferimento e descrizione dei contenuti
- Naming convention di cartelle e file
- Specifiche e utilizzo development framework per i progetti
- Sviluppo di applicazioni in self service

3.1 Cartella applicativa di riferimento

La struttura della cartella di riferimento è la seguente:



La struttura della directory delle applicazioni si riferisce agli sviluppi dei documenti di Qlik ed è strutturata per area funzionale. La convenzione di naming per questo livello di directory è un acronimo a 3 caratteri.

Esempi di aree funzionali sono:

Area Funzionale (3 caratteri)	Descrizione
COP	Customer operations
MKT	Marketing
SAL3	Sales

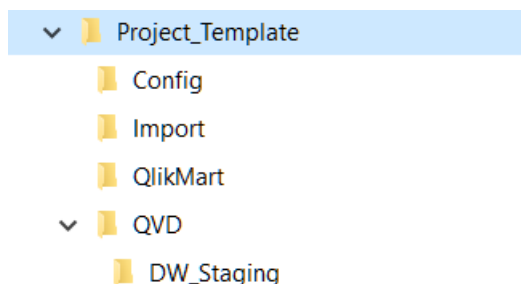
All'interno di ogni cartella definite dall'area funzionale sono presenti le cartelle di progetto.
La naming convention delle cartelle di progetto è: **##.[Breve descrizione del progetto]** dove:

= Numero incrementale di 2 cifre

[Breve descrizione del progetto] = Descrizione del progetto in massimo 15 caratteri

Esempio: **01.MyTestPage**

Ogni cartella di progetto contiene la seguente struttura di cartelle:



Le diverse cartelle conterranno:

- **QVD:** File QVD generati dai processi di ETL
 - **DW_Staging:** cartella contenente i file QVD generate dalle applicazioni di ETL01; questa cartella potrebbe contenere delle sottocartelle ad esempio per le Anagrafiche o degli anni per dati segmentati
- **Config:** cartella che contiene i file di settings ETL01_Setting.xls ed eventuali altri file di configurazione
 - **ETL Audit Log:** contiene file QVD generate dal processo di ETL con dati di monitoring delle esecuzioni;
 - **Logos:** cartella utilizzata come backup contenente loghi, immagini, icone, etc. utilizzati nelle app
- **Import:** cartella contenente file di appoggio provenienti da altri sistemi o manuali
- **Qlik Mart:** cartella contenente i file QlikMart. Queste sono applicazioni che contengono una nuova dati completa e che possono essere utilizzate come fonte dati per i documenti di presentation

Ogni progetto può avere diverse fonti:

- Database sorgenti
- Database locali (QlikMart e QVD)
- Flat file

Nel caso in cui il progetto richieda dati da un database locale il processo di ETL01 salverà i file nella cartella DW_Staging con la seguente naming convention:

- In caso di reload incrementale o completo:
 - o *[sorgente]_[E]_[nome tabella originale].QVD*
- In caso di storicizzazione: *[sorgente]_[E]_[nome tabella originale]_[tempo (a seconda della modalità storica scelta: YYYY, YYYYMM, YYYYMMDD)].QVD*

La naming convention per l'applicazione è la seguente:

[area funzionale]_[breve descrizione del progetto]_[nome del file].qv

Dove:

- **Area funzionale:** area funzionale (3 digits)
- **Breve descrizione del progetto:** una breve descrizione (max 15 digits)
- **Nome file:** ETL01 (Mandatory)

Esempio: **COP_MyTestPage_ETL01.qvf;**

Le linee guida per le estrazioni sono:

- Corrispondenza uno a uno delle tabelle sorgenti – QVD
- Estrazione consigliata in incrementale e/o segmentato per volume importanti
- Estrazione in * (tutti i campi della tabella)

I QlikMart devono essere salvati nella cartella QlikMart con la seguente naming convention:

[area funzionale]_[breve descrizione del progetto]_[file name].qv

Dove:

- **Area funzionale:** area funzionale (3 digits)
- **Breve descrizione del progetto:** una breve descrizione (max 15 digits)
- **Nome file:** MART (Mandatory)

Esempio: **COP_MyTestPage_MART.qvf;**

Il documento di presentation dovrà caricare:

- QVF dalla cartella QlikMart dalla area di *progetto*

La naming convention dell'applicazione sarà:

[area funzionale]_[breve descrizione del progetto]_[file name].qvf

Dove

- **Area funzionale:** area funzionale (3 digits)
- **Breve descrizione del progetto:** una breve descrizione (max 15 digits)
- **Nome file:** nome file (max 15 digits)
 - o **Il nome del file sarà modificato da Qlik Sense quando l'applicazione sarà caricata nel server.** In ogni caso l'adozione della naming convention è importante per mantenere una corretta overview delle app nella management console e negli streams

Esempio: **COP_MyTestPage_Dashboard.qvf;**

Se viene utilizzata una architettura single tier (diretta) il file qvf conterrà sia la data transformation che la parte di visualizzazione e la naming convention dovrà essere:

[area funzionale]_[breve descrizione del progetto]_[file name].qvf

Dove:

- **Area funzionale:** area funzionale (3 digits)
- **Breve descrizione del progetto:** una breve descrizione (max 15 digits)
- **Nome file:** DIRECT (Mandatory)

Example: **COP_MyTestPage_DIRECT.qvf;**

L'applicazione di presentation caricherà:

- Dati direttamente dalle fonti dati
- File flat dalla cartella di Import

La naming convention dell'applicazione sarà:

[area funzionale]_[breve descrizione del progetto]_[file name].qvf

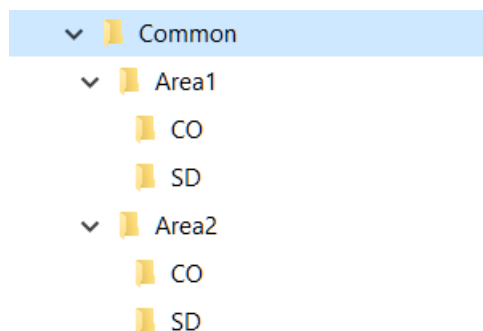
Dove:

- **Area funzionale:** area funzionale (3 digits)
- **Breve descrizione del progetto:** una breve descrizione (max 15 digits)
- **Nome file:** nome file (max 15 digits)
 - o **Il nome del file sarà modificato da Qlik Sense quando l'applicazione sarà caricata nel server.** In ogni caso l'adozione della naming convention è importante per mantenere una corretta overview delle app nella management console e negli streams

Esempio: **COP_MyTestPage_Dashboard.qvf**;

3.2 Gestione cartella QVD comuni (cross progetto)

Per poter gestire dati che sono utilizzati da diversi progetti contemporaneamente si definisce una cartella chiamata Common che, suddivisa per Area, conterrà al suo interno i file QVD condivisi. La gestione degli script di creazione di questi QVD sarà gestita centralmente da DSI e/o DOF che ne garantiranno l'aggiornamento, ognuno per la sua parte di competenza, secondo quanto pianificato in fase di implementazione dello script di generazione.



3.3 Development framework

Per facilitare l'adozione dell'architettura applicativa di progetto di cui sopra per lo sviluppo di applicazioni Qlik e massimizzare il vantaggio delle implementazioni, Intesa adotta un framework di sviluppo che contiene applicazioni Qlik parametriche e una struttura di directory conforme all'architettura di riferimento

Le pagine seguenti descrivono la funzionalità e l'uso di queste applicazioni. Vengono identificate due macro aree, a seconda degli scopi dei componenti che ne fanno parte:

- Back - End (Extraction Layer, Transformation Layer)
- Front - End (Presentation Layer)

3.3.1 Applicazioni Qlik naming convention

I file QVF devono seguire la seguente naming convention:

[area funzionale]_[breve descrizione del progetto]_[file name].qvf

Dove

- **Area funzionale:** area funzionale (3 digits)
- **Breve descrizione del progetto:** una breve descrizione (max 15 digits)
- **Nome file:** ETL01 per i processi di ETL e nome file per le presentation (max 15 digits)

- **Il nome del file sarà modificato da Qlik Sense quando l'applicazione sarà caricata nel server.** In ogni caso l'adozione della naming convention è importante per mantenere una corretta overview delle app nella management console e negli streams

Esempi:

COP_MyTestPage_ETL01.qvf,

COP_MyTestPage_MART.qvf

COP_MyTestPage_DIRECT.qvf

COP_MyTestPage_DashBoard.qvf;

3.3.2 Back-end

L'area macro Back-End è composta da tre applicazioni parametriche utilizzate per creare il cloud associativo dalle origini dati:

- ETL01.qvf
- QlikMart.qvf

La prima applicazione, ETL01.qvf, gestisce l'estrazione delle tabelle di origine e la loro archiviazione in formato file binario QVD. I QVD di output dell'applicazione ETL01 devono essere denominati con lo stesso nome della tabella di origine, con la seguente convenzione:

- In caso di reload incrementale o completo:
 - *[sorgente]_[E]_[nome tabella originale].QVD*
- In caso di storicizzazione: *[sorgente]_[E]_[nome tabella originale]_[tempo (a seconda della modalità storica scelta: YYYY, YYYYMM, YYYYMMDD)].QVD*

La seconda applicazione, QlikMart.qvf, utilizza i file QVD, l'output dell'applicazione precedente, per creare il cloud associativo che verrà inserito nelle applicazioni front-end tramite il carico in Binary.

Queste applicazioni, che si trovano nella cartella di riferimento rispettano le seguenti linee guida:

- rendere parametrico il maggior numero possibile di funzionalità di scripting
- utilizzare le variabili per rendere lo script dinamico e riutilizzabile
- estrarre dallo script il maggior numero possibile di funzioni: elenco di tabelle da estrarre, dizionario dei dati dell'applicazione, condizioni per l'estrazione delle tabelle
- gestione dei metadati
- congelamento del controllo dei processi dei file

- script di supporto con commenti appropriati per consentire una facile lettura

Le impostazioni e l'utilizzo specifici relativi a ciascuna applicazione sono esposti in dettaglio di seguito.

3.3.3 Front-end

L'area front-end comprende le applicazioni "Presentation", orientate all'analisi e alla navigazione dei dati.

A seconda dell'architettura utilizzata per il singolo progetto, possiamo avere due scenari:

1. L'applicazione di presentazione legge direttamente il Database di origine. Viene utilizzato con le applicazioni App-On Demand o Direct Discovery
2. L'applicazione di presentazione legge il file QVF di QlikMart. Solo un file può essere letto in questo modo

Quando viene utilizzato l'approccio 1), l'applicazione può avere uno script complesso all'interno per caricare i dati.

Quando viene utilizzato l'approccio 2), viene eseguito un carico binario da un'applicazione QlikMart da cui viene caricato tutto il cloud associativo.

Le applicazioni di "presentazione" si concentrano su metodi e tecniche di visualizzazione dei dati, nonché sulla massimizzazione delle prestazioni di navigazione.

Le prestazioni sono influenzate sia dalla struttura del cloud associativo sia dal numero e dalla struttura delle espressioni utilizzate negli oggetti Qlik che compongono le applicazioni "Presentazione".

Pertanto, le applicazioni front-end dovrebbero seguire le linee guida seguenti:

- Semplicità di script composta principalmente da istruzioni come Carica *, per importare il formato file QVD creato nell'area Back-End
- Uso di variabili per espressioni di calcolo, utilizzate in più oggetti
- Visualizzare oggetti sul foglio di visualizzazione in modo efficace
- Basso numero di oggetti sullo stesso foglio

Nel capitolo "Ergonomia e progettazione visiva" saranno presentati i principi di ergonomia e design visivo, questi principi sono utili per l'implementazione di applicazioni "Presentazione" che migliorano le capacità grafiche della piattaforma Qlik. Un esempio di applicazione basata su questi principi è allegato, per supportare sviluppatori e progettisti.

3.4 Architetture applicative

Questo capitolo descrive la struttura delle architetture, in relazione al flusso di informazioni.

A seconda del tipo di architettura selezionata Diretta o a 3 livelli è necessario adottare un approccio diverso.

Architettura a 3 livelli

Identifichiamo due processi ETL sequenziali basati su applicazioni QVF (ETL01.qvf, QlikMart.qvf) che memorizzano i dati nel file binario nativo di Qlik QVD, uno o più file qvf di QlikMart e una o più applicazioni per analizzare e navigare i dati importati dai file QVD creati nei processi ETL e / o da QlikMart per visualizzare le informazioni in modo chiaro ed efficace.

Questa architettura si basa quindi sul concetto di "triplo strato misto di QVD e QVF" ed è considerata una delle soluzioni adeguate alle esigenze di Intesa.

Il primo livello di ETL legge le origini dati e crea una prima serie di file in formato QVD, gestendo il carico incrementale, quando necessario. Il secondo livello utilizza un'applicazione Mart che carica i file QVD creati nel primo livello di ETL, normalizza e aggrega i dati per generare un file QlikMart.qvf che contiene la nuvola associativa. Una o più applicazioni di "Presentation" costituiscono il quarto e ultimo livello di questa architettura.

I livelli architetturali sono definiti in maniera sintetica come:

- Livello 0 - Sorgente
- Livello 1 – Estrazione
- Livello 2 - Trasformazione
- Livello 3 - Presentazione

Architettura ad un solo livello (Diretta)

Quando viene utilizzata questa architettura, esiste una sola applicazione che contiene sia ETL che elementi di presentazione.

Questo tipo di approccio è per progetti in cui è necessario leggere enormi quantità di dati o creare app "on demand".

All'interno dell'app sono presenti l'intero processo di lettura dei dati dal database di origine, la trasformazione dei dati, la creazione del cloud associativo e gli elementi di presentazione. Per un confronto con l'Architettura a 3 livelli tutti i livelli da 0 a 3 sono all'interno della stessa app.

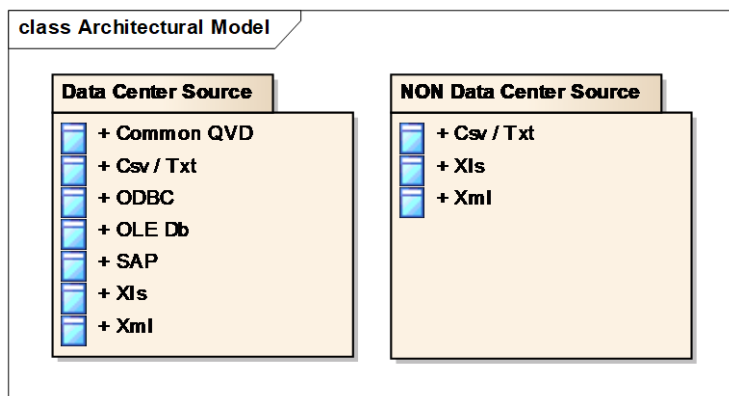
Il database di origine può essere anche file QVD.

3.4.1 livello "0" - data source

Il "Livello 0" è costituito da fonti esterne contenenti i dati che alimentano la piattaforma Qlik per ciascun progetto

- sources ODBC / OLE Db
- QVD files
- SAP Connector
- sources Xml
- sources Csv, Txt
- sources Xls

Le fonti sono divise tra quelle memorizzate nel Data Center di Intesa e quelle archiviate esternamente al Data Center, come mostrato nello schema seguente:



Le fonti nel Data Center sono accessibili tramite indirizzo IP (ODBC, OLE Db, SAP) con USER-ID e Password.

Le fonti situate al di fuori del Data Center sono disponibili solo tramite collegamento al Filesystem.

ODBC / OLE Db

Questa è la fonte più comune in cui vengono raccolte le informazioni. Si compone di database relazionali esterni come Oracle, SQL Server, MySql, Teradata, DB2, ecc.

Quando possibile, la connessione OLE DB è migliore di ODBC perché ha prestazioni migliori.

In alcuni casi (ad esempio per elevate complessità delle tabelle originali) può essere valutata attentamente la possibilità di creare una vista logica per separare i database.

QVD

QVD è una fonte gestita dal team centrale come descritto sopra

Gli scopi di questa fonte sono:

- generare un set di file QVD disponibili per gli sviluppatori senza accesso diretto al database di origine aziendale
- gestire centralmente le informazioni richieste trasversalmente da diversi progetti Qlik
- minimizzare l'accesso al database di origine aziendale
- minimizzare la ridondanza delle informazioni nel livello di estrazione

Sources Xml

Le fonti Xml richiedono particolare attenzione a causa della struttura di queste fonti. Pertanto, è necessario esaminare attentamente ciascun campo e identificare esattamente le informazioni da importare.

Sources Csv / Txt / Microsoft Excel

Le fonti consistevano nel file Csv o Txt caricati in Qlik molto semplicemente e non mostrano particolari punti di attenzione

3.4.2 Extraction level

Le attività sono eseguite dall'applicazione ETL01.qvf allegata al presente documento. Di solito, c'è solo un ETL01.qvf per ogni progetto. A volte è possibile configurare più di una applicazione ETL01.qvf per un progetto specifico se vi è una diversa tempistica di ricarica dei dati per una tabella dei fatti diversa.

L'utilizzo parametrico di questa applicazione è definito dalle impostazioni contenute nel file di Microsoft Excel "ETL01_Settings.xls".

In questi file vengono definiti i parametri e i metodi di estrazione dei dati dalle origini dati e i nomi dei file corrispondenti in formato QVD, in cui sono memorizzate le tabelle. Pertanto, l'applicazione ETL01.qvf è un motore parametrico per l'estrazione dei dati in cui i parametri per l'estrazione vengono gestiti esternamente dallo script.

Ogni applicazione ETL01.qvf interfaccia il proprio "ETL01_Settings.xls": la connessione tra i due file è 1: 1. Il collegamento è fatto dal nome: ad esempio:

COP_MyTestPage_ETL01.qvf → **COP_MyTestPage_ETL01_Settings.xls**

Non è necessario modificare lo script dell'applicazione ETL01.qvf poiché l'elenco delle origini dati e i metodi di estrazione sono definiti tramite file esterni Microsoft Excel. Tuttavia, esiste una sezione (Foglio "Estrazioni personalizzate") nella struttura del componente di scripting in cui gli sviluppatori possono creare il proprio script personalizzato per l'estrazione (es: estrazioni attraverso query che richiedono operazioni di unione tra le tabelle di fonti che non possono essere gestite con impostazioni parametriche sul file Microsoft Excel "ETL01_Settings.xls"). La funzionalità di cui sopra dovrebbe essere usata con parsimonia per non vanificare lo scopo dell'applicazione. Lo strato di QVD elementare è il risultato delle operazioni eseguite dall'applicazione ETL01.qvf. Il Qvd elementare di questo livello replica le fonti di dati sostanzialmente "as-is", con l'applicazione di alcune trasformazioni come la condizione Where o il partizionamento annuale o mensile delle tabelle dei fatti molto pesanti. Ciò significa che in questo livello le fonti dei dati di alimentazione vengono replicate in un rapporto 1-> 1 (1 tabella sorgente-> 1 file QVD) o, se necessario, diviso per periodi in un rapporto 1-> n (1 tabella sorgente -> n file QVD per ogni anno, ad esempio). A questo livello, le operazioni tra le tabelle di origine come Join, concatenate, mapping, non vengono eseguite. Lo scopo principale di questo livello è creare un'area estratta costituita da file in formato QVD pronti per essere utilizzati dall'applicazione QlikMart.qvf. La manutenzione e l'aggiornamento di questo livello è molto semplice e consente di disporre di un'area di staging dei dati separata da fonti originali, più facile accesso e accesso di lettura molto veloce. In questa area estratta è possibile effettuare una serie di controlli di qualità.



3.4.3 Specifiche di setup e uso

Questo paragrafo descrive le specifiche di installazione e uso l'applicazione ETL01.qvf e del relativo file Excel "ETL01_Settings.xls", che compongono il livello di architettura 1.

3.4.4 Impostazione delle variabili di base

L'impostazione delle variabili di base (principalmente variabili di percorso) e l'istruzione include sono definite in Tab Main dello script ETL01.qvf come mostrato di seguito a titolo di esempio:

```
1 SET ThousandSep='.';
2 SET DecimalSep='.';
3 SET MoneyThousandSep='.';
4 SET MoneyDecimalSep='.';
5 SET MoneyFormat='€ #.##0,00;-€ #.##0,00';
6 SET TimeFormat='hh:mm:ss';
7 SET DateFormat='DD/MM/YYYY';
8 SET TimestampFormat='DD/MM/YYYY hh:mm:ss[.fff]';
9 SET FirstWeekDay=0;
10 SET BrokenWeeks=0;
11 SET ReferenceDay=4;
12 SET FirstMonthOfYear=1;
13 SET CollationLocale='it-IT';
14 SET MonthNames='gen;feb;mar;apr;mag;giu;lug;ago;set;ott;nov;dic';
15 SET LongMonthNames='gennaio;febbraio;marzo;aprile;maggio;giugno;luglio;agosto;settembre;ottobre;novembre;diciembre';
16 SET DayNames='lun;mar;mer;gio;ven;sab;dom';
17 SET LongDayNames='lunedì;martedì;mercoledì;giovedì;venerdì;sabato;domenica';
18
19 ///----Development Kit
20 set vL.ProjectName = 'Timecards';
21 let vL.LibProjectName = 'lib://$(vL.ProjectName)';
22
23 //Basic Config
24 set vL.Prefix_QVD_Staging = 'St_'; //Prefix used for file format QVD created by ETL01 - Extractor.qvw in Staging area.
25 set vL.NumOfScriptLogStored = 10; //Number of processes observed through Log file created by default by ETL01 - Extractor.qvw application,
26 set vL.NumOfMonitoringLogStored = 200; //Number of processes observed in synthetic mode displayed in presentation of the ETL01 - Extractor.qvw
27
28 Let vL.ETLName = 'ETL01';
29 // Let vL.QvwName = 'ETL02.qvf';
30
31 Directory lib://Timecards/; //This need to be changed with the project name used in the QDFAdb.mdb
32
33 //Paths Config
34 let vL.Path_Settings = 'Config\';
35 let vL.Path_QVD_Log_ETL = 'Logs\';
36
37
38
```

Variabili di base sono:

- **vL.Prefix_QVD_Extracted**: Prefisso per il file QVD generato da ETL01.qvf (seguendo la convenzione di denominazione definita come prefisso) [Source]_E_)
- **vL.NumOfScriptLogStored**: Numero di processi osservati tramite il file di registro creato per impostazione predefinita dall'applicazione ETL01.qvf e visualizzato nella presentazione di ETL01.qvf e archiviato in formato file QVD ScriptLogETL01.qvd
- **vL.NumOfMonitoringLogStored**: Numero di processi osservati in modalità sintetica displayed nella presentazione di ETL01.qvf e memorizzati in formato file QVD MonitoringLogETL01.qvd
- **vL.Path_ExternalData** Path relativa per I file esterni
- **vL.Path_QVD_Extracted** Path relativa per I file QVD generate da ETL01.qvf
- **vL.Path_Settings** Path relativa per I file di configurazione
- **vL.Path_QVD_Log_ETL** Path relativa per I file di log dei QVD

3.4.4.1.1 Microsoft Excel file "ETL01_Settings.xls"

Il file Microsoft Excel "ETL01_Settings.xls" consente la standardizzazione parametrica della fase di estrazione, quindi è possibile evitare di modificare direttamente lo script dell'applicazione ETL01.qvf.

Il file contiene 3 fogli di lavoro:

Sheet	Description
Connections SetUp	Impostare le connessioni alle origini dati: includere le istruzioni di un file QVS per database relazionale o percorso per origini esterne
Tables SetUp	Tabelle da estrarre e specifiche di estrazione.
Fields SetUp	Campi di installazione (alias, campi calcolati)

Worksheet [Connections SetUp]

Il foglio di lavoro [Connections SetUp] consente di impostare le connessioni ai database relazionali, tramite la corrispondente istruzione di inclusione del file QVS memorizzato in 3.Include \ 3.ConnString della struttura di directory e / o di origini dati esterne con la sottodirectory specifica. *Nelle seguenti tabelle l'asterisco (*) indica che il campo è richiesto per il corretto funzionamento dello script ETL. L'asterisco tra virgolette (**) indica che il campo è richiesto in condizioni specifiche required on specific conditions*

Field	Description
Source ID (*)	<p>Formato:</p> <ul style="list-style-type: none"> Stringa di connessione; <p>Dirige la lettura sul connettore destro. Questi valori devono essere utilizzati come elenco di selezione dal campo [ID sorgente] del foglio [Impostazione tabella].</p> <p>IMPORTANT: È necessario creare un valore univoco di StringSource per ogni combinazione [SubPathQVD] / [SubPathExternalSources]</p>
Enable (*)	<p>Picklist:</p> <ul style="list-style-type: none"> Y N (o vuoto) <p>Y = connessione da eseguire N / empty = script non esegue la connessione</p>
Source Type (*)	<p>Picklist:</p> <ul style="list-style-type: none"> DB External ODS (con DSO SAP Connector) <p>Identifica il tipo di Datasources</p>
DB Type	<p>Solo per Source Type = 'DB' e il reload incrementale richiesto</p> <p>Picklist:</p> <p>ORA = Oracle</p>

		SS = MS SQL Server ACC = MS Access (serial date)
Connection String (*)		Include del file QVS contained connection string (ODBC/OLE DB) <i>Campo obbligatorio se Source Type="DB"</i>
SubPath - External Source		Formato: <ul style="list-style-type: none"> \Path Sub-directory dove salvare i files relative a sorgenti dati esterne (xls, txt) ..\8.Import\SubPathExternalSources Se vuoto il default è ..\8.Import\
SubPath - Store Staging		Formato: <ul style="list-style-type: none"> \Path Sub-directory dove salvare i file in formato QVD elementare (Extracted) ..\QVD\DW_Staging\SubPathQVD Se vuota è la root ..\QVD\DW_Staging_QVD \ In caso di segmentazione, è utile impostare una sottocartella a seconda dell'anno di segmentazione; per fare ciò la sottodirectory può essere riempita con la variabile \$(vL.Year) \
Description		Campo descrittivo

Esempio di compilazione:

Source ID	Enable	Source Type	DB Type	Connection String	SubPath - External Source	SubPath - Store Extracted	Description
RCO_VIEW	Y	DB	SS	\$(Include=..\3.include\3.connstring\RCO_VIEW.txt);		\$(vL.Year)\	
SQL	Y	DB	ORA	\$(Include=..\3.include\3.connstring\Connessione_ORACLE.qvs);			
Txt	Y	External			\Folder_DS_1		
Xls		External			\Folder_DS_2	\Folder to store	
Xls2	N	External				\Folder to store	
QVD	Y	External					

Esempio di Include \$(Include=..\3.include\3.connstring\Connessione_ORACLE.qvs);

Worksheet [Tables SetUp]

Il foglio di lavoro contiene l'elenco delle tabelle da estrarre e le specifiche dell'estrazione per ogni tabella

.

Campo	Descrizione
Source ID (*)	Picklist: <ul style="list-style-type: none"> Foglio: [Connections Setup] Campo: [Source ID] Identifica la source da leggere. Indirizza la lettura al connettore corretto.

	ATTENTION: i valori inseriti nel campo [ID sorgente] devono corrispondere ai valori nel foglio di lavoro [Impostazioni connessioni] Campo: [ID sorgente]
Table Name (*)	Formaot: <ul style="list-style-type: none"> Relational DB: [table name] External Sources: [File name].[File extension]
QVD Name (*)	E' il nome del file QVD in cui sarà salvata la tabella.
Description	Descrizione
Extraction Mode (*)	Picklist: <ul style="list-style-type: none"> FULL INCR empty <p>Tipo di estrazione. Se vuoto la tabella non viene estratta</p>
Store Partial	Picklist: <ul style="list-style-type: none"> Y N (o empty) <p>In combinazione con Extraction Mode = 'INCR' il requisito è quello di memorizzare la parte incrementale in una QVD separata con suffisso il campo di incrementale</p>
Where ('*')	Picklist: <ul style="list-style-type: none"> Y N (o empty) <p>Y = where condition applicata N / empty = where condition non applicata</p>
Where on SELECT / LOAD	Picklist: <ul style="list-style-type: none"> SELECT LOAD <p>Mostra se la clausola deve essere utilizzata nell'istruzione Select o Load (carico precedente).</p> <p>ATTENZIONE: Campo obbligatorio se viene usata la clausola Where</p>
Where Clause ('*')	Codice relativo alla clausola Where. Il codice utilizzato deve essere strutturato con la sintassi Qlik se la clausola Where è definita in "LOAD" o con una sintassi specifica della sorgente da cui viene estratta la tabella. <p>ATTENZIONE: Campo obbligatorio se viene usata la clausola Where</p>
Note	Field Note

Import Configuration - External Sources (**)	<p>Format:</p> <ul style="list-style-type: none"> (specific import statement) <p>Dichiarazioni specifiche di importazione in lingua Qlik per le fonti esterne (txt, xls, qvd). Esempio: LOAD * FROM Path \ File.ext specificImp;</p> <p>ATTENZIONE: campo obbligatorio se TypeDB = External</p>
To Segment	<p>Picklist:</p> <ul style="list-style-type: none"> Y N <p>'Y' mostra che la tabella deve essere congelata, ovvero la tabella originale viene estratta e archiviata in diversi formati di file QVD suddivisi per anno o anno o ogni giorno in base a una data di campo definita.</p> <p>In questo modo le tabelle vengono memorizzate in tanti file in formato QVD come i periodi (anno o anno). I file sono memorizzati con il nome [vs_Prefix_Name_Extracted] _ [nametable] _ [YEAR o YEARMONTH o YEARMONTHDAY].qvd.</p>
Segment Type (**)	<p>Picklist:</p> <ul style="list-style-type: none"> YEARLY MONTHLY DAILY <p>Mostra se il congelamento deve essere eseguito per periodo anno o mese o giorno</p> <p>Yearly: qvd stored [vs_Prefix_Name_Extracted]_[table name]_[YYYY].qvd</p> <p>Monthly: qvd stored [vs_Prefix_Name_Extracted]_[table name]_[YYYYMM].qvd</p> <p>Daily: qvd stored [vs_Prefix_Name_Extracted]_[table name]_[YYYYMMDD].qvd</p> <p>ATTENZIONE : Campo obbligatorio se To Segment = 'Y'</p>
Segment Field (**)	<p>Identifica il campo contenente la data (memorizzazione mensile e giornaliera) o l'anno (archiviazione annuale) Per il congelamento annuale se il campo è una data è necessario utilizzare la formula: year (fieldData)</p> <p>eg monthly, daily: fieldData eg annual: year(fieldData)</p> <p>ATTENZIONE : Campo obbligatorio se To Segment = 'Y'</p>
Segment From (**)	<p>Identifica la data (mensile, congelamento giornaliero) o l'anno (congelamento annuale) da cui inizia il congelamento della</p>

	<p>tabella</p> <p>es monthly: AddMonths(today(),-3) or 20160101</p> <p>es annual: (year(today()))-2) or 2010</p> <p>ATTENZIONE : Campo obbligatorio se To Segment = 'Y'</p>
Segment To ('*')	<p>Identifica la data (mensile, congelamento giornaliero) o l'anno (congelamento annuale) in cui deve finire il congelamento della tabella</p> <p>es monthly: today()</p> <p>es annual: year(today())</p> <p>ATTENZIONE : Campo obbligatorio se To Segment = 'Y'</p>
Incremental Load - Date Field ('*')	<p>Mostra il campo che nella tabella specifica la data di immissione del nuovo record.</p> <p>Il campo viene anche utilizzato per mostrare la data del tipo da utilizzare per il caricamento incrementale.</p> <p>Di seguito, questo campo sarà discusso in profondità.</p>
Incremental Load - Counter Filed	<p>Viene utilizzato per specificare il campo che in una tabella contiene il contatore di tipo numerico da utilizzare per il caricamento incrementale.</p> <p>Di seguito, questo campo sarà discusso in profondità</p>
Incremental Load - Update - Table Primary Key	<p>Mostra il campo che nella tabella specifica il primario per il carico di aggiornamento incrementale.</p>
Incremental Load - Delete - Table Primary Key	<p>Mostra il campo che nella tabella specifica il primario per il carico di cancellazione incrementale.</p>

Esempio di compilazione:

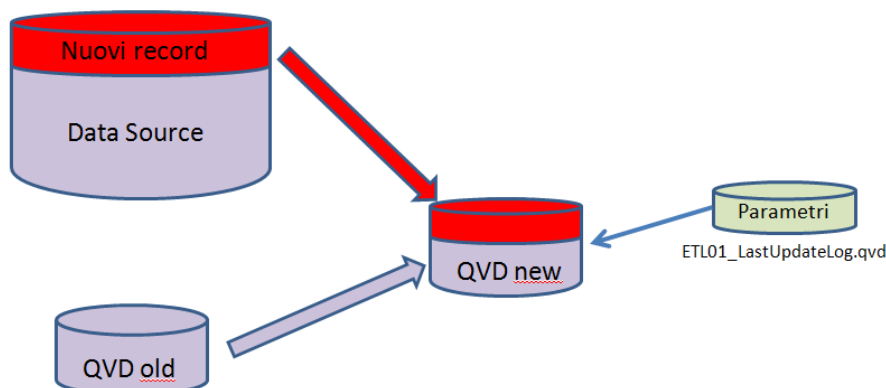
Source ID	Table Name	QVD Name	Description	Extraction Mode	Store Partial	Where	Where on SELECT / LOAD	Where Clause	Note	Import Configuration - External Sources
SQL	dbo.DemoTable	TableSQL		FULL		Y	SELECT	[Product]='A003'		
SQL	[dbo.Demo Table with space]	TableSQL		INCR	Y	N				
Txt	Test_1.txt	TableTXT		FULL						(ansi, fix, no labels, header is 0, record is line)
Xls	MS_Excel_1.xls	XLS1		FULL						(biff, embedded labels, table is DatiBilancio\$)
Xls	MS_Excel_1.xls	XLS1		FULL						(biff, embedded labels, table is Respo_Vendite\$)
Xls2	MS_Excel_2.xls	XLS2		FULL						(biff, embedded labels, table is Respo_Saldi\$)
qvd	QVD_1.qvd	XLS3		INCR		Y	LOAD	Customer<>'ABC'		(qvd)

Segue:

L	M	N	O	P	Q	R	S	T
To Segment	Segment Type	Segment Field	Segment From	Segment To	Incremental Load - Date Field	Incremental Load - Counter Filed	Incremental Load - Update - Table Primary Key	Incremental Load - Delete - Table Primary Key
					DATA_INOLTRO		[Primary ID]	[Primary ID]
Y	YEARLY	year(DATA_REG)	2008	year(today())				
Y	MONTHLY	DATA_REG	AddMonths(today(),-3)	Today()				
						COUNTER	[Primary ID]	[Primary ID]

La funzionalità di caricamento incrementale consente di specificare se una tabella di origine deve essere caricata interamente o applicare la logica incrementale, al fine di ottimizzare le prestazioni del processo.

Questa funzionalità può essere utilizzata se nella tabella per caricare c'è un campo che consente di identificare nuovi record dal carico precedente.



Il campo che identifica i nuovi record può essere una data o un contatore numerico. Per ogni caricamento, il valore massimo di questo campo viene memorizzato in una tabella di sistema e utilizzato come filtro durante i successivi caricamenti incrementali.

A livello di script, la modalità di caricamento è determinata dal valore (FULL o INCR) del campo Modalità di estrazione

Il carico incrementale potrebbe essere:

- INSERT: caricamento di nuovi record
- UPDATE: caricamento di nuovi record e record aggiornati
- DELETE: caricamento di nuovi record, record aggiornati e record cancellati

INSERT

Per abilitare il caricamento incrementale su una tabella in modalità INSERT è necessario impostare nel foglio di lavoro [Impostazione tabelle] del campo ETL01_Settings.xls [Campo data caricamento incrementale] (se il campo che identifica nuovi record è il tipo di dati) o il campo [Contatore carico incrementale Campo] (se il campo che identifica i nuovi record è di tipo numerico)

UPDATE and DELETE

Per abilitare il caricamento incrementale su una tabella in modalità UPDATE e DELETE è necessario impostare nel foglio di lavoro [Table Setup] del file ETL01_Settings.xls il campo [Incremental Load - Update - Table Primary Key] e [Incremental Load - Delete - Table Primary Key] con il valore o la concatenazione di valori che identificano univocamente i record

P	Q	R	S
Incremental Load - Date Field	Incremental Load - Counter Filed	Incremental Load - Update - Table Primary Key	Incremental Load - Delete - Table Primary Key
Table_Date_1		[Primary ID]	[Primary ID]
	ID_Table_1	[Primary ID]	[Primary ID]

WorkSheet [Fields SetUp]

Il worksheet [Fields SetUp] permette di gestire i campi estratti da ogni tabella. In particolare consente di:

- Decidi quali campi di una tabella / file estrarre
- Assegna un alias ai campi estratti
- Aggiungi nuovi campi calcolati
- Applicare una funzione nella fase di carico precedente

L'impostazione del foglio di lavoro [Fields SetUp] è facoltativa: se una tabella è menzionata nel foglio [Table SetUp] e non è dettagliata nel foglio di lavoro [Fields SetUp], l'applicazione ETL01.qvf esegue un LOAD * importando tutti i campi.

Field	Description
QVD Name (*)	Picklist: <ul style="list-style-type: none"> • [table name] worksheet [Tables Setup]
To Read (*)	Picklist: <ul style="list-style-type: none"> • Y • N (o vuoto) <p>Y = il campo viene estratto</p> <p>N / vuoto il campo non viene estratto</p>
Field (*)	Nome campo (è possibile usare *)
Alias	Se compilato l'alias viene applicato
To Load Preceding	Picklist: <ul style="list-style-type: none"> • Y • N (o vuoto) <p>Y = viene applicata la preceding load</p>
Field / Function -	Nome campo / Preceding Load.

Preceding Load	
Alias - Preceding Load	Se compilato l'alias viene applicato / Preceding Load
Note	Note

Esempio di compilazione:

QVD Name	To Read	Field	Alias	To Preceding Load	Field / Function - Preceding Load	Alias - Preceding Load	Note
TableTXT	n	[@1:2T]					
TableTXT	y	[@3:32T]	CONTRAENTE				
TableTXT		[@33:67T]					
TableTXT	y	[@157:159T]	COD_PROD				
TableTXT	y	[@160:167T]	tmp_mov	Y	Year(tmp_mov)	Y_Movimento	
XLS1	y	AREA					
XLS1	y	[RESP ZONA]		Y	num([RESP ZONA]/000)	[RESP 000]	
TableSQL	y	*		y			
TableSQL	y	date(campo_data)	Data_tmp	y	Data_tmp	Data	

4.1.2 ETL01.qvf Application

Il file ETL01.qvf ha le seguenti caratteristiche:

- Memorizza in formato QVD i dati caricati da diverse origini dati: database relazionali (testati con connettore OLE DB e ODBC) e / o fonti esterne
- Caricare fonti esterne dalla sottodirectory / memorizzare il QVD creato nella sottodirectory
- Impostare le singole tabelle da estrarre con l'istruzione import corrispondente
- Applicare una condizione di Where sulla tabella singola (in Select o in Load (Preceding Load))
- Congela le tabelle annuali, mensili o giornaliere
- Esegue un'estrazione incrementale (in insert per data o per contatore incrementale)
- Permette di specificare i campi da estrarre specificando un alias con la possibilità di eseguire una trasformazione nella Preceding Load
- Aggiungere campi calcolati (cn alcune limitazioni)
- Calcolare la data dell'ultimo aggiornamento di una tabella estratta
- Gestire delle estrazioni personalizzate
- Mostra tabelle riepilogative con informazioni relative ai processi di estrazione
- Monitorare le estrazioni (cardinalità, prestazioni, tabelle estratte, scripting)

Per la gestione di estrazioni personalizzate, nello scripting dell'applicazione ETL01.qvf c'è il foglio "Custom Extractions".



In questo foglio è possibile inserire il codice che consente ad esempio di estrarre una tabella risultante da una query SQL, che collega in join tabelle di fonti diverse. Altre estensioni personalizzate sono per istruzioni che non includono "LOAD ... FROM" o "SQL SELECT ... FROM" (ad esempio SAP OLAP Connector, importazione massiva di file flat). Questo tipo di estrazione non è gestito in ETL01_Settings.xls.

Nello script c'è un esempio di 3 blocchi che mostra come usare la [Sub CustomCode] con connessione parametrica al foglio [Connessioni] del file Microsoft Excel "ETL01_Settings.xls", per poi inserire del codice personalizzato e memorizzare / scaricare dalla memoria la tabella, preservare le funzioni di monitoraggio dell'applicazione ETL01.qvf (Extractor):

1. **Chiamata alla connessione:** call Fz_ConnectionCustom ('Name Connection') il nome della connessione nel campo [StringSource] di ETL01_Settings.xls
2. **Codice script della estrazione:** (eg: query SQL...)
3. **Store and Drop** con funzioni di Auditing: Call Fz_StoreTable('Name Table')

System Tables

Durante l'esecuzione di ETL01.qvf I seguenti file QVD possono essere creati se le relative chiamate di monitoring sono attive nello script (..\Config\ETL Audit Log\):

- ETL01_MonitoringLog:
 - *ETL_ExtractionCounter*: contatori
 - *ETL_Source*: origine dati
 - *ETL_ExtractionDuration*: durata esecuzione
 - *ETL_ExtractionMode*: modalità di estrazione
 - *ETL_StoredTableName*: nome QVD
 - *ETL_StorePath*: QVD path
 - *ETL_StoreQvdName*: nome file QVD

- *ETL_DropMode*: drop mode (system field)
- *ETL_RowNum*: numero di records nel file QVD
- *ETL_FieldNum*: number of fields nel file QVD
- *ETL_StoreDate*: store date nel file QVD
- *ETL_StoreTime*: store time nel file QVD
- ETL01_ScriptLog:
 - *ETL_ScriptCounter*: contatore
 - *ETL_Key_ETL01Timestamp*: timestamp inizio processo
 - *ETL_ScriptLog_TimeStamp*: timestamp inizio logging
 - *ETL_ScriptLog_ScriptRows*: righe di script
 - *ETL_ScriptLog_RowNo*: riga di script
- ETL01_TraceLog:
 - *ETL_Key_ETL01Timestamp*: timestamp inizio processo
 - *ETL_TraceDate*: data di trace
 - *ETL_TraceMsg*: messaggio di trace definite nello script
- ETL01_MonitoringLogSintesi:
 - *ETL_Key_ETL01Timestamp*: timestamp inizio processo
 - *ETL_StartTimeStamp*: timestamp inizio estrazione
 - *ETL_EndTimeStamp*: timestamp fine estrazione
 - *ETL_Duration*: durata processo
 - *ETL_TotalFieldNum*: numero di campi estratti
 - *ETL_TotalNoOfTable*: numero di tabelle estratte
 - *ETL_TotalRowNo*: numero of records estratti

3.4.5 Transformation level

Le operazioni vengono eseguite dall'applicazione QlikMart.qvf, se si utilizza l'architettura a 3 livelli. Tutti i passaggi necessari come trasformare, normalizzare e aggregare i dati memorizzati nel file QVD creato dall'applicazione ETL01, in modo funzionale per creare il cloud associativo utilizzato dalle applicazioni di "Presentation" viene fatto sull'app QlikMart.

La stessa base dati, quindi, può creare nuvole associative strutturate in modo diverso a seconda delle finalità e dei metodi di navigazione delle informazioni richieste nelle applicazioni di "Presentation".

Tuttavia, nell'implementazione di uno script specifico, è richiesto agli sviluppatori di rispettare gli standard di sviluppo.

Un principio molto importante è il seguente: l'applicazione QlikMart.qvf deve considerare tutte le possibili trasformazioni, aggregazioni, formattazioni necessarie per l'applicazione "Presentazione", che devono semplicemente ottenere le informazioni da questo secondo livello senza eseguire altre elaborazioni.

Pertanto, la parte principale della logica che guida il cloud associativo del documento finale Qvf deve essere realizzata a questo livello.

Per prestare particolare attenzione alle prestazioni nella navigazione nelle applicazioni di "Presentation", vengono identificate le seguenti linee guida per strutturare la nuvola associativa in particolare durante lo sviluppo di progetti che coinvolgono grandi volumi di dati:

- Mancanza di chiavi sintetiche native di Qlik: l'associazione tra le tabelle di un data cloud deve essere gestita attraverso i campi chiave definiti all'interno dei file QVD di livello 2.
- Adottare il più possibile strutture di cloud associativo come Star Schema: questo modello consente, a differenza di schemi Snow Flake, di "accorciare" i percorsi associativi tra campi di diverse tabelle e tra campi utilizzati all'interno degli oggetti delle applicazioni "Presentation" (sia come dimensioni e come espressioni).
- Gestione dei campi non utilizzati nell'applicazione "Presentation": non importare le informazioni che sicuramente non sono oggetto di analisi e navigazione da parte degli utenti.
- Precalcolare le espressioni complicate replicate negli oggetti di "Presentation" all'interno dello script dell'applicazione QlikMart.qvf, in modo da renderle disponibili come campi semplici a questo livello.

3.4.6 Specifiche di setup e uso

Questo paragrafo descrive le specifiche di installazione e uso dell'applicazione QlikMart.qvf che compongono il livello di architettura 2.

3.4.6.1 SetUp Base Variables

L'impostazione di Variabili Base (principalmente variabili di percorso) e l'istruzione include sono definite nella Scheda Principale dello script QLIK MART.qvf come mostrato di seguito.

```
////---Development Kit
//Basic Config
set vL.Prefix_QVD_Staging      = 'St_';      //Prefix used for file format QVD created by ETL01 - Extractor.qvw in Staging area.
set vL.NumOfScriptLogStored    = 10;          //Number of processes observed through Log file created by default by ETL01 - Extractor.qvw application,
set vL.NumOfMonitoringLogStored = 200;        //Number of processes observed in synthetic mode displayed in presentation of the ETL01 - Extractor.qvw

Let vL.ETLName                  = 'ETL01_Sales_EMEA';
// Let vL.QvwName                = 'ETL02.qvf';

Directory lib://Dev@QlikCentralNode/Sales\EMEA\;

//Paths Config
let vL.Path_ExternalData       = 'Import\';
let vL.Path_QVD_DwStaging      = 'QVD\DW_Staging\';
let vL.Path_Settings           = 'Config\';

let vL.Path_QVD_Log_ETL        = '$(vL.Path_Settings)' & '\ETL Audit Log\';
let vL.Path_Subs_ETL           = '$(vL.Path_Include)' & '\4.Sub\1.ETL01_Subs\';
```

Le Variabili di Base sono:

- **vL.Prefix_QVD_Extracted**: Prefisso dei file QVD generati da ETL01.qvf
- **vL.NumOfScriptLogStored**: Numero di processi osservati tramite il file di registro creato per impostazione predefinita dall'applicazione QLIK MART.qvf e visualizzato nella presentazione di QLIK MART.qvf e archiviato in formato file QVD ScriptLogQLIK MART.qvd
- **vL.Path_ExternalData** Path relativa alla cartella dei file esterni
- **vL.Path_QVD_Extracted** Path relativa ai file QVD generati in ETL01
- **vL.Path_Settings** Path relativa ai file di configurazione
- **vL.Path_QVD_Log_ETL** Path relativa per I log QVD

Le principali subroutines disponibili sono:

- **Fz_StoreTable**: subroutine per lo store e drop delle tabelle
 - Syntax: Fz_StoreTable ('TableName' [,DropMode,StorePath,StoreQvdName])
- **Fz_Calendar**: subroutine per la generazione del calendario
 - Call Fz_Calendar('Static Calendar' , ' St' , '01/08/2010' , '04/08/2012')

Le subroutines con prefisso Fz_Check e Fz_Trace sono disponibili per il trace log e dei controlli base sui dati.

3.4.7 Livello presentation

Il terzo livello consiste in applicazioni di "Presentation" orientate all'analisi e alla navigazione dei dati da parte degli utenti

Le elaborazioni integrate a questo livello devono essere considerate come eccezioni: lo scripting delle applicazioni "Presentation" dovrebbe contenere principalmente istruzioni per l'importazione dei dati dalle applicazioni di QlikMart. Le eccezioni di calcolo (lato script) devono essere valutate attentamente, perché se da un lato può essere utile per evitare di lavorare sul livello sottostante, dall'altro lato peggiorano le prestazioni di ricarica delle applicazioni e possono creare una perdita di flessibilità del cloud associativo.

Per prestare particolare attenzione alle prestazioni di navigazione delle applicazioni a questo livello, vengono identificate le seguenti linee guida per progettare gli oggetti di "Presentation", in particolare durante lo sviluppo di progetti che coinvolgono grandi volumi di dati è opportuno tenere conto di:

- Limitare il numero di oggetti sullo stesso foglio
- Limitare l'uso di Set Analysis nelle espressioni, in special modo se nidificate
- Limitare le dimensioni calcolate negli oggetti grafico
- Utilizzare le variabili per le espressioni replicate in più oggetti

Inoltre, al fine di massimizzare l'efficacia delle applicazioni di "Presentation", devono essere utilizzati i principi di ergonomia e design visivo.

3.5 Architettura ad un solo livello (Diretta)

Questa architettura consiste in una singola app che contiene tutti i livelli necessari per navigare e analizzare i dati.

Questo tipo di architettura viene utilizzato nel caso di un approccio specializzato all'analisi di big data, applicazioni altamente personalizzate, creazione di app on demand e altre applicazioni guidate da esigenze aziendali.

Lo scripting delle Architetture ad un livello dovrebbe contenere principalmente istruzioni per l'importazione dei dati dal database di origine, sia esso di origine database o QVD, direttamente o tramite connettori. Le eccezioni sui calcoli effettuati in script devono essere valutate attentamente, perché se da un lato può essere utile per evitare di lavorare sul livello sottostante, dall'altro lato peggiorano le prestazioni di ricarica delle applicazioni e possono creare una perdita di flessibilità del modello associativo.

Come per le applicazioni a 3 livelli è opportuno tenere conto dei seguenti suggerimenti:

- Limitare il numero di oggetti sullo stesso foglio
- Limitare l'uso di Set Analysis nelle espressioni, in special modo se nidificate
- Limitare le dimensioni calcolate negli oggetti grafico
- Utilizzare le variabili per le espressioni replicate in più oggetti

Inoltre, al fine di massimizzare l'efficacia delle applicazioni devono essere utilizzati i principi di ergonomia e design visivo.

3.6 Best practice per la creazione di app da utilizzare in Self Service

Come indicato sopra la scelta di utilizzare un accesso diretto è fondamentale nei casi di applicazioni in Self Service. Una applicazione creata in self service, oltre che per uso personale, può essere condivisa con altri utenti a direttamente, se l'utente che la crea ha i diritti per pubblicarla, o attraverso un processo di pubblicazione gestito da DOF.

In entrambi i casi è importante che vengano utilizzate delle best practice che consentano:

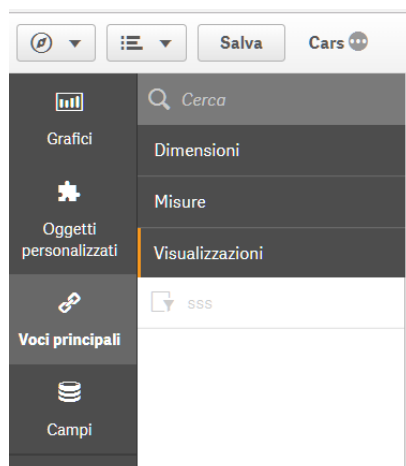
- Master Library (Voci Principali)
 - di impostare oggetti di base dell'applicazione come Dimensioni, Espressioni (Misure) e Visualizzazioni all'interno della Master Library
- Variabili
 - Impostare le Espressioni utilizzando le variabili per una gestione più dinamica rispetto all'utilizzo da parte di altri utilizzatori e per semplicità di manutenzione
- Data visualization
 - Seguire le regole della corretta Data Visualization per rappresentare nel miglior modo possibile le informazioni
- Riutilizzo applicazione
 - di poter riutilizzare l'applicazione creato in altre situazioni di analisi simili

A seguire andremo ad analizzare le diverse voci per definire in dettaglio i passi operativi necessari per il corretto approccio.

3.6.1.1 Master Library (Voci Principali)

Le misure, le espressioni e le visualizzazioni, siano esse grafici, tabelle, casella di filtro o altro, possono essere inserite all'interno della Master Library (Voci Principali) dell'applicazione.

La Master Library (Voci Principali) è un'area dell'applicazione accessibile durante la modifica di un foglio nel menù laterale sinistro:



e contiene tre voci Dimensioni, Misure e Visualizzazioni.

Il concetto di base è lo stesso per ognuna di queste e si riferisce al riutilizzo di oggetti che possono essere necessari in diversi punti di una applicazione salvandoli in una libreria comune. Questo garantisce che la definizione di un oggetto sia sempre la stessa indipendentemente da dove sia utilizzato all'interno dell'app e permette, modificando l'oggetto di libreria, di aggiornare tutte le istanze dello stesso garantendo la congruenza dell'applicazione. Per semplicità di seguito si farà riferimento a “grafico” intendendo indifferentemente un grafico o una tabella sia lineare che pivot, se non espressamente definito.

Vediamo ora le tre voci in dettaglio:

- Dimensioni
 - Mi permette di creare sia una singola Dimensione che un gruppo di Drill-down. Selezionando il campo dalla lista di quelli disponibili nella nuvola dati. Il campo può essere calcolato anche se Dimensioni calcolate possono essere causa di un maggiore tempo di calcolo se usate all'interno di grafici.

- Misure

- E' possibile definire una nuova misura che può essere una qualsiasi formula sui campi della nuvola dati. La misura avrà un nome e un'etichetta; quest'ultima può essere una formula che restituisce un unico valore su qualsiasi campo della nuvola dati e che sarà riportato come intestazione della colonna quando la misura sarà inserita all'interno di un grafico. Questo è importante per contestualizzare la misura stessa, ad esempio "Forecast 2018" dove 2018 è il valore del campo Anno selezionato, questo aiuta l'utilizzatore dell'app a capire lo scopo e quindi l'uso della misura selezionata.

Crea nuova misura

Espressione:

Nome:

Espressione etichetta

Descrizione:

Colore misura

Tag:

Annulla Crea

- Visualizzazioni principali

- È possibile creare una visualizzazione principale per poterla riutilizzare. Gli utenti di un'app pubblicata avranno accesso alle visualizzazioni principali, ma non potranno modificarle. L'uso delle visualizzazioni principali consente di creare una libreria di oggetti che conterranno dimensioni e misure consistenti per l'ambito di analisi scelto durante la creazione.
- Attraverso le visualizzazioni principali sarà possibile creare fogli di analisi coerenti e molto velocemente.
- Una visualizzazione principale viene creata o trascinando l'oggetto nell'area del menu laterale relativo alle visualizzazioni o con il click destro sopra all'oggetto che voglio inserire nella visualizzazione principale.
- È importante definire bene il nome della visualizzazione per garantire chiarezza agli utilizzatori. Anche nel caso in cui la visualizzazione sia creata dallo stesso utente che la deve utilizzare si consiglia, come linea guida, di definire in modo chiaro scopo e contenuto della stessa.

3.6.1.2 Variabili

All'interno di Qlik Sense è possibile definire delle variabili che possono contenere valori numerici o di testo ma anche formule di qualsiasi livello di complessità. L'uso delle variabili consente di rendere dinamica e facilmente manutenibile una applicazione Qlik Sense.

Una variabile è un contenitore a cui far riferimento attraverso il suo nome per avere accesso al suo contenuto.

Il processo più corretto per sfruttare la potenzialità della variabile può essere reso evidente con un esempio:

- Devo impostare una espressione che calcola il valore del forecast dell'anno in corso che devo utilizzare in diversi grafici e/o tabelle all'interno della mia applicazione e lo confronto con il fatturato dello stesso anno

L'approccio più corretto secondo le best practice è il seguente:

- Creare una variabile chiamata `v_Forecast_Anno_Corrente` con valore `sum(Forecast)`. Attenzione a non aggiungere un '=' davanti al sum
- Creare una variabile chiamata `v_Fatturato_Anno_Corrente` con valore `sum(Fatturato)`. Attenzione a non aggiungere un '=' davanti al sum
- Creare una nuova misura di Master Library chiamata "Forecast Anno Corrente" che ha come espressione `$(v_Forecast_Anno_Corrente)`
- Creare una nuova misura di Master Library chiamata "Fatturato Anno Corrente" che ha come espressione `$(v_Fatturato_Anno_Corrente)`
- Creare una nuova misura di Master Library chiamata "Fatturato vs Forecat Anno Corrente" che ha come espressione `$(v_Fatturato_Anno_Corrente)- $(v_Forecast_Anno_Corrente))/ $(v_Fatturato_Anno_Corrente)`
-

In questo modo posso utilizzare le tre misure create all'interno di vari oggetti per creare la mia applicazione.

Supponiamo che sia necessario inserire un divisore per 1.000 su entrambe le misure. Sarà sufficiente modificare le due variabili inserendo il divisore e tutte le formule dirette e derivate saranno impattate.

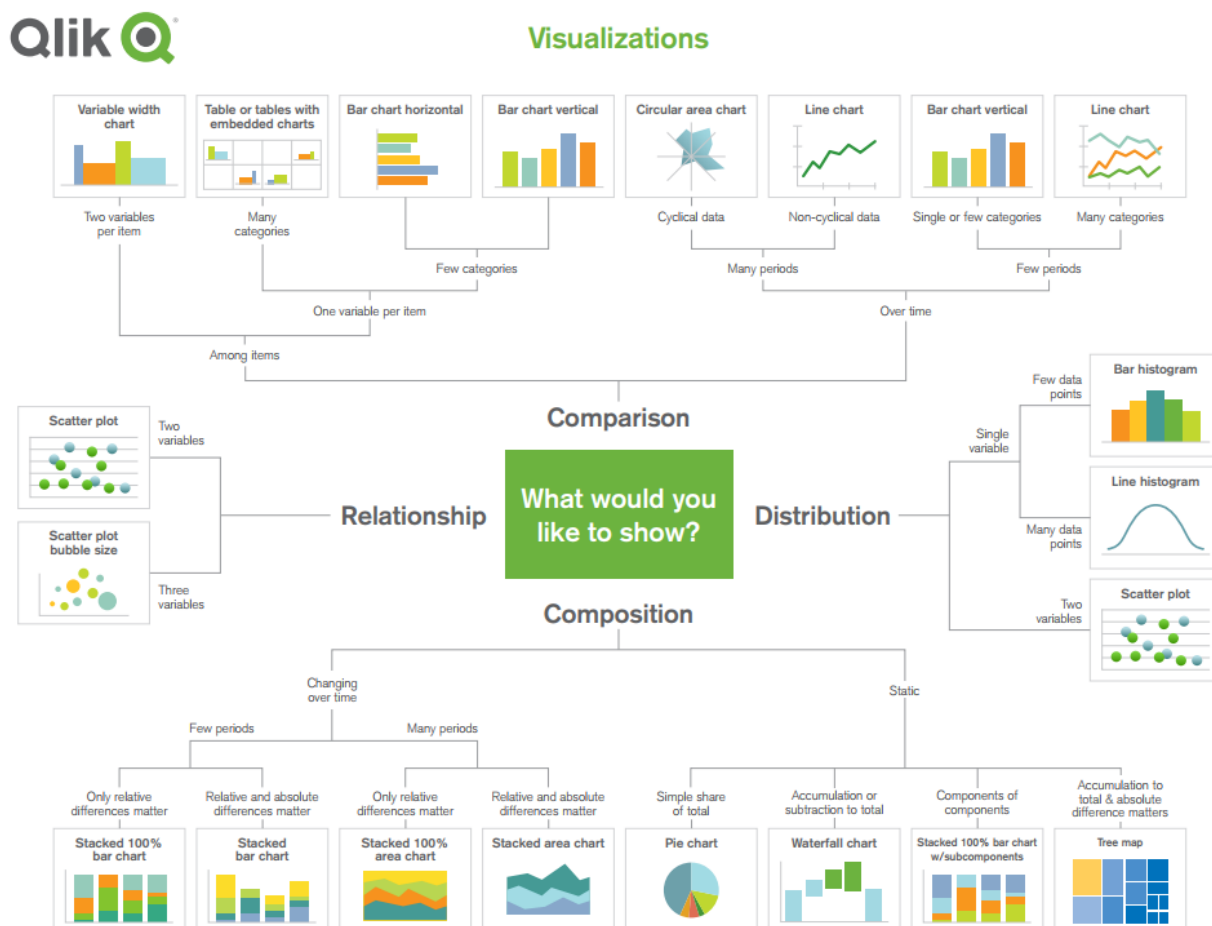
Questo permette un alto livello di adattabilità delle applicazioni anche in fase di modifica dei kpi individuati.

Le applicazioni costruite in questo modo possono facilmente diventare dei template per altre applicazioni dello stesso ambito di analisi. Sarà sufficiente adattare le variabili a nuovi campi del database o a formule più o meno complesse per avere già una base di partenza.

4 Data visualization

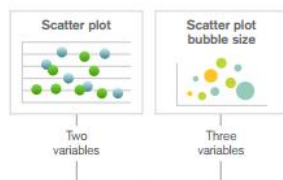
Un importante aspetto della visualizzazione dei dati passa attraverso la scelta della giusta scelta del grafico con cui rappresentare i dati stessi. Essendo la data visualization un'area molto ampia suggeriamo di accedere alla documentazione presente nei link riportati di seguito e alla ampia documentazione presente nel sito Qlik.com e blog.qlik.com.

Per semplicità riportiamo uno schema dei tipi di grafici più indicati da utilizzare a seconda del tipo di dato da visualizzare.





Relationship Visualizations

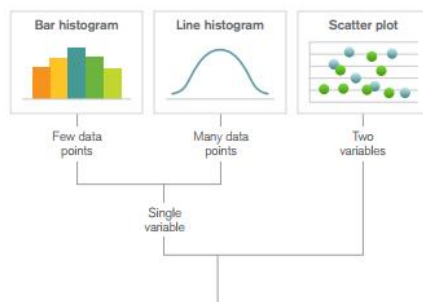


Relationship

Relationship charts are used to see the relationship between the data and can be used to find correlations, outliers and clusters of data. Common questions are "is there a correlation between advertising spend and sales for our products" or "how does expenses and income vary per region and what's the deviation".



Distribution Visualizations

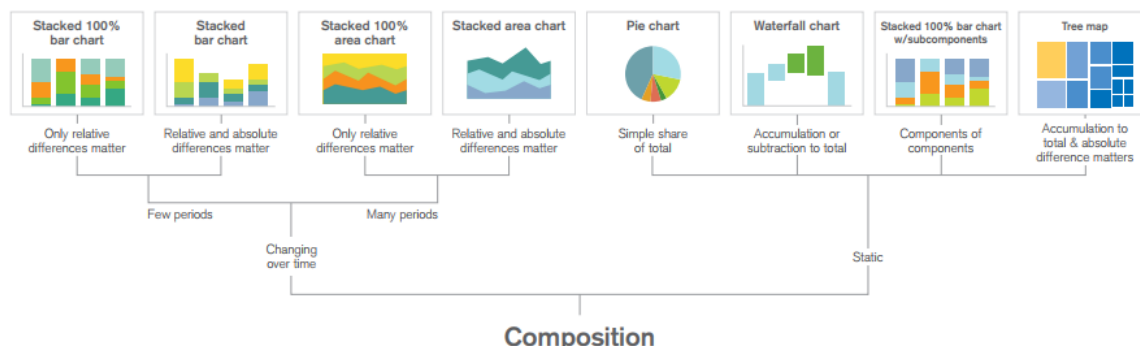


Distribution

Distribution charts are used to see how quantitative values are distributed along an axis from lowest to highest. Looking at the shape of the data a user can identify characteristics such as the range of values, central tendency, shape and outliers. It can be used to answer questions such as "number of customers per age group" or "how many days late are our payments".



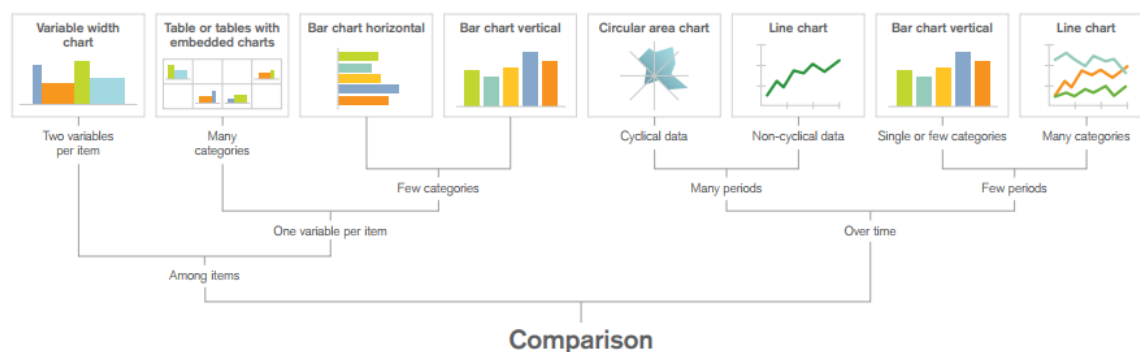
Composition Visualizations



Composition charts are used to see how a part compares to the whole and how a total value can be divided into shares. A composition charts shows the relative value, but some charts can also be used to show the absolute difference. The difference is between looking at percentage of total and value of total. Commons questions are “how big part of the market to we have in a region” or “what areas is our budget divided into”.



Comparison Visualizations



Comparison charts are used to compare the magnitude of values to each other and can be used to easily find the lowest and highest values in the data. It can also be used to compare current values versus old to see if the values are increasing or decreasing. Common questions are “what products sells best” and “how are our sales compared to last year”.

Link a documentazione sulla data visualization online:

<https://blog.qlik.com/visualizing-ideas-and-data>
http://blog.qlik.com

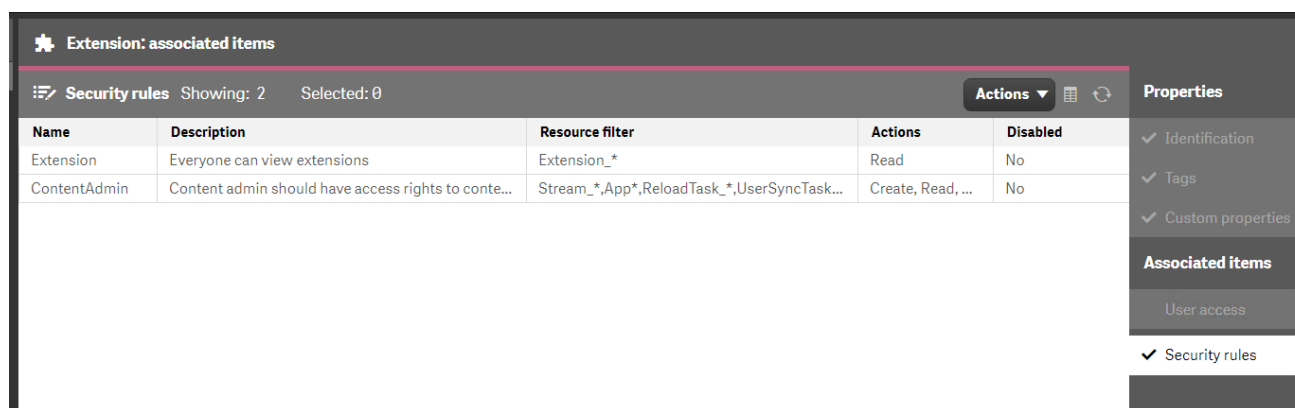
Documento che esplora le basi della visualizzazione efficace (in inglese):
User Experience Best Practices for Data Analysis.pdf

5 Gestione extensions

Per estendere le funzionalità di front end di Qlik Sense possono essere utilizzate le Extension certificate da Qlik che ampliano le capacità di visualizzazione, calcolo e interfacciamento delle applicazioni.

Le Extension certificate per la versione November 2017 sono:

- [gsVariable](#) — Consente all'utente di impostare il valore di una variabile utilizzando, bottoni, links, campi o sliders (Erik Wetterberg)
- [ShowHide Container](#) — Permette di nascondere dei grafici basandosi sui valori delle variabili (Fady Heiba)
- [Sheet Navigation + Actions for Qlik Sense](#) — Permette di inserire dei bottoni per gestire delle azioni di navigazione all'interno dell'app (Stefan Walther)
- [Tabs for Qlik Sense](#) — Consente di scegliere tra diverse visualizzazioni condividendo lo stesso spazio (Masaki Hamano)



Extension: associated items				
Security rules Showing: 2 Selected: 0 Actions				
Name	Description	Resource filter	Actions	Disabled
Extension	Everyone can view extensions	Extension_*	Read	No
ContentAdmin	Content admin should have access rights to conte...	Stream_*,App*,ReloadTask_*,UserSyncTask...	Create, Read, ...	No

Le Extensions sono proprie dell'ambiente in cui sono caricate e non vengono inglobate nelle app che le utilizzano.

E' necessario quindi che ogni extension sia trasportata in ambiente di produzione per poter essere utilizzata.

6 Creazione connessioni ODBC

Per poter avere accesso ai dati da una applicazione è necessario utilizzare una Data Connection. Saranno disponibili tre data connection, tutte di tipo ODBC, per accedere ai diversi sistemi.

Le connection saranno di tre tipi:

- Teradata – BFD10A
- Hive – Z3777T
- Oracle – CDMZ0

Le connessioni saranno già configurate per accedere ai sistemi sorgenti ma, per poter attivare la connessione, saranno richieste le credenziali dello sviluppatore che accede al dato. Questa scelta è legata a regole di sicurezza necessarie per la tracciatura degli accessi.

La connction via Hive avverrà sempre via ODBC con una data connection già pronta che si collega al database sorgente utilizzando una login di sistema non modificabile dall'utente.

IMPORTANTE: *questa impostazione richiederà l'inserimento di nuove credenziali nel momento in cui alla persona che ha sviluppato l'applicazione dovessero essere, per qualsiasi motivo, tolti o cambiati i diritti di accesso*

6.1 Gestione cartelle generiche per utente

Per gli utenti che utilizzano la piattaforma sarà necessario predisporre delle cartelle personali all'interno delle quali saranno salvati i file di dati che l'utente stesso utilizza in modalità self service. Tali cartelle saranno generate sulla NAS di riferimento da DSI e saranno ad accesso esclusivo di ogni singolo utente.

La cartella sarà denominata con lo userid dell'utente e avrà uno spazio di storage limitato, uguale per ogni utente.

Le necessità di incremento di spazio dovranno essere richieste dall'utente a DSI.

IN SOSPESO FINO ALLA DEFINIZIONE DI UN EVENTUALE ACCESSO VIA SHAREPOINT

7 Mappatura esigenza di reporting e architettura consigliata

Riportiamo una matrice che collega l'esigenza di reporting con la tipologia di architettura consigliata per indirizzarla.

	Esigenza reporting	Architettura consigliata
Report centralizzato	Report «business» per aree organizzative che non operano in self-service	3 livelli
Report urgente end-to-end	Report direttamente sviluppati dalla struttura di coordinamento centrale DOF, a fronte di esigenze che rivestono carattere di urgenza e per cui è richiesta elevata flessibilità e tempestività di azione	3 livelli
Flusso tracciato / automatizzato	Estrazione dati e fornitura di un tracciato, senza esigenza di presentazione dati statica o dinamica	Diretta
Self service su set info già predisposto	Report business (e.g., operativo, commerciale) che possono essere sviluppati o fruiti in data discovery in modalità self-service, su set di dati già disponibile per il Report Owner	Diretta
Self service su set info da predisporre centralmente	Report business (e.g., operativo, commerciale) che possono essere sviluppati o fruiti in data discovery in modalità self-service, su set di dati da predisporre centralmente	Diretta
Self service su set info predisposto da utente	Report business (e.g., operativo, commerciale) che possono essere sviluppati o fruiti in data discovery in modalità self-service, su set di dati predisposti dal report owner (avanzato)	Diretta

8 Coni di visibilità dei dati

Per garantire i giusti livelli di visibilità del dato, oltre alle regole implementate direttamente nei sistemi sorgenti, è necessario inserire il controllo della Section Access nello script di caricamento dell'applicazione utilizzata dagli utenti.

È possibile utilizzare la sezione relativa al controllo degli accessi nello script di caricamento dei dati per gestire la sicurezza. In questo modo, è possibile utilizzare un singolo file per conservare i dati di un certo numero di utenti singoli o di gruppi di utenti. Qlik Sense utilizza le informazioni della sezione relativa al controllo degli accessi per le operazioni di autenticazione e autorizzazione; in questo modo, i dati vengono ridotti in modo dinamico e vengono visualizzati solo i dati pertinenti agli utenti.

Le opzioni di sicurezza sono incorporate nel file stesso, pertanto anche un file scaricato può ritenersi in qualche modo protetto. Se, tuttavia, i requisiti di sicurezza sono elevati, è consigliabile impedire il download di file e il relativo utilizzo offline e i file dovrebbero essere pubblicati solo dal server di Qlik Sense. Poiché tutti i dati sono memorizzati in un file, è possibile che tale file sia di dimensioni molto grandi.

Il controllo degli accessi viene gestito mediante una o più tabelle di sicurezza caricate secondo la normale procedura di caricamento dei dati utilizzata da Qlik Sense. Ciò consente di memorizzare queste tabelle in un database standard. Le istruzioni dello script che gestiscono le tabelle di sicurezza sono incluse nella sezione relativa al controllo degli accessi, riconoscibile nello script perché viene inizializzata dall'istruzione Section Access.

Se nello script è definita una sezione relativa al controllo degli accessi, la parte dello script destinata al caricamento dei dati dell'app deve fare parte di una sezione differente, inizializzata dall'istruzione Section Application.

Esempio:

```
Section Access;
```

```
LOAD * inline
```

```
[ACCESS,USERID
```

```
USER,U ];
```

```
Section Application;
```

```
LOAD... .. from... ..
```

8.1 Campi di sistema della sezione relativa al controllo degli accessi

I livelli di accesso vengono assegnati agli utenti tramite una o più tabelle caricate all'interno della sezione relativa al controllo degli accessi. Queste tabelle possono contenere diversi campi del sistema specifici dell'utente, in genere USERID e il campo impiegato per definire il livello di accesso ACCESS. Tutti i campi del sistema della sezione relativa al controllo degli accessi vengono utilizzati nelle operazioni di autenticazione o autorizzazione. Di seguito è descritto il gruppo completo dei campi del sistema della sezione relativa al controllo degli accessi.

ACCESS Definisce il tipo di accesso che l'utente corrispondente dovrà avere.

L'accesso alle app di Qlik Sense può essere autorizzato per utenti o gruppi di utenti specifici. Nella tabella di sicurezza, gli utenti possono essere assegnati ai livelli di accesso ADMIN o USER. Se non viene assegnato alcun livello di accesso valido, all'utente non è consentito aprire l'app.

Un persona con privilegi ADMIN dispone dell'accesso a tutti i dati dell'app. Una persona con privilegi USER può esclusivamente accedere ai dati, così come definito nella tabella di sicurezza.

USERID Contiene una stringa corrispondente al nome utente di Qlik Sense. Qlik Sense recupera le informazioni di accesso dal proxy e le confronta con il valore incluso in questo campo.

GROUP Contiene una stringa corrispondente a un gruppo in Qlik Sense. Qlik Sense risolverà l'utente specificato dal proxy confrontandolo con questo gruppo.

Quando si utilizzano gruppi per ridurre i dati, l'utente dell'account INTERNAL\SA_SCHEDULER dovrà comunque abilitare il ricaricamento dello script in un'attività di Qlik Management Console.

OMIT Contiene il nome del campo che deve essere omissso per questo utente specifico. È possibile utilizzare caratteri speciali e il campo può essere vuoto. Un modo semplice per agire in questo senso consiste nell'utilizzare un campo secondario.

Qlik Sense confronterà l'utente fornito dal proxy con UserID e lo risolverà l'utente relativamente ai gruppi della tabella. Se l'utente appartiene a un gruppo a cui è consentito l'accesso oppure l'utente corrisponde, questi otterrà l'accesso all'app.

IMPORTANTE: *se non è possibile accedere a un'app dopo aver impostato la sezione relativa al controllo degli accessi, sarà comunque possibile aprirla senza dati e modificare la sezione relativa al controllo degli accessi nello script di caricamento dei dati. L'utente deve disporre dell'accesso per la modifica e il ricaricamento dello script di caricamento dei dati.*

8.2 Riduzione dinamica dei dati

Qlik Sense supporta la riduzione dinamica dei dati, che consente di nascondere all'utente alcuni dati in un'app in base all'autenticazione eseguita nella sezione relativa al controllo degli accessi:

- I campi (colonne) possono essere nascosti utilizzando il campo di sistema OMIT.
- I record (righe) possono essere nascosti collegando i dati della sezione relativa al controllo degli accessi ai dati reali: la selezione di valori da mostrare o escludere viene controllata tramite uno o più campi aventi nomi comuni nella sezione relativa al controllo degli accessi e nell'a sezione dell'applicazione. Dopo l'autenticazione dell'utente, Qlik Sense esegue un tentativo di associare le selezioni effettuate nei campi nella sezione relativa al controllo degli accessi a tutti i campi nella sezione dell'applicazione, rispettando esattamente gli stessi nomi di campo (i nomi di campo devono essere scritti in MAIUSCOLO). Dopo avere effettuato le selezioni, Qlik Sense nasconde definitivamente all'utente tutti i dati esclusi da queste selezioni.

IMPORTANTE: *Tutti i nomi di campo utilizzati nel trasferimento descritto sopra e tutti i valori di campo in questi campi devono essere in caratteri maiuscoli in quanto, per impostazione predefinita, tutti i nomi di campo e i valori di campo vengono convertiti in caratteri maiuscoli nella sezione relativa al controllo degli accessi.*

Esempio: Riduzione dei dati basata su ID utente

```
section access;
```

```
LOAD * inline [
```

```
ACCESS, USERID,REDUCTION, OMIT
```

```
USER, AD_DOMAIN\ADMIN,*,
```

```
USER, AD_DOMAIN\A,1,
```

```
USER, AD_DOMAIN\B, 2,NUM
```

```
USER, AD_DOMAIN\C, 3, ALPHA
```

```
ADMIN, INTERNAL\SA_SCHEDULER,*,
```

```
];
```

```
section application;
```

```
T1:
```

```
LOAD *,
```

```
NUM AS REDUCTION;
```

LOAD

Chr(RecNo()+ord('A')-1) AS ALPHA,

RecNo() AS NUM

AUTOGENERATE 3;

Il campo REDUCTION (maiuscolo) è incluso sia nella sezione relativa al controllo degli accessi che nell'applicazione della sezione (tutti i valori di campo sono in maiuscolo). In genere i due campi sono completamente diversi e separati; tuttavia, utilizzando la sezione relativa al controllo degli accessi, questi campi vengono collegati e il numero dei record visualizzati dall'utente viene ridotto.

Il campo OMIT nella sezione relativa al controllo degli accessi definisce i campi da nascondere all'utente.

Il risultato sarà:

- L'utente ADMIN può visualizzare tutti i campi e solo i record che possono vedere gli altri utenti in questo esempio quando REDUCTION è 1, 2 o 3.
- L'utente A può visualizzare tutti i campi, ma solo i record associati a REDUCTION=1.
- L'utente B può visualizzare tutti i campi eccetto NUM e solo i record associati a REDUCTION=2.
- L'utente C può visualizzare tutti i campi eccetto ALPHA e solo i record associati a REDUCTION=3.

Esempio: Riduzione dei dati basata su gruppi di utenti

section access;

```
LOAD * inline [  
  
ACCESS, USERID, GROUP, REDUCTION, OMIT  
  
USER, *, AD_DOMAIN\ADMIN, *,  
  
USER, *, AD_DOMAIN\A, 1,  
  
USER, *, AD_DOMAIN\B, 2, NUM  
  
USER, *, AD_DOMAIN\C, 3, ALPHA  
  
USER, *, GROUP1, 3,  
  
ADMIN, INTERNAL\SA_SCHEDULER, *, *,
```

```
];  
  
section application;  
  
T1:  
  
LOAD *,  
  
NUM AS REDUCTION;  
  
LOAD  
  
Chr( RecNo()+ord('A')-1) AS ALPHA,  
  
RecNo() AS NUM  
  
AUTOGENERATE 3;
```

Il risultato sarà:

- Gli utenti che appartengono al gruppo ADMIN possono visualizzare tutti i dati e tutti i campi.
- Gli utenti che appartengono al gruppo A possono visualizzare i dati associati a REDUCTION=1 in tutti i campi.
- Gli utenti che appartengono al gruppo B possono visualizzare i dati associati a REDUCTION=2, ma non nel campo NUM.
- Gli utenti che appartengono al gruppo C possono visualizzare i dati associati a REDUCTION=3, ma non nel campo ALPHA.
- Gli utenti che appartengono al gruppo GROUP1 possono visualizzare i dati associati a REDUCTION=3 in tutti i campi.
- L'utente INTERNAL\SA_SCHEDULER non appartiene ad alcun gruppo ma può visualizzare tutti i dati in tutti i campi.

Il carattere speciale * in questa riga si riferisce soltanto a tutti i valori all'interno della tabella Section Access. Se in Section Application vi sono valori non disponibili nel campo REDUCTION in Section Access, verranno ridotti.

9 Allegati

Di seguito si trovano gli allegati di:

- ETL01 Template
 - È la base da utilizzare per la fase di ETL01 in combinazione con il file Excel di configurazione presente nella cartella Config all'interno del Project_Template.zip.
 - Per accedere alla cartella suddetta e per salvare i file QVD sarà necessario definire una Data Connection che punti alla relativa struttura di progetto
- ETL02 Template
 - È la base da utilizzare per lo sviluppo di ETL02. Questo può essere il file di QlikMart.
- Project Directory Template
 - Contiene la struttura di progetto e il file Excel di configurazione di ETL01
- Data Visualization Guidelines (inglese)
 - Documento che contiene le Best Practices di Data Visualization



ETL01_Template.qvf



ETL02_Template.qvf



Project_Template.zip



User Experience Best
Practices for Data An