

**<Customer Name>**

**<Project Name>**

**Architecture Document**

**<Version No.>**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Prepared By / Last Updated By** | **Reviewed By** | **Approved By** |
| **Name** | <Author> | <Reviewer> | <Approver> |
| **Role** | <Designation / Role> | <Designation / Role> | <Designation / Role> |
| **Signature** | <Signature / Initials> | <Signature / Initials> | <Signature / Initials> |
| **Date** | <DD-Mmm-YYYY> | <DD-Mmm-YYYY> | <DD-Mmm-YYYY> |

**Table of Contents**

[1.0 <Heading 1> 3](#_Toc522036293)

[1.1 <Heading 2> 3](#_Toc522036294)

[1.1.1 <Heading 3> 3](#_Toc522036295)

[2.0 Change Log 3](#_Toc522036296)

# Introduction

## Purpose of this Document

The solution architecture of project ICE is a description of the components and services of the warehouse and how they are placed together. Based on the given descriptions a skilled professional should be able to implement the solution. The solution architecture provides a framework to achieve enterprise integration in order to support the business. It also allows data sharing between various lines of businesses, allow faster development, reusability, data consistency. The most important aspect of the architecture is that it is an evolutionary process.

The Solution Architecture describes the underlying foundation or basic framework for the data warehouse and business intelligence, as well as the application components that are necessary to implement and deploy the application.

This document does not explain how specific requirements are implemented nor does it capture the technical design of the system; this information can be found in the individual Detail Design documents.

## Audience

* Cognizant Project, Program management team
* Cognizant development team
* Client IT and Business Team
* Project Key stake holders

# Introduction to the project

## Purpose of the project

ICE Project has been envisioned by the BICC team to address the opportunities of improvements identified with Summit and to provide one Scalable, robust Framework for all future BI development in RB.

The purpose of ICE project is to provide a scalable, flexible, scalable framework which can cater/adapt to the fast changing business needs. Speed of response is a critical factor to the success. A new enterprise data warehouse which acts as a single source of data for enterprise BI reporting will ensure consistent data availability.

The ICE project will

* Deliver a scalable framework that meets performance expectations and is a single source for following business needs:
* Regional and Global KPI scorecard (Based on the defined KPI framework)
* Global Key Customer and Product profitability analysis
* Regional Customer and Product profitability analysis
* Develop and implement key analytical areas for Market Share, Storm analysis, Balance sheet analysis and Gross Margin Variance Analysis on ICE infrastructure
* Develop and implement additional KPIs not currently delivered – Variance analysis, Market share, Sales (STORM, ROI, Efficiency)
* Develop and implement external interfaces from ICE to Global scorecard, Global Pricing, NPD Cube and Siebel reporting on ICE infrastructure

Global Data warehouse that is agnostic of source transactional system and contains data from following data sources:

* General Ledger (GL) transactional data from regional JDE instances
* Sales transactional data from regional JDE instances
* Financial plan and forecast scenarios from JDE instances, Hyperion Enterprise and flat files
* Required master data: customer, product, organizational hierarchies, account hierarchies etc.
* Global Nielsen data
* Global data standards and Local hierarchies from the Hyperion DMR system.

## Scope

* The ICE project will leverage an independent ETL infrastructure that will optimize for the enterprise rather than for a specific project
* Design for simplicity, flexibility and performance as opposed to cost e.g. higher cost for additional storage requirements for landing and landing history files that will speed up process execution.
* Handle data transformations and computations upstream in the ETL layer so that the data delivery (reporting) layer complexities are reduced aiding performance of report queries.
* Design with reusability as primary goal e.g. reuse of same jobs to pull data from multiple JDE instances. Also repeating design components to be modularized (e.g. reusable jobs) for reuse and ease of change and maintenance.
* All jobs must use the parallel execution strategy of the integration tool.
* Abiding by all Reckitt Benckiser standards (hardware, software, security, architecture, etc) or file appropriate exceptions as necessary.

## Business Requirement

Current Application provides the ability to consolidate and report on the core business KPI’s across geographical and business hierarchies in RB. It also allows information to be sliced and diced by various dimensions and be monitored at different levels in the dimension hierarchies.

However, the current application does not deliver non-financial KPI’s (Market Share, Sales, Supply, Variance Analysis) and has been identified as an opportunity for improvement.

## System Requirements

This section provides a brief overview of the individual components that together constitute the conceptual architecture.

### Source Systems

The landing area of the enterprise data warehouse will receive data from the following sources. For more information on source systems please refer to the ICE Source System Analysis document.

#### JDE

The JDE system is the enterprise planning system which captures the transactional data. The JDE instances that are co-located in UK and each instance are for a specific region provides information for various business operations such as for GL, Sales Order Processing, Reference data, Planning and Forecasting etc. In this phase only general ledger data, sales, planning and forecasting and master reference data will be brought into the landing for processing into the warehouse.

Information in this system is stored in DB2/400 database in library file formats.

#### AC Nielsen

AC Nielsen a market research intelligence company that captures market share of various CPG companies will provide market share data for Reckitt Benckiser & its competitor’s. This market share information will be sent by the external company (AC Nielsen) to the datastage server in the form of flat files. The market share data is available bi-monthly with a lag of 2 months.

#### Global Data Standards (GDS)

RB has defined data standards that are applicable to its business processes worldwide. These standards are called Global Data Standards which dictates the rules and structures of the information to support regional and global consolidation. These data standards are developed in spreadsheet format and later implemented in the enterprise planning JD Edwards system. RB has also taken an initiative to move the global data standards to the new Hyperion DRM system. In phase 1 only identified global data standards will be pulled from the Hyperion DRM system into the data warehouse.

#### Local Hierarchies

Local business maintains specific/alternate hierarchies for product, customer, brand, legal entities and cost center in a spreadsheet. This information will be available in the Hyperion DRM system and the landing process will pull data from the Hyperion DRM database.

#### Planning & Forecasting

Planning and forecasting data is available in JDE instances, in Hyperion enterprise management system and in spreadsheets. Planning and forecasting are done at a global, regional and country level down to brand and segment hence related to products. Country level plan and forecasting are referred as local plan and forecast which defines the target, vision and latest estimate sales and volume figures for both products and customer. Planning is done at a monthly level around the last quarter of the current calendar year for the next year. Forecasting is done at a montly level around the preceding month of every quarter. There are times when forecasting are restated for the same period and these restatements are known as versions.

#### Exchange Rate

Currency exchange rate information to convert actuals and plan and forecasting figures from the local currency to the identified currencies will be made available to the warehouse in a flat file. These files will be uploaded to the landing database via the web application

### Landing Area

Landing area is divided into two areas.

Landing – Data from multiple source systems will be brought into this in a format that will heavily mimic the source system data structures. This will be a non-persistent layer and data in this layer will remain for 1 day or fully processed.

Landing History – Landing history area will be a replica of the landing area. After processing source system data from the landing into the next layer all the data from landing area will be moved as is to the landing history area. This layer will be a persistent layer and data that is older than 7 years will be archived into an archive database.

### Enterprise Data Warehouse

The enterprise data warehouse serve as the central repository for integrated data and provide a single version of truth. Data in this layer will depict the business model and contain data at an atomic level of the source systems. Data from all sources will be brought into this layer and integrated with each other into a third normal form. Data to this layer will be loaded everyday. The EDW will be the single source of data to downstream systems.

### Aggregated Data Marts

In phase 1 eight aggregated data marts have been identified. These data marts will contain information at a monthly granularity. These data marts will be created in their respective schemas and conformed dimension and local dimensions will be created in another schema. Role based and Row level security model for logged in user will also be implemented at this level.

### Cognos Framework Manager (Semantic Layer)

Cognos framework modeling will connect to the aggregated data marts in order to do the business modeling. The model may contain the following layers as per the phase 1 requirements

Foundation Layer - import View of data marts such as Global customer profitability, Regional customer profitability, Balance Sheet Analysis, Market Share Analysis, KPI Scorecard, etc.

Modeling layer – This layer will contain the semantic view of the business.

Presentation Layer – This layer will be modeled in a star schema format and exposed to end user via different Cognos reporting tools such as Report Studio, Query Studio, etc.

Analytical Layer

Subject area packages

### Cognos Connection Portal

The Cognos connection portal will be the presentation layer where dashboards and reports will be rendered for allowed user groups.

## Architectural Requirements / Quality Constraints

### Performance

|  |  |
| --- | --- |
| Optimal performance for end user querying and reporting | 1. Since the star schemas will be deployed in ROLAP architecture a ROLAP performance assessment will be carried out. 2. Comparisons will be drawn between the pre-defined performance requirements and ROLAP assessment facts. If ROLAP assessment facts is considerably lower than architecture strategy will be revisited in order to conceive MOLAP approach. 3. If ROLAP performance assessment facts are within a permissible range than performance test will be carried out and based on the results performance tuning will be done for the troubled areas. |

### Scalability

|  |  |
| --- | --- |
| Minimal dependency on source systems | 1. All available data elements sourced into DW. 2. Data at the lowest grain sourced into DW. 3. Persistent landing history area to provide flexibility for re-processing. |
| Minimal or no dependency on source system changes | 1. Solution agnostic will be achieved in the ETL layer. Every job that will pull data from source systems and push to data warehouse will first push the data through the pipeline to an intermediate data structure that will closely resemble the data warehouse structure. |
| Need to support analysis across subject areas as well as nuances specific to regions | 1. Dimensional logical star schemas sharing conformed dimensions. 2. Flexibility to utilize local hierarchies |

### Reliability / Fault Tolerance

1. A permissive design approach will be adopted for the ICE project towards error handling. This method will provide a balance between missing information and incorrect information.
2. Records that cannot be processed will be treated as “hard rejects”. Hard rejects will constitute files and/or records that do not satisfy the file and field level checks & validations.
3. Such files and/or records will not be loaded into the Landing Area. Hard rejects can originate from a few steps in the overall process flow:
4. File level validations (syntactic checks)
5. Field level validations (syntactic checks)
6. For JDE source data that fail the file/field level validations, a notification will be sent to the identified team to resolve the issue and resubmit the corrected information for processing. This notification functionality will be coded as a part of the ETL infrastructure that is used to create the file validation program.
7. For source extract files that fail file level validations, the files will be routed to the Datastage BAD files directory. A notification report will be created and sent to the identified IS personnel to look into the issue and fix it before uploading the file into the source extract area.
8. Field level validations will be performed as part of the ETL process. With the ability in Datastage to write to heterogeneous targets, the bad records detected in a job can be routed to a flat file or table where they can be accessed by the source team in order to analyze and understand the underlying issues.
9. The files and records rejected will be assigned a “reject code” and a “reject description” that will facilitate easier understanding and resolution of the issue.
10. No changes or corrections will be made directly in the Data Warehousing environment. This will ensure that the Data Warehouse and the source systems are in sync.
11. For file level validation failures, there are two options that need to be considered:
12. File having issues is rejected and the other good files from the same / different systems are processed. This can potentially result in a partial load that can impact summaries in the Data Mart, OR
13. If one/more files related to one data domain have issues, the entire data load process is stopped so that the Data Warehouse is not “incomplete”. For example if we are going to receive multiple files from AC Nielsen for market share, one for category tab, one for product tab, one for market tab and one for the market share. Lets assume the category tab has incomplete information and it will be good to terminate the load process for market share information.
14. If there are issues with reference data files (e.g. customer, product, etc), it is recommended that the entire data load process be stopped. If there are issues with some library files from core product systems (say JDE), the other good files can be processed. However adequate communication needs to be sent to the users indicating that base data and summaries in the DW and Data Mart will be incomplete.
15. The decisions around processing of partial source files / records will be finalized as a part of the Detailed ETL Design phase once the entire source to target mappings between the source files and the target tables have been defined.
16. Regardless of whether an error requires manual inspection, correction of data and a rerun of the process, the Data Stewards and related Technical personnel need to know if any files and/or rows were rejected during the load, especially if a response is critical to the continuation of the process. Therefore, it is critical to have a notification process in place.
17. While processing data the ETL jobs should be able to log ETL load numbers such as records read, processed, loaded, rejected etc into a set of audit tables. Information logged in this table can be used reporting/auditing purposes.
18. The notification will be handled using the post-session e-mail functionality in Datastage to trigger the delivery of an e-mail. Post-session scripts can be written to increase the functionality of the notification process to send detailed messages upon receipt of an error.
19. In case there is a datastage job failure email can be generated and sent to identified person at a session level. Once the source system IS personnel or the identified person have been notified of the error, they will need to conduct a root cause analysis the file/records to undersFollowing table describes some high level events and associated notifications. The complete notification and alert processing can be designed during the ETL Detail Design phase.

|  |  |  |  |
| --- | --- | --- | --- |
| Event/Condition | Action | Notification Recipients | Alert Recipients |
| Successful completion of a daily job flow | Send an email daily with job flow statistics | ETL Admin/Data Steward |  |
| Successful completion of a monthly job flow | Send an email monthly with job flow statistics | ETL Admin/Data Steward |  |
| Failure of a daily job | Send an email daily with failure details | ETL Admin/Data Steward | DBA/ETL Support |
| Failure of a monthly job | Send an email daily with failure details | ETL Admin/Data Steward | DBA/ETL Support |

## Modular Components

1. Design with reusability as primary goal e.g. reuse of same jobs to pull data from multiple JDE instances. Also repeating design components to be modularized (e.g. reusable jobs) for reuse and ease of change and maintenance.

### Evolution

* Deliver a scalable framework that meets performance expectations and is a single source for following business needs:
* Regional and Global KPI scorecard (Based on the defined KPI framework)
* Global Key Customer and Product profitability analysis
* Regional Customer and Product profitability analysis
* Develop and implement key analytical areas for Market Share, Storm analysis, Balance sheet analysis and Gross Margin Variance Analysis on ICE infrastructure
* Develop and implement additional KPIs not currently delivered – Variance analysis, Market share, Sales (STORM, ROI, Efficiency)
* Develop and implement external interfaces from ICE to Global scorecard, Global Pricing, NPD Cube and Siebel reporting on ICE infrastructure
* Global Data warehouse that is agnostic of source transactional system and contains data from following data sources:
* General Ledger (GL) transactional data from regional JDE instances
* Sales transactional data from regional JDE instances
* Financial plan and forecast scenarios from JDE instances, Hyperion Enterprise and flat files
* Required master data: customer, product, organizational hierarchies, account hierarchies etc.
* Global Nielsen data
* Global data standards and Local hierarchies from the Hyperion DMR system.

### Layer Decoupling

### Security

Users logging into the ICE system will be authenticated through Cognos infrastructure by connecting to the corporate LDAP.

The data authorization will be implemented by using a role level and row level security model in order to restrict data access for unauthorized user. The defined security model will be deployed within the analytical areas. A web interface will be developed to administer the data authorization security framework.

## User Groups, Roles and Geography Matrix

The following are the high level user groups identified for ICE project by Functional area. The user groups are defined as a combination of functional areas and geographic dimension.

### Role - Management:

* EC & Upper Mgmt
* Regional Mgmt Team
* Local Mgmt Team

### Role - Finance:

* Global Finance
* Regional Finance
* Local Finance

### Role - Sales:

* Global Sales
* Area Sales
* Regional Sales
* Local Sales

### Role - Marketing:

* Global Category
* Area Marketing
* Regional Marketing
* Local Marketing

### Role - HR:

* Area HR
* Regional HR
* Local HR

### Role - IT:

* Area IT
* Regional IT
* Local IT
* CHQ IT
* BICC IT

### Role - Regulatory:

* Area Regulatory
* Regional Regulatory
* Local Regulatory
* There are three geographical areas which will be part of the security framework.
* Area
* Regional
* Local/Country

### Interoperability

*NA*

### Configurable

NA

### Concurrency

NA

### Legacy Reuse & Integration

*NA*

### Error handling and system degradation

Covered in section 2.5.3 on Reliability and Fault Tolerance

### Availability

This new enterprise data warehouse which acts as a single source of data for enterprise BI reporting will ensure consistent data availability.

### Stability

Covered in section 2.5.3 on Reliability and Fault Tolerance

# Data warehouse Architecture

## Technical Architecture Diagram



## Description

The central theme of the architecture is to de-fragment the BI environment in order to provide manageability. To that extent a number of components has been centralized and integrated. The following factors are taken into consideration for the solution architecture.

1. Support scalability at both functional and workload level.
2. Support flexibility by providing easy integration with upstream and downstream systems
3. Support maintainability by lowering overall effort and providing better change management
4. Support usability by providing better performance, appealing visualization and guided analytics.
5. Support security at both report objects and row level.

## External Interface View

### Actors

Covered on section 2.5.7 on Security.

### Primary Scenarios

***Covered as in section 3.1. as part of Technical architecture diagram***

# Components of the Architecture

The architecture for a datawarehouse could be divided into 4 conceptual layers viz.,

* Data Acquisition layer (ETCL Processes)
* Data Layer (Databases for the staging area, the datawarehouse, and the multi-dimensional structure)
* Data Access Layer (Front End - OLAP, Mining etc)
* Metadata Layer

The following four sections (sec 2.2.1 to 2.2.4) shall talk about each layer in detail.

## Data Acquisition Layer

### Landing (Layer 1)

This layer is defined as the landing area and divided into the landing and landing history. Once the data in the landing area is processed the data will be moved as is to the landing history area. The landing layer job flows must have the following abilities

Ability to extract data from heterogeneous sources such as relational data, flat files, JDE library files.

Ability to perform data validation and discard the erroneous information to the error processing section.

Ability to push data into the warehouse once processing is finished in this layer

Ability to move processed data from landing area to the landing history area.

Ability to retrieve data from landing history to landing area for repeat processing.

While extracting data from multiple instance must mark each rows with the instance information in order for the downstream job flows to identify them and process accordingly

Data extraction processes that will run in this layer to bring source data for further processing will not filter any data elements in the landing area and have everything that source systems are capturing and can provide. However there are pre defined business rules that will be applied to extract relevant information from some source systems such as extracting only finished goods from JDE item faster files, extracting only posted transactions from the JDE general ledger file.

## Data Layer

* DataBase –DB2
* ETL-DataStage
* BI & Reporting – Cognos

## Data Access Layer

Cognos framework modeling will connect to the aggregated data marts in order to do the business modeling. The model may contain the following layers as per the phase 1 requirements

1. Foundation Layer - import View of data marts such as Global customer profitability, Regional customer profitability, Balance Sheet Analysis, Market Share Analysis, KPI Scorecard, etc.
2. Modeling layer – This layer will contain the semantic view of the business.
3. Presentation Layer – This layer will be modeled in a star schema format and exposed to end user via different Cognos reporting tools such as Report Studio, Query Studio, etc.
4. Analytical Layer
5. Subject area packages

### Cognos Connection Portal

The Cognos connection portal will be the presentation layer where dashboards and reports will be rendered for allowed user groups.

## Metadata Layer

All technical metadata related to source and target tables, ETL jobs, job flows will be stored in the datastage repository. The inbuilt functionality of the datastage tool will be used to access the technical metadata.

# Primary Quality Constraints and Quality Analysis

## Strategies

|  |  |  |
| --- | --- | --- |
| # | Requirement | Architecture Impact |
| 1 | Minimal dependency on source systems | 1. All available data elements sourced into DW. 2. Data at the lowest grain sourced into DW. 3. Persistent landing history area to provide flexibility for re-processing. |
| 2 | Minimal or no dependency on source system changes | Solution agnostic will be achieved in the ETL layer. Every job that will pull data from source systems and push to data warehouse will first push the data through the pipeline to an intermediate data structure that will closely resemble the data warehouse structure. |
| 3 | Need to support analysis across subject areas as well as nuances specific to regions | 1. Dimensional logical star schemas sharing conformed dimensions. 2. Flexibility to utilize local hierarchies |
| 4 | Optimal performance for end user querying and reporting | 1. Since the star schemas will be deployed in ROLAP architecture a ROLAP performance assessment will be carried out. 2. Comparisons will be drawn between the pre-defined performance requirements and ROLAP assessment facts. If ROLAP assessment facts is considerably lower than architecture strategy will be revisited in order to conceive MOLAP approach. 3. If ROLAP performance assessment facts are within a permissible range than performance test will be carried out and based on the results performance tuning will be done for the troubled areas. |

## Trade Offs

Performance considerations should be evaluated against considerations for Manageability, Scalability and Security.

# Architecture – Layered View

Covered in section 3.1 as part of Technical Architecture.

## Layer 1 to 2

### Responsibility

### Landing (Layer 1)

This layer is defined as the landing area and divided into the landing and landing history. Once the data in the landing area is processed the data will be moved as is to the landing history area. The landing layer job flows must have the following abilities

1. Ability to extract data from heterogeneous sources such as relational data, flat files, JDE library files.
2. Ability to perform data validation and discard the erroneous information to the error processing section.
3. Ability to push data into the warehouse once processing is finished in this layer
4. Ability to move processed data from landing area to the landing history area.
5. Ability to retrieve data from landing history to landing area for repeat processing.
6. While extracting data from multiple instance must mark each rows with the instance information in order for the downstream job flows to identify them and process accordingly

Data extraction processes that will run in this layer to bring source data for further processing will not filter any data elements in the landing area and have everything that source systems are capturing and can provide. However there are pre defined business rules that will be applied to extract relevant information from some source systems such as extracting only finished goods from JDE item faster files, extracting only posted transactions from the JDE general ledger file.

#### Information Detection

Source information detection will be required for the following scenarios

1. Detection of AC Nielsen Market share files in the landing area in order to process them into the data repository
2. Detection of Hyperion planning and forecasting file and local planning and forecasting flat files upload to landing area via web application
3. Detection of exchange rate file uploaded to landing area via web application.
4. On demand load of account balance information from JDE instances.
5. Data transform and load from landing to warehouse to data marts will be controlled through proper job flow and scheduling.

The detection process will serve as a trigger for initiating the respective ETL processes. The triggering event will be event based rather than time based. All other information will be scheduled to run on a timely fashion

#### Global Data Standards

RB has defined data standards that are applicable to its business processes worldwide. These standards are called Global Data Standards which dictates the rules and structures of the information to support regional and global consolidation. These data standards are developed in spreadsheet format and later used in the JD Edwards enterprise planning system implementation.

During the writing of this document RB decided to deploy Hyperion DRM system to move the storage of the Global data standards from spreadsheet to Hyperion DRM.. Once the GDS information is available in the Hyperion DRM system it will be pushed to an intermediate database. The ETL layer will then pull the data from the intermediate database into the landing area. The global data standards will be loaded to landing layer only once a day.

Table 5‑1 GDS Details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Domain** | **Acquisition Strategy** | **Frequency** | **CDC** | **Technical Details** | **# of tables/files** |
| Data Standards | Hyperion DRM will push data to an intermediate database in a specified format. Extraction process will pull daily from the intermediate database | Full refresh – Daily | Compare current set of data with the previous set of data and push only delta to downstream | N/A | 14 |

#### JDE

Data for GL, Sales Order Processing, Reference data, Planning and Forecasting will be sourced from the enterprise planning system JD Edwards. In phase 1 only 3 countries are selected as part of the rollout process.

* There are multiple instances of JDE from where data will be extracted. These information will be captured in the parameter files/tables.
* Each instance will have a time window and data will be pulled from the JDE instances within the pre-defined time window
* The triggering process will trigger the jobs within the defined time window in the instance parameterization ETL component to pull data from the respective JDE instances
* Full reference data will be extracted from the JDE instances into the landing area.
* Before loading actuals the process must ensure the availability of currency exchange rates.
* Transaction data for general ledger and sales will be extracted on a daily basis to the landing area. In order to capture incremental transaction data the extraction process will look into the followings
* Check for date and time stamp
* For general ledger pull only those rows where post flag = ‘Y’
* Account balances will be extracted on a monthly basis or on demand to the landing area once the books are closed. The web interface will provide a feature to an instance user to indicate the book closure event and subsequently start the account balance extraction process.
* Planning and forecasting data for a product business unit will be extracted from JDE into the landing area for further processing into the data repository.

Table 5‑2 JDE Details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Domain** | **Acquisition Strategy** | **Frequency** | **CDC** | **Technical Details** | **# of tables/files** |
| GL Data | Pull | Incremental refresh – Daily | Last update date > today – # of days defeined in the parameter file and post\_flag = ‘Y’ | DB2/400 system  RBGDC0P5  RBGDC0P4  RBGDC0P1 | 1 |
| Account Balance | Pull | Incremental refresh – Monthly,  On Demand | Previous month data | DB2/400 system  RBGDC0P5  RBGDC0P4  RBGDC0P1 | 1 |
| Sales | Pull | Incremental refresh – Daily | Last update date > today – # of days defeined in the parameter file | DB2/400 system  RBGDC0P5  RBGDC0P4  RBGDC0P1 | 4 |
| Master Data | Pull | Full refresh daily (compare with old set and push delta) | NA | DB2/400 system  RBGDC0P5  RBGDC0P4  RBGDC0P1 | 6 |
| Planning | Pull | Full refresh – yearly | NA | DB2/400 system  RBGDC0P5  RBGDC0P4  RBGDC0P1 | 1 |
| Forecasting | Pull | Full refresh – quarterly or as soon as a version is available in the JDE instance | NA | DB2/400 system  RBGDC0P5  RBGDC0P4  RBGDC0P1 | 1 |

#### AC Nielsen (Sales & Volume data)

Market share data provided by an external entity called AC Nielsen will be pushed by the external company to a designated area in the datastage server on a bi-monthly basis with a lag of 2 months. This designated area can be used as a drop off area by the external processes outside of the ICE infrastructure Once this file is available in the common area the ETL extraction process will extract this data and load into the landing area for further processing.

Table 5‑3 AC Nielsen Details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Domain** | **Acquisition Strategy** | **Frequency** | **CDC** | **Technical Details** | **# of tables/files** |
| Market Share | Pull | Full refresh – Bi-Monthly | NA | Flat File | 1 |

#### Planning & Forecasting (Hyperion)

Regional users will use a web application to upload the planning & forecasting data into the landing area. Before loading planning and forecasting data the process must ensure the availability of currency exchange rates. Once the data is available in the landing area the ETL jobs should push the data to the presentation layer provided all the pre-requisite checks are completed and satisfied.

Table 5‑4 Planning & Forecasting Details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data Domain | Acquisition Strategy | Frequency | CDC | Technical Details | # of tables/files |
| Planning | Pushed by user via a web application to the landing layer and subsequently to the presentation layer via the data repository. | Full refresh – Yearly | NA |  | 1 |
| Forecasting | Pushed by user via a web application to the landing layer and subsequently to the presentation layer via the data repository | Full refresh – Quarterly | NA |  | 1 |

#### Local Hierarchies

Local hierarchies are defined in the Hyperion DRM system. Hyperion DRM system will push the data to an intermediate database. The ETL layer will then pull the data from the intermediate database into the landing area. After creating the hierarchy in the Hyperion DRM system users of that system might want to see the local hierarchy in ICE immediately and in such a case DRM system must provide the update to the intermediate database. In such a scenario an ETL scheduler can be created which can call the information detection program for local hierarchy in order to extract the local hierarchies into the data repository.

Table 5‑5 Local Hierarchies Details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data Domain | Acquisition Strategy | Frequency | CDC | Technical Details | # of tables/files |
| Local/alternate hierarchies of a global dimension | Hyperion DRM will push data to an intermediate database in a specified format. Extraction process will pull daily from the intermediate database | Full refresh – Daily  On Demand | Compare current set of data with the previous set of data and push only delta to downstream | N/A | 5 |

#### Exchange Rates

Currency exchange rates will be uploaded to the landing area via a web application on a monthly basis. There will be two types of currency exchange rates that will be uploaded to the landing area.

1. Currency exchange rates for actuals – This file will contain the monthly average exchange rates in order to convert the actuals from local currency to the identified currencies. This file must be loaded into the warehouse before the actuals are pushed from the data warehouse to data marts.
2. Currency exchange rates for plan and forecasting – This file will contain the monthly average exchange rates in order to convert the local planning and forecasting data from local currency to the identified currencies. This file must be loaded into the warehouse before the plan and forecast data is loaded.

Table 5‑6 Planning & Forecasting Details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data Domain | Acquisition Strategy | Frequency | CDC | Technical Details | # of tables /files |
| Currency exchange rates for actuals | Pushed by user via a web application to the landing layer | Full refresh – Monthly | NA |  | 1 |
| Currency exchange rates for plan and forecasting | Pushed by user via a web application to the landing layer | Full refresh – Monthly | NA |  | 1 |

### Enterprise Data Warehouse ETL (Layer 2)

Once data is arrived in the landing area and ready to be loaded into the warehouse the warehouse ETL jobs will start pushing data into the warehouse from the landing area.. The warehouse job flows must have the following abilities

* Ability to run daily as per the defined schedule.
* Ability to convert landing data into the desired format of the 3NF repository area (such as converting Julian dates in JDE systems to the Gregorian format).
* Ability to load data dictionary from multiple JDE instances and interpret JDE master and transaction data as per the dictionary rules.
* Ability to discard erroneous information and report the same to the error processing section.
* Ability to transform and integrate data into a 3rd normal form schema.
* Must load master data before loading transactional data.
* While loading master data must compare the new set of data with the old set of data in order to load only the delta.
* Identify data that are ready to be processed into the warehouse and extract, transform and load them into the data warehouse
* While loading to the warehouse the jobs must create surrogate keys for each record for the master data sets.

### Benefits

Layer 1

* Jobs defined in this layer will have minimal business rules to pull data from source systems.
* Jobs defined in this layer will require minimum transformation while loading the data into the landing area.
* Jobs defined in this layer will run as per the pre-defined frequency for the source systems.
* Before pulling data from source systems jobs will perform file and field level validations.
* Jobs define in this layer will use information detection techniques to determine data availability for loading into the landing area.. For example if an external file such as global market share file from the external sources arrive in a pre-agreed landing area jobs must be invoked in order to process them into the data repository.
* Jobs defined in this layer will call appropriate downstream workflows based on the defined dependencies in order to load data into the next layer after data is loaded into the landing area.
* Workflows in a layer should be independent of their upstream and downstream workflows and should have the ability to process data into their respective layers.
* Jobs defined in this layer will move the data into the landing history after a successful load into the enterprise data warehouse.
* A set of jobs should be defined to move data from landing history to history for a specified period.
* This layer will require error handling, reprocessing of rejected records, scheduling, alerts, notifications and monitoring.
* Data that are being extracted into the landing area must be properly marked so that downstream job flows can pick them up properly to process into the downstream systems.

**Layer 2**

* Jobs defined in this layer will apply heavy business rules to integrate the heterogeneous sources into a third normal form.
* Jobs defined in this layer will require heavy transformations while loading the data into the enterprise data warehouse.
* Jobs defined in this layer will be invoked by the appropriate jobs in layer 1.
* Job flow dependencies must be defined to avoid integrity issues.
* A job flow must be defined to load the master tables before loading the transactions.
* Jobs in this layer will generate the EDW surrogate keys for the master record before loading to the master tables.
* While loading changed master and reference data jobs will apply the slowly changing dimension strategy defined in the data model.
* Jobs must normalize source information in order to load the third normal form enterprise date warehouse.
* This layer will require error handling, reprocessing of rejected records, scheduling, alerts, notifications and monitoring.
* This layer will act as the data integration layer and provide agnostics to the data repository and data marts. If there are new source systems added to this layer then based on the requirement either the existing ETL processes will be partially modified or new ETL processes will be written.

## Technology Selection

Covered in section 4.2 as part of Data Layer description.

## Mapping Layers to Functional Components

Covered in section 3.1 as part of Technical Architecture.

# Deployment Architecture

## Scenario 1

Will be covered as part of production roll out document.

## Scenario 2

Will be covered as part of production roll out document***.***

## Scenario 3

Will be covered as part of production roll out document.

# Change Log

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version Number** | **Changes Made** | | | |
| V1.0.0 | Initial baseline created on <dd-Mmm-yyyy> by <Name of Author> | | | |
| V X.Y.Z | <Please refer the configuration control tool / change item status form if the details of changes are maintained separately. If not, the template given below needs to be followed> | | | |
| **Section No.** | **Changed By** | **Effective Date** | **Changes Effected** |
| NA | NA | NA | NA |