Project End to End Design – *detailed version*

Project: - ***Insert name of project***

Author: - ***End to End Designer for project***

Date:-

Version:-

VERSION INFORMATION

LAST UPDATED	MASTER VERSION LOCATION

CHANGE HISTORY

VERSION NO.	DATE	CHANGE DESCRIPTION	APPROVED BY

REVIEWERS

VERSION NO.	DATE	NAME	TITLE / ROLE
			Delivery Manager

APPROVALS

VERSION NO.	DATE	NAME	TITLE / ROLE
			TDA lead
			Tower Lead Back End
			Tower Lead Semantic Layer
			Tower Lead Front End
			Project Director
			SME / Business Contact
			ES IT
			AM

This document contains many tables and diagrams. This reflects the remark that the E2E design is mostly used as a reference. In that case, tables are easier to use.

Since tables and diagrams are often used in this document, it is important to use a common colour scheme in the tables and diagrams.

TABLE OF CONTENTS

1 INTRODUCTION 5

1.1 Document purpose 5

1.2 Related documents 5

1.3 Key design decisions 5

2 SOLUTION OVERVIEW 7

2.1 Architecture 7

2.2 Data Sources (DS) 7

2.2.1 Sources 7

2.2.2 Data receipt & loading 7

2.2.5 Master data & reference data 9

2.3 Source data layer (SA) 9

2.3.1 Tables 9

2.3.2 Transformation 9

2.3.3 Performance activities 10

2.3.4 Databases 10

2.4 Enterprise data layer (EDL) 10

2.4.1 Tables 10

2.4.2 Transformation 12

2.4.3 Data cleansing & data quality 12

2.4.4 Performance activities 13

2.4.5 Databases 13

2.5 Business Semantic Layer (BSL) 13

2.5.1 Views 13

2.5.2 Transformation 14

2.5.3 Performance activities 14

2.5.4 Databases 14

2.6 Reporting & analytics 14

2.6.1 Usage of the data from Semantic Layer / EDL for reporting 14

2.6.2 Reporting data structures 15

2.6.3 Data access considerations 15

2.6.4 Report Front End 15

2.7 Allocation 15

2.8 Capacity planning 16
2.8.1 Initial volumes 16
2.8.2 Incremental volumes 16

2.9 User profiles and security 16
2.9.1 Personal Users 16
2.9.2 System Accounts 16
2.9.3 Data Security for data at rest 16
2.9.4 Data Security for data in motion 17

2.10 Network requirements 17

2.11 Data retention requirements 17

2.12 Archiving & back up 17

2.13 Metadata 17

2.14 Control table 18

3 DATA MIGRATION 19

3.1 Source systems 19

3.2 Migration approach 19

4 COMPLIANCE WITH PROGRAM STANDARDS 20

1 INTRODUCTION

1.1 Document purpose

This document is intended to provide a description of the physical design of the solution for the *xxx* project. It is intended to be a living document and as the project proceeds through the system lifecycle it will be updated and information appended so that at the completion of the project a detailed description of the projects deliverables along with design decisions, capacity planning considerations and data migration will be detailed within the one document.

This document only details project specific design.

It is intended that this document will be reviewed and signed off by members of the TDA along with the design standards compliance certificate.

1.2 Related documents

Identify any related documents that should be referenced alongside this document in order to provide context or background to the contents within this document. Provide referencing documents and their version numbers to fully understand the document is based on.

As a minimum, provide reference information on the Project End to End Design high-level version. A link suffices.

1.3 Key design decisions

Detail any key design decisions that have been made as part of this project, specifically where they may not form part of the strategic roadmap for information delivery and the background / justification around these decisions.

Decision taken	*Root case for the problem that necessitates the decision*	*Justification for the decision taken*	*Likely consequences from the decision*	*Is the decision compliant with the IG-TDA-OneEDW*

2 SOLUTION OVERVIEW

The solution is provided at the detailed level – "*With what* can this be achieved?" It addresses the transformations and software at a detailed level that will be required to deliver the solution.

2.1 Architecture

The purpose of this section is to clearly define upon what infrastructure the solution will be built. The architecture is built upon the high level Project End to End Design.

Indicate any deviations from the strategic infrastructure.

Indicate information flows that can be decommissioned as result from this project.

2.2 Data Sources (DS)

2.2.1 Sources

Provide information for each source system around extraction / data provisioning / delta extraction mechanisms and how the data will be sourced and transferred between the various components of the system. Indicate the connection that is used to capture the data. It is expected that a push mechanism is used, where the source system provisions the data on the Staging Platform. Indicate if deviations to this principle are applied here.

Show which source system Codes and Region Indicators are used.

This information can be given in a diagram:

Expected Source System	*Source*	*Connection Used*	*Delta extraction?*	*Push mechanism?*
For example: ECC Sirius	*For example: 2LIS_06_INV*	*SAP business content standard and custom extractors*	*Yes*	*Yes*

2.2.2 Data receipt & loading

Provide details around the extract processes such as audit processes and delta identification processes where applicable. This section will go down into individual data extract processes and detail the processing within.

If flat files are used as source extract information, provide details on the naming that is used for such files. What happens if the actual file does not comply with this naming convention?

In which Archive Location will the source files be stored? Which purging mechanism will be applied on the files?

2.2.3 Data files & volumetrics

Provide details of data files that are to be provided, along with details of estimated volumes and frequency of availability.

Indicate which files are stored in an encrypted form. If they are stored encrypted, indicate how/ where the decrypt password is stored.

Example:

Expected Source System	*Source*	*Estimated Volume (GB/#rows/width)*			*Frequency*	*Encrypted? Password stored in?*
For example: ECC Sirius	*For example: 2LIS_06_INV*	*For example: 3 GB*	*For example: 3 million rows*	*For example: row width 1000 B*	*For example: Monthly*	*No. Wallet.*

2.2.4 Servers

On what server will the extractions be landed? The so-called landing zone is given here. Create an overview for development / test / production situation.

Example:

Expected Source System	*Source*	*Server*	*Directory:*
For example: ECC Sirius	*For example: 2LIS_06_INV*	*For example: ITSG53171 (DEV)*	*For example: S2\dfs\es-groups\cor\cgt*
		For example: ITSG53172 (TST)	*For example: S2\dfs\es-groups\cor\cgt*
		For example: ITSG53173 (PRD)	*For example: S2\dfs\es-groups\cor\cgt*

.

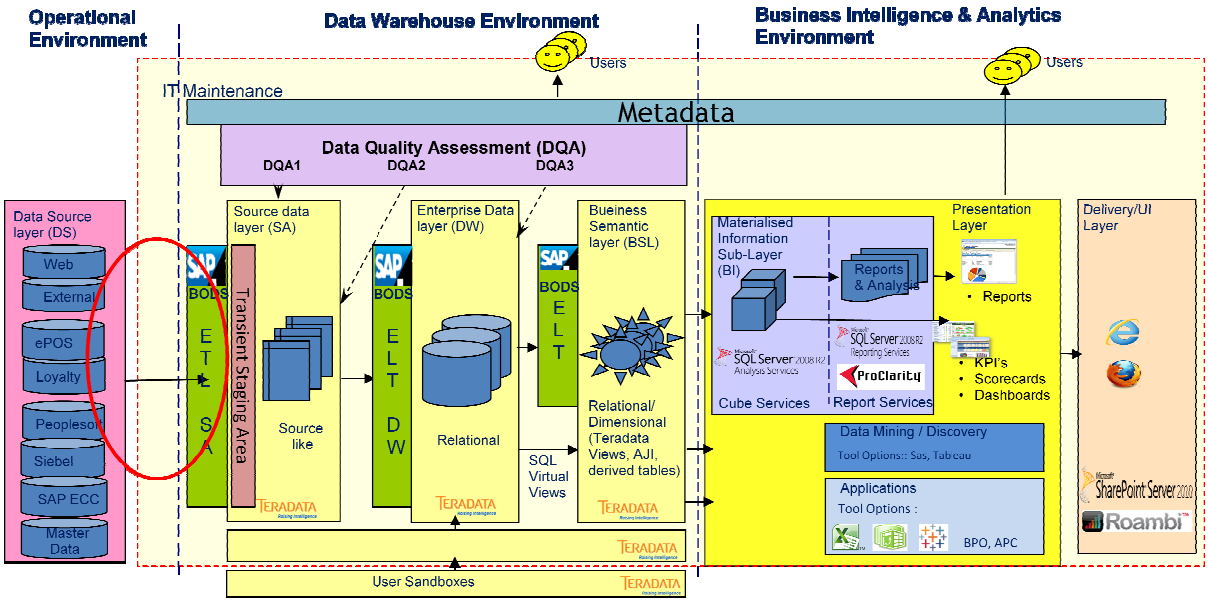
2.2.5 Master data & reference data

Detail if master data for the project will be provisioned from outside the solution. Are project delivered master data (re-)used instead? Do the same for other reference data.

2.3 Source data layer (SA)

2.3.1 Tables

Provide a reference (link only!) to the physical data model that describes the environment where source files will be captured.



2.3.2 Transformation

In principle, a one to one mapping is used in the transformation from source to the targets in the Source data layer. Provide the mappings from sources to the targets in Transient Staging Area and the Persistent Data Copy. Provide this logic on field level. Whenever a deviation from the one to one mapping is implemented, an explicit indication of the logics is required.

Indicate where the mapping logic is implemented: in Teradata via the Push-Down mechanism or in BODS. Ideally it is expected that the BODS tool controls the transformations, whereas the actual transformation is done in the Teradata DBMS. Indicate deviations from this principle.

Indicate which purging mechanism is available to avoid storage of data beyond the retention period.

Indicate the key measures that are used to reconcile data between sources and the Source Data Layer. How will these key measures be made available?

2.3.3 Performance activities

Provide detail around any performance activities that will be put into place to optimise performance to load tables in the Source data layer (SA). Here, one may include how Data Skewness is handled.

2.3.4 Databases

In which databases will the tables from the Source data layer (SA) be stored? Make a distinction between development / test/ production environment.

As an example:

Data Requirement	*Stored on*	*Database*
For example: invoice information	*Server 130.24.99.37 (DEV)*	*EDL > IPA_DV>Staging*
	Server 999.99.99.98 (TST)	*EDL*
	Server 999.99.99.99 (PRD)	*EDL*

2.4 Enterprise data layer (EDL)

2.4.1 Tables

Provide a reference to the EDL Physical Data Model (link only!) that is approved by the Tower. Indicate the tables that are used in the project, split by new tables/ re-usage and transaction versus master data

	Tables with new data	*Tables that re-use data*
Transactional Data	*√*	*√*
Master Data	*√*	*√*

Provide initial size (after initial migration) in rows and row width.

Example:

	Tables with new data	*Tables that re-use data*

Transactional Data	*99 Rows*	*Row width 99*	*99 Rows*	*Row width 99*
Master Data	*99 Rows*	*Row width 99*	*99 Rows*	*Row width 99*

Provide growth size (per load iteration)

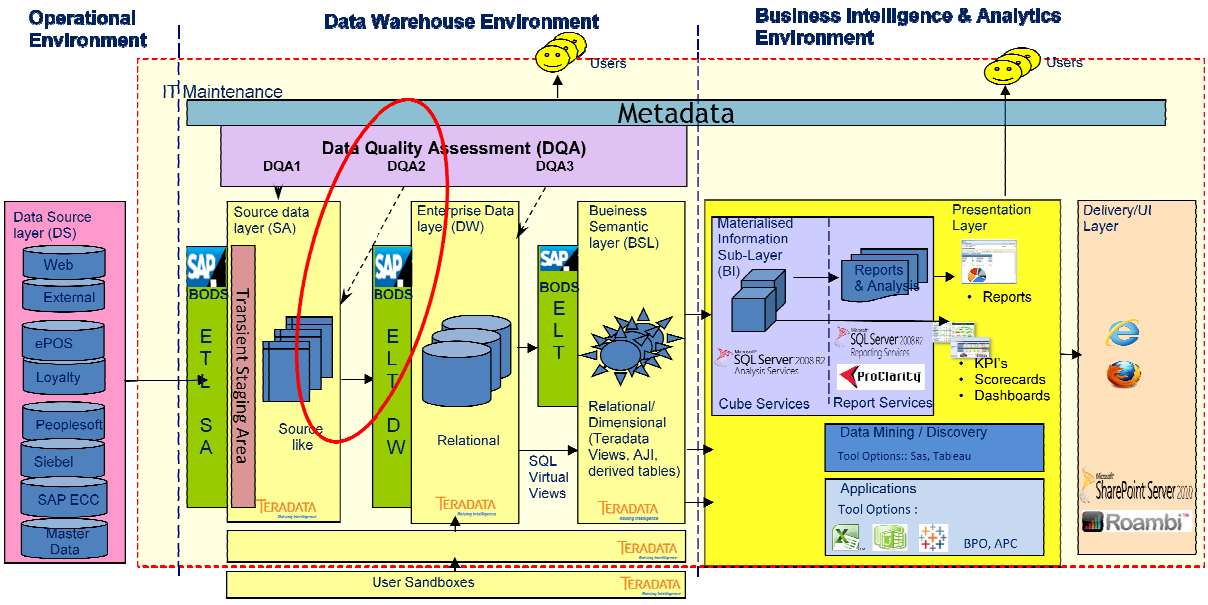
Example:

	Tables with new data		*Tables that re-use data*	
Transactional Data	*99 Rows*	*Row width 99*	*99 Rows*	*Row width 99*
Master Data	*99 Rows*	*Row width 99*	*99 Rows*	*Row width 99*

Provide iteration frequency

Example:

	Tables with new data	*Tables that re-use data*
Transactional Data	*monthly*	*monthly*
Master Data	*monthly*	*monthly*



2.4.2 Transformation

Provide the logic that is used to load the tables in Enterprise Data layer (EDL) from the Source data layer (SA). Only new transformations need to be addressed here. Provide this logic on field level. If this information is available in the DMR, a reference to the DMR is sufficient. (Link only is sufficient).

Indicate where the logic is implemented: in Teradata via the Push-Down mechanism or in BODS. Ideally it is expected that the BODS tool controls the transformations, whereas the actual transformation is done in the Teradata DBMS. Indicate deviations from this principle.

Indicate which purging mechanism is available to avoid storage of data in Enterprise Data layer (EDL) beyond the retention period. The project is responsible to design (and implement) purging mechanisms for new tables that are introduced by the project.

Indicate that in case of data enrichment, only non-destructive techniques are applied. Also, when it looks necessary to cleanse data, source data are not modified. Derived data should then be stored in their own attributes.

Indicate the key measures that are used to reconcile data between the Source data layer (SA) and the Enterprise Data layer (EDL). How will these key measures be made available?

Consider usage of Data Flow Diagrams (DFD) here. As this document will be used as a reference document, usage of such diagrams benefits future usage of this document.

2.4.3 Data cleansing & data quality

What detailed data quality processes will be put in place to ensure data is of sufficient quality to be used by the business and support key business processes?

- *Which environment is used for data quality purposes?*
- *Is full data volume being employed to assess the data quality?*
- *How will be reported on Data Quality? To whom?*
- *Which business rules will be used to assess Data Quality?*
- *In which environments are Data Quality rules implemented?*
- *Are the Data Quality rules scheduled?*

If data issues are found, where will data cleansing be carried out?

2.4.4 Performance activities

Provide detail around any performance activities that will be put into place to optimise performance to load tables in the Enterprise Data layer (EDL).

2.4.5 Databases

In which databases will the tables from the Enterprise Data layer be stored? Make a distinction between development / test/ production environment.

As an example:

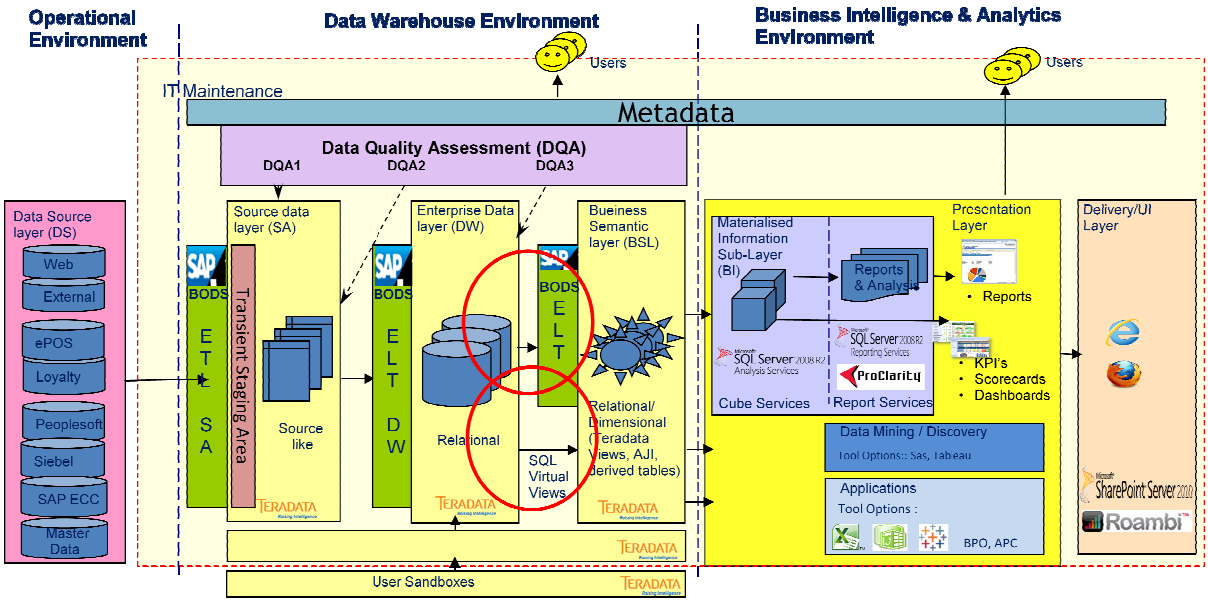
Data Requirement	*Stored on*	*Database*
For example: invoice information	*Server 130.24.99.37 (DEV)*	*EDW > IPA_DV>EDL*
	Server 999.99.99.98 (TST)	*EDW > IPA_TST> EDL*
	Server 999.99.99.99 (PRD)	*EDW > IPA_PRD> EDL*

2.5 Business Semantic Layer (BSL)

2.5.1 Views

Provide a reference to the Physical Data Model for the Semantic Layer that is approved by the Tower.

Indicate which Global Master Hierarchies are used. Indicate which local hierarchies are used.



2.5.2 Transformation

What views will be instantiated in the semantic layer as part of this project and what information will they contain, for example join conditions, locking mechanisms etc. Provide this logic on field level. If this information is available in the DMR, a reference to the DMR is sufficient.

2.5.3 Performance activities

Provide detail around any performance activities that will be put into place to support the performance of the views, for example AJI's, Statistics collection, table partitioning etc .

2.5.4 Databases

In which databases will the views from the Semantic Layer be stored? Make a distinction between development / test/ production environment.

As an example:

Data Requirement	*Stored on*	*Database*
For example: invoice information	*Server 130.24.99.37 (DEV)*	*EDW > IPA_DV>Semantic*
	Server 999.99.99.98 (TST)	*EDW > IPA_TST> Semantic*
	Server 999.99.99.99 (PRD)	*EDW > IPA_PRD> Semantic*

2.6 Reporting & analytics

2.6.1 Usage of the data from Semantic Layer / EDL for reporting

In this section, the usage of the Semantic Layer / EDL as source for reporting is discussed. Items to be addressed are:

- *Which mechanism is used to transfer data from the Semantic Layer to the Reporting Environment? Note: a push mechanism is preferred.*
- *Does the introduction of the Reporting environment lead to a situation where the EDL starts being a "System Of Records". In that case, the legal consequences should be given.*

2.6.2 Reporting data structures

Provide information with regards to the data structures that will be implemented / utilised as part of this solution. This includes details of ROLAP Cubes & dimensions that will be utilised for reporting. Here, information can be given on the hierarchies.

Provide the logic that is used in the reporting environment; provide this logic on field level.

If the project writes a separate design document for the Front-End, a link to the design is sufficient.

2.6.3 Report Front End

A description of each of the reports can be given here. If the project writes a separate design document for the Front-End, a link to the design is sufficient.

2.6.4 Data access considerations

How are the report accessed. Is this done from a Portal? Which portal is used? What data access considerations are there? This includes details around data security and limiting access to certain users / departments / geographies etc.

2.7 Allocation

It might be that Enterprise Data layer (EDL) is used for allocation purposes. This is understood as data being distributed according to data in the EDL. In that case, this section can be used to provide a design. Attention should be given to:

- *What allocation rules are foreseen? What is the level of simplicity of allocation rules; in general EDL is not meant for complicated allocation rules.*
- *Which tables are used in the calculation of such rules?*
- *Is the calculation required on a scheduled base?*
- *What is the usage of the allocated data: is this limited to reporting and / or planning purpose only?*
- *Does the calculation of the allocation factor lead to a situation whereby the EDL starts being a "System Of Records". In that case, the legal consequences should be given.*

2.8 Capacity planning

2.8.1 Initial volumes

Provide information around the initial data volumes that are involved in the solution, for example what data will be migrated from historical systems.

2.8.2 Incremental volumes

What growth volumes of data will be provisioned as part of the regular batch processes? This should tie in with the data file volumetric provided earlier.

2.9 User profiles and security

2.9.1 Personal Users

What will the users be doing with the system when delivered? Will they be heavy analytical users or "lighter" operational users? How many of each type of users are expected and when? Where are the users located and how will they access the tools?

Which service accounts are implemented? Make a distinction between the dev / test / production environment.

2.9.2 System Accounts

Which system accounts are used? For what purpose are they used?

2.9.3 Data Security for data at rest

What security measures are implemented to protect data at rest? Make a difference between the Data Source Layer (DS) and data that are stored in databases (SA, EDL, BSL). Are the data encrypted?

Provide the security mapping of end users to data access requirements. Indicate the Teradata roles that are used.

What restricted information do we create in this project. In which databases will this be stored?

What access mechanisms are provided to the data? Think of SQL Assistant, access via Excel PowerPivot, Tableau, Sharepoint etc. What security mechanisms are created: Active Directory, Teradata roles etc. How do they interact?

2.9.4 Data Security for data in motion

What security measures are implemented to protect data in motion? Make a distinction between the

- *different modes of transport (for example BODS) and*
- *the type of flow (for example between Datasource Layer (DS) and Staging Area(SA), between SA and EDL etc.)*

2.10 Network requirements

Is there are requirement to transmit significant levels of data across the WAN for example or will all data transfer be limited to within data centres?

2.11 Data retention requirements

- *There will be program level data retention policies but does the project require anything above this – for example do records have to be kept for 10 years for regulatory reasons?*
- *It is assumed that the data retention period is equal between the Source Data Layer (SA) and the Enterprise Data Layer (EDL). If this project needs to deviate from that assumption, plse indicate so.*
- *Provide a list of tables that are created within the project in the SA with the retention period.*
- *Provide a list of tables that are created within the project in the EDL with the retention period.*
- *It might well be that the implementation of the retention requires a certain order of cleanup (because of referential integrity). Provide such an order here.*

2.12 Archiving & back up

There will be program level archive policies but does the project require anything above / different to this?

2.13 Metadata

Detail how metadata capture will be facilitated and in particular detail any deviations from the metadata capture and integration design standards.

2.14 Control table

Whenever control tables are used, one may provide here a list of such control tables. Provide for each control table its purpose. Moreover, indicate how such table can be updated, when required.
Example: when a table contains a list of years that for which data must be shown, the table must be updated when a new year starts.

3 DATA MIGRATION

3.1 Source systems

What source systems are in scope for this project and what data is required at the start of production? Where are these data located? How easy is the data to extract and what tools will be used to this?

3.2 Migration approach

Provide detail around the approach used for data migration – will it be a take everything-once-approach or will it require a number of smaller data migrations?

4 COMPLIANCE WITH PROGRAM STANDARDS

This section should detail the non compliance with program standards and should refer to the design compliance statement which should also be completed by the project and reviewed by the relevant TDA members.