|  |
| --- |
|  |
|  |
| SOLUTION ARCHITECTURE  VISION BI |

Versions: 9.0

Project code: VISION BI

Document Serial: Vision\_ARC\_v9.0.doc

Date of Modification: 3rd December, 2018

Reviser: Karthik Dhanasekaran

Mục lục

Table of contents

[MỤC LỤC 2](#_Toc531599427)

[TABLE OF CONTENTS 2](#_Toc531599428)

[A. VERSION CONTROL 6](#_Toc531599429)

[I. VERSIONS 6](#_Toc531599430)

[II. APPROVAL 7](#_Toc531599431)

[B. DOCUMENT INTRODUCTION 8](#_Toc531599432)

[I. INTRODUCTION / OVERVIEW 8](#_Toc531599433)

[II. OBJECTIVE / PURPOSE 8](#_Toc531599434)

[III. SCOPE 8](#_Toc531599435)

[IV. TERMINOLOGY 9](#_Toc531599436)

[V. REFERENCE 10](#_Toc531599437)

[C. PROJECT OVERVIEW 11](#_Toc531599438)

[I. OBJECTIVE 11](#_Toc531599439)

[I.1. OBJECTIVE 11](#_Toc531599440)

[I.2. OUTCOME 11](#_Toc531599441)

[II. PROJECT SCOPE 11](#_Toc531599442)

[II.1. IN SCOPE 11](#_Toc531599443)

[II.2. OUT SCOPE 12](#_Toc531599444)

[III. CONSTRAINTS 12](#_Toc531599445)

[III.1. ASSUMPTION 12](#_Toc531599446)

[III.2. CONSTAINTS 13](#_Toc531599447)

[III.3. DEPENDENCY 13](#_Toc531599448)

[III.4. PROJECT SCHEDULE 14](#_Toc531599449)

[D. REQUIREMENTS 15](#_Toc531599450)

[I. SYSTEM CONTEXT MODEL 15](#_Toc531599451)

[II. BUSINESS PROCESS 16](#_Toc531599452)

[III. BUSINESS REQUIREMENTS 16](#_Toc531599453)

[IV. NON FUNCTIONAL REQUIREMENTS 17](#_Toc531599454)

[V. SECURITY REQUIREMENTS 18](#_Toc531599455)

[E. SOLUTION ARCHITECTURE 22](#_Toc531599456)

[I. DATA ARCHITECTURE 22](#_Toc531599457)

[II. SECURITY DESIGN 25](#_Toc531599458)

[APPLICATION FIREWALL 25](#_Toc531599459)

[SECURED SOCKET LAYER (SSL) 25](#_Toc531599460)

[BROWSER 25](#_Toc531599461)

[FIREWALL & PROXY 25](#_Toc531599462)

[VISION SYSTEM – SECURED WEB SERVER 26](#_Toc531599463)

[ENCRYPTION 26](#_Toc531599464)

[APPLICATION SECURITY MECHANISM IN VISION 26](#_Toc531599465)

[ACCESS RIGHTS BY MODULE / USER 26](#_Toc531599466)

[AUDIT TRAILS 26](#_Toc531599467)

[AUTO LOGOFF 27](#_Toc531599468)

[SESSION ID & SESSION TRACKING IN VISION 27](#_Toc531599469)

[III. DEPLOYMENT ARCHITECTURE 28](#_Toc531599470)

[III.1. SYSTEM ARCHITECTURE 28](#_Toc531599471)

[III.2. DATA BACKUP & RECOVERY 28](#_Toc531599472)

[III.3. SERVER SIZING & DATA STORAGE 29](#_Toc531599473)

[III.4. REQUIREMENTS ON SOFTWARE 29](#_Toc531599474)

[III.5. REQUIREMENTS OF WORKSTATION 29](#_Toc531599475)

[III.6. REQUIREMENTS ON MONITORING 30](#_Toc531599476)

[IV. CONSTRAINTS 30](#_Toc531599477)

[IV.1. ABILITY OF EXTENSION 30](#_Toc531599478)

[IV.2. DATA INTEGRITY 31](#_Toc531599479)

[V. VISION FRAMEWORK OVERVIEW 32](#_Toc531599480)

[VISION TECHNOLOGY 33](#_Toc531599481)

[PRESENTATION TIER 34](#_Toc531599482)

[NAVIGATION & WORKFLOW TIER 35](#_Toc531599483)

[INTEGRATION TIER 35](#_Toc531599484)

[SALIENT FEATURES 36](#_Toc531599485)

[VI. ARCHITECTURAL RISK 37](#_Toc531599486)

[F. ARCHITECT EVALUATION 38](#_Toc531599487)

1. Version control
   1. VersionS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S. No** | **Created / Modified By** | **Approved By** | **Version** | **Date** |
| 1 | Karthik Dhanasekaran | Suleman Dossani | 1.0 | 23.5.2018 |
| 2 | Karthik Dhanasekaran | Suleman Dossani | 2.0 | 15.8.2018 |
| 3 | Ramprakash Thangavel | Suleman Dossani | 3.0 | 20.8.2018 |
| 4 | Karthik Dhanasekaran | Suleman Dossani | 4.0 | 22.8.2018 |
| 5 | Karthik Dhanasekaran | Suleman Dossani | 5.0 | 28.8.2018 |
| 6 | Karthik Dhanasekaran | Suleman Dossani | 6.0 | 12.9.2018 |
| 7 | Karthik Dhanasekaran | Suleman Dossani | 7.0 | 25.9.2018 |
| 8 | Karthik Dhanasekaran | Suleman Dossani | 8.0 | 4.10.2018 |
| 9 | Karthik Dhanasekaran | Suleman Dossani | 9.0 | 3.12.2018 |

**Revision History V 4.0 to 5.0**

|  |  |
| --- | --- |
| **S. No** | **Changes Summary** |
| 01 | Updated Explicit Scope Exclusion |
| 02 | Updated Harware Architecture Digram |
| 03 | Updated Production DR Syncronization Diagram |
| 04 | Updated Production Server Specification under Production & DR Server Specification |
| 05 | Added Vision Main Table Components under ETL |
| 06 | Updated Framework Architecture Diagram under Vision Framework Technical Overview |

**Revision History V 5.0 to 6.0**

|  |  |
| --- | --- |
| **S. No** | **Changes Summary** |
| 01 | Updated Explicit Scope Exclusion |
| 02 | Updated Harware Architecture Digram |
| 03 | Updated Production DR Syncronization Diagram |
| 04 | Updated Production Server Specification under Production & DR Server Specification |
| 05 | Updated Vision Main Table Components under ETL |
| 06 | Updated Vision Modules and Key Functionailities |

**Revision History V 6.0 to 7.0**

|  |  |
| --- | --- |
| **S. No** | **Changes Summary** |
| 01 | Updated Production DR Syncronization Diagram & Notes |
| 02 | Updated MSB Application Architecture Diagram |

**Revision History V 7.0 to 8.0**

|  |  |
| --- | --- |
| **S. No** | **Changes Summary** |
| 01 | Updated Application Server – Hardware/OS – Virtual Server Specifications |

**Revision History V 8.0 to 9.0**

|  |  |
| --- | --- |
| **S. No** | **Changes Summary** |
| 01 | Updated Document Format to Maritime Bank Architecture Document Standard |

* 1. APPROVAL

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Name | Position/Role | Approval (Y/N) | Sign/Note |
| 1 | Sai Prasad Tummapudi | Chủ tịch hội đồng kiến trúc |  |  |
| 2 | Nguyễn Viết Sao | Head of Operation Center |  |  |
| 3 | Nguyễn Oen | Architect |  |  |
| 4 | Nguyễn Thị Minh Hà |  |  |  |

1. dOCUMENT INTRODUCTION
   1. INTRODUCTION / OVERVIEW

This introduction provides an overview of the entire Software Architecture Document for Vision BI System. It includes the objective / purpose, scope, terminology and references. This document consists of 6 sections, which are described below:

* Section A includes this document version control and document approver information
* Section B is an architecture document overview
* Section C describes project overview, objective, scope, contraints and project schedule
* Section D describes requirements with reference to system context model, business process, business requirement, non-functional requirements and security requirements
* Section E describes the solution arcitecture from the perspective of business architecture, component architecture, data architecture, security design, deployment architecture, constraints and architectural risk
* Section F includes architect evaluation
  1. OBJECTIVE / PURPOSE

This document provides a comprehensive architectural overview of the Vision BI Application, using a number of different architectural views like business, application, component and deployment architecture. The primary purpose of the Vision Business Intelligence (BI) System is to satisfy the reporting and analytical capabilities in Maritime Bank.

This document is intended to capture and convey the significant architectural decisions which have been made in designing and building the Vision BI system.

* 1. SCOPE

The scope of this architecture document is to depict the architecture of the Vision Business Intelligence application created by the Sunoida Solutions DMCC.

* 1. TERMINOLOGY

|  |  |
| --- | --- |
| **Addressability** | Ability to model the scoping information in a consistent way |
| **Application Components** | Components in the implementation architecture capturing the domain level responsibilities. |
| **Behaviour** | Description of how the system achieved what it is intended to do |
| **Conceptual Architecture** | Model of the architecture focusing on domain level responsibilities, i.e. what groups of functionality does exist and how do these groups interact to achieve a certain goal, e.g. a single use case |
| **Data-Centered Architecture** | An architecture with the goal to achieve data integrity, typically by choosing a central componentfor datamanagement, e.g. a database. |
| **Data-Flow Architecture** | An architecture with the goal to achieve decoupling between components, by defining a data flow (pipes) with a series of transformations (filters). |
| **Execution Architecture** | Model of the architecture focusing on the runtime aspects, i.e. what types of parallel executing components exist. |
| **Functionality** | Description of what the system can do |
| **Implementation Architecture** | View of the architecture focusing on how the system is build |
| **Information Flow** | Connector used in the conceptual architecture used to model what type of information is needed for component to accomplish its responsibilities. |
| **Infrastructure Components** | Components in the implementation architecture designed to make the system run. |
| **Layered Architecture** | Architecture pattern that organises the components into individual layers (on top of each other) with restrictions on the connections between the components. |
| **Metrics** | The non-runtime quality attributes: maintainability, evolvability, testability, reusability, integrability, configurability, scalability. |
| **Model** | Abstraction of the system, focusing on a single (or multiple) aspect, e.g. security, domain level responsibilities, executions. |
| **Quality Attributes** | Key characteristics of a software system, including the run and the build time. |
| **SOA** | Service Oriented Architectures |
| **SOAP** | Simple Object Access Protocol - a protocol designed to exchange structured information between services (used for method calls). |
| **WSDL** | Web Services Description Language - language designed for the specification of interfaces, in particular for services. |
| **System Context Model** | Representation all external entities that may interact with a system. Such a model pictures the system at the center, with no details of its interior structure, surrounded by all its interacting systems, environments and activities. The objective of the system context diagram is to focus attention on external factors and events that should be considered in developing a complete set of systems requirements and constraints. |

* 1. REFERENCE

Software License and Services Agreement for the Implementation of Vision Banking Business Intelligence (**SLSA: SSV1178A)**

Pre-Deployment Study (PDS) for the Implementation of Vision Banking Business Intelligence (**Signed on 25th June, 2018**)

1. PROJECT OVERVIEW
   1. OBJECTIVE
      1. OBJECTIVE

The key objective of the Vision Business Intelligence (BI) project is to implement a centralized management information system (MIS) which will enable Maritime Bank to efficiently and effectively monitor its business performance and adequately support Finance, Business, IT in terms of reporting & analytics, performance monitoring of branches by the different departments. Vision MIS needs to result in an improvement of the overall quality of the management reports especially in the presentation format of the reported information (dashboards, widgets, standard reports and alerts) across Maritime Bank.

* + 1. OUTCOME

Vision is a web-based MIS solution that delivers rules-based enterprise wide analytics having Vission BI Data marts as single source of truth for all reporting solution across Maritime Bank. It is the foundation of a quantifiable successful corporate performance management solution. The key is the accuracy, enterprise-wide accessibility and usability of the performance measurement data spread across 12 dimensions. Vision provides reporting and analytical capabilities by account, product, branch, business line, relationship, sector, currency, business segments and a number of other dimensions and attributes for Maritime Bank.

* 1. PROJECT SCOPE
     1. IN SCOPE

The following modules of Vision BI are in scope of the implementation:

* + ETL
  + Financial Module
  + MIS Module (Including Fund Transfer Pricing)
  + Cost Allocations
  + Vision Catalog: Ad-hoc Reporting Tool
  + 115 Standard Reports and Dashboards
  + 450 Report Rationalization
    1. OUT SCOPE

The following are the out of scope activities to provide by Maritime bank team for achieving the project outcome:

* Maritime Bank’s technical staff will provide UAT environment and access to other pheripheral software systems listed in Pre-Deployment Study (PDS) document for Sunoida Team to carry out the implementation.
* Maritime Bank Finance team will provide the business requirement documents for all the reports covered under the scope of implementation.
* Maritime Bank will provide acceptance criteria for each functionality and conduct user acceptance testing for all the developed functionality / reports in the UAT environment and sign-off.
* Maritime Bank’s technical staff will provide Production environment for Go-Live and activites related to DR Server set-up / configurations.
  1. CONSTRAINTS
     1. ASSUMPTION

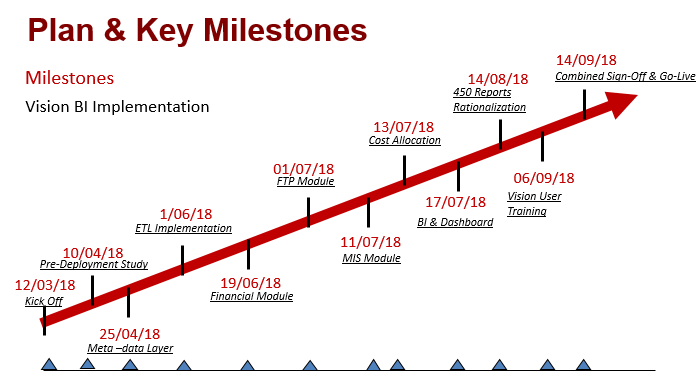
Below list provides assumptions which will be monitored / updated during project duration on regular basis.

* Maritime Bank will provide the business knowledge and requirements based on the scope of implementation
* The implementation will follow the guidelines and milestones in the statement of work and will be further defined in the detailed pre-deployment study (PDS) document that will be agreed and signed off before the Implementation of work in each stage of the project.
* Maritime Bank’s resources will be deployed for appropriate tasks as mutually agreed and in line with the signed off project plan. Sunoida team will requires Maritime Bank’s resources immediate support for ad-hoc queries on case-to-case basis.
* The project plan will be mutually discussed and agreed between the Sunoida project manager and Maritime Bank project manager, at the start of the project. Once agreed, the Maritime Bank and Sunoida Project Managers will jointly drive the project timeline.
* Maritime Bank’s senior management will be involved in the decision making process, especially on issue resolution and change management. Mutually agreed interpretation of the functionality will prevail.
* Maritime Bank’s team/s will prepare the scope of the conversion strategy and drive the reconciliation process. This will be jointly reviewed and mutually agreed between Maritime Bank and Sunoida.
* Sunoida will retain intellectual property rights for any custom work (if any) undertaken as part of this project, in far as it relates to Sunoida products and processes.
  + 1. CONSTRAINTS

Below list provides the summary of constraints:

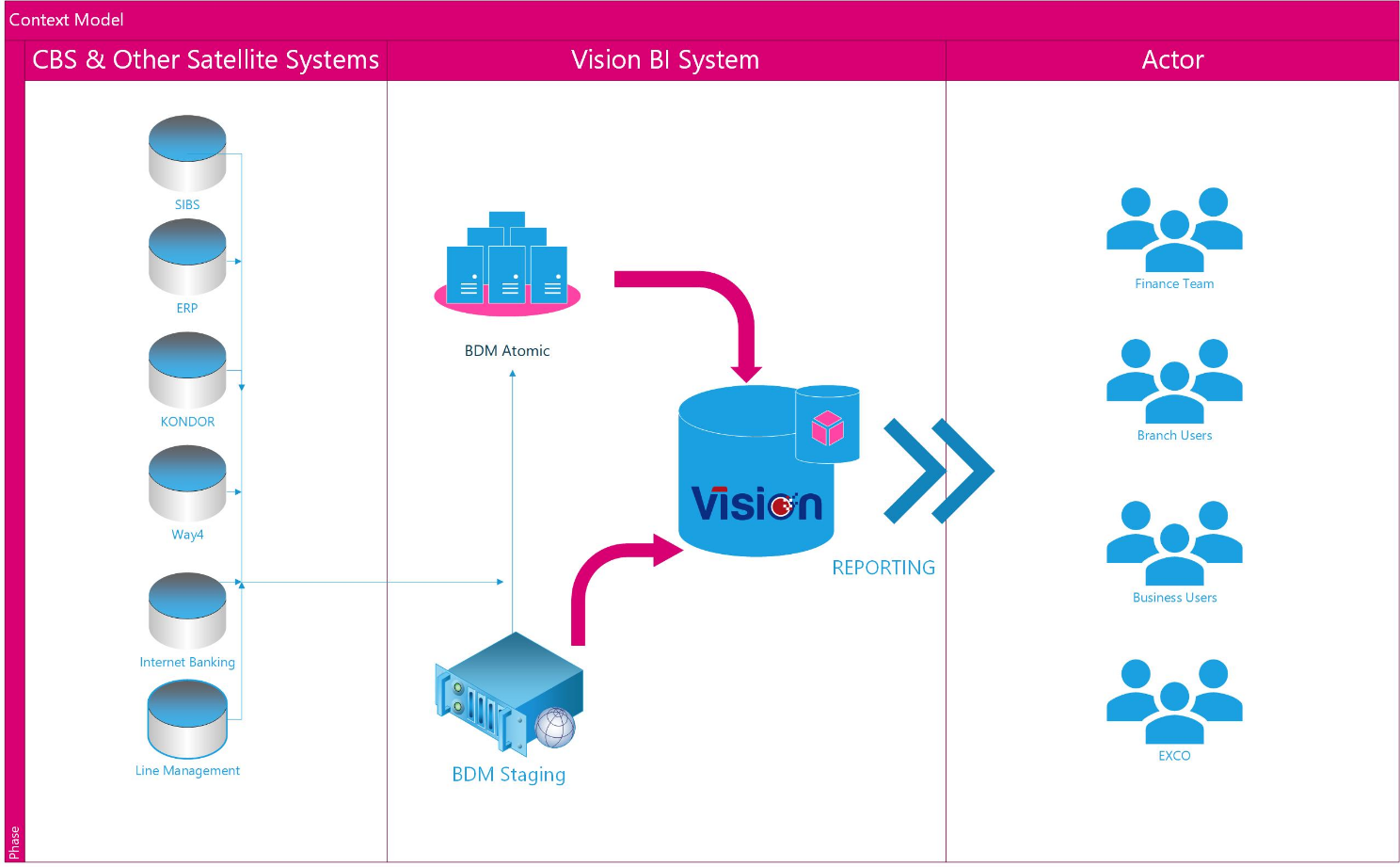
* Vision BI Implementation project’s budget, schedule and scope need to be aligned to the overall benefits of the intended output using benefits management.
* Maritime Bank’s satisfaction should be measured with a view to delivering the agreed benefits during Vision BI project and the operation of its end output as report delivery.
* Integrating Vision BI into the MSB organization structure, it is being delivered for depends on successful implementation planning
* Team dynamics, the organizational structure in which the project exists within and the methodology used to implement the project are important constraints that impact project success
* Maritime Bank's team need to update the IBM BDM staging data on daily basis to serve its purpose of facilitating the data requirements to Vision BI Application as IBM BDM remains the single source of information for Vision reporting for Finance & Business Units. Any challenges resulting in data in-consistency with IBM BDM will remain as a contstraint
* Vision Application DB and Application Server should not be down for active performace of the BI Application
  + 1. DEPENDENCY

Below list provides the dependencies which Sunoida team require during implementation.

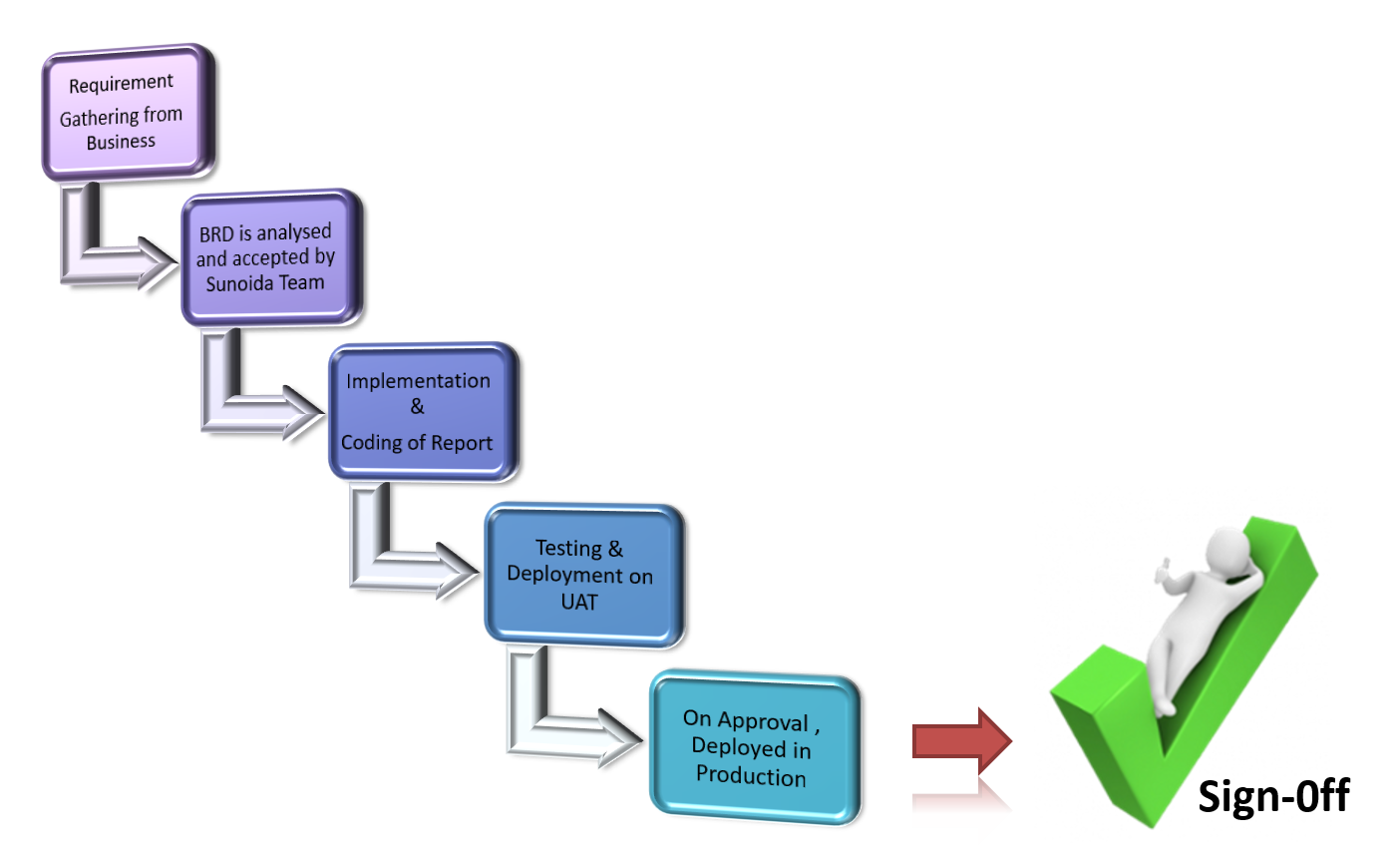
* Maritime Bank’s technical staff will provide UAT environment and access to other pheripheral software systems listed in Pre-Deployment Study (PDS) document for Sunoida Team to carry out the implementation.
* Maritime Bank Finance team will provide the business requirement documents for all the reports covered under the scope of implementation.
* Daily EOD must be completed on time and Provide data to Vision for Timely reporting to all business across MSB
* Daily Basis IT support is required to carry out EOD and EOM activities on Vision
  + 1. PROJECT SCHEDULE

1. REQUIREMENTS
   1. system context MODEL

The following System Context Model show Vision BI application, as a whole and its [inputs](https://en.wikipedia.org/wiki/Input/output) and [outputs](https://en.wikipedia.org/wiki/Output_(computing)) from/to external factors.



* 1. BUSINESS PROCESS

The following business process summarizes the life cycle of the report implementation from an end user prespective. 

* 1. BUSINESS REQUIREMENTS

The Business Requirements includes the following modules of Vision BI to be deployed:

* + ETL
  + Financial Module
  + MIS Module (Including Fund Transfer Pricing)
  + Cost Allocations
  + Vision Catalog: Ad-hoc Reporting Tool
  + 115 Standard Reports and Dashboards
  + 450 Report Rationalization

As well as the following key requirements agreed and signed on the Pre-Deployment Study Document by Mr. Nilesh & Ms. Hang on 26th June, 2018.

* + Legal Vehicle
  + No. Of Entities
  + Vision Meta Data Definitions / Dimensions
  + FRL Family / MRL Family
  + Financial Adjustments
  + MIS Adjustments
  + FTP Rates
  + Currency Rates/LV Rates
  + Products
  + OUC Hierarchy
  + SBU Hierarchy
  + SBU Grouping
  + FX Income
  + KPI
  + Incentive Calculation
  + Cross Sell
  + Supply Chain
  + Relationship Manager Derivation
  + Business rule for Currencies
  + Business rule for Double Booking
  + Business rule for Balances
  + Compution of Average Balances in Vision
  1. NON FUNCTIONAL REQUIREMENTS

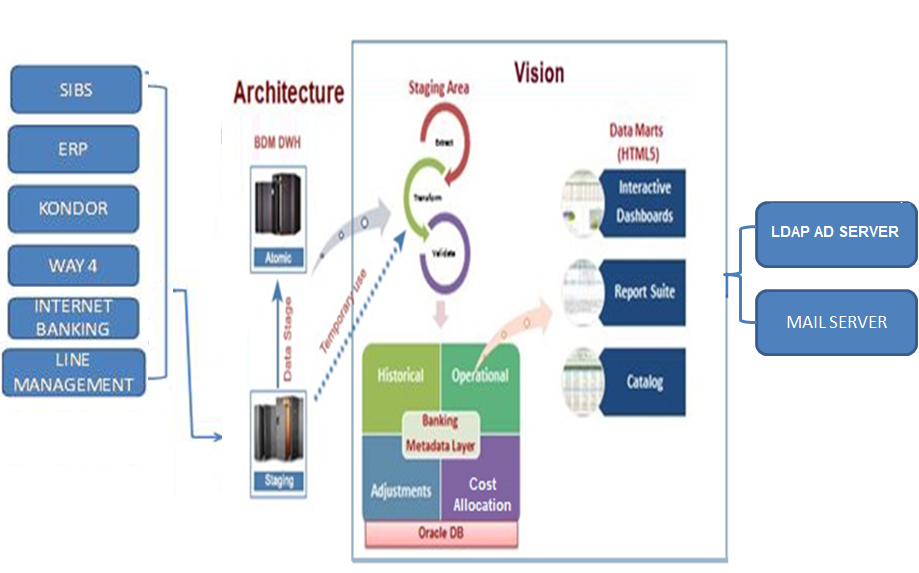
|  |  |  |
| --- | --- | --- |
| # | Non function requirement | Description |
| 1 | Total number of users/ concurrent users | 200 concurrent users |
| 2 | Availability requirements | NA |
| 3 | Support Availability | Offshore support during Vietnam Business Hours only |
| 4 | Performance Requirements | All the Vision Reports run less than 1 minute only |
| 5 | Backup Data and Schedule | Sunoida recommends daily incremental backup |
| 6 | Data Volume | Year 1 (average 1 TB)  Year 2 (40% increase)  Year 3 (40% increase)  Year 4 (40% increase)  Year 5 (40% increase) |
| 7 | Data retention period requirement | Currently Vision has enabled for 10 years as Data Retention period, which can be altered based on Maritime Bank’s request |

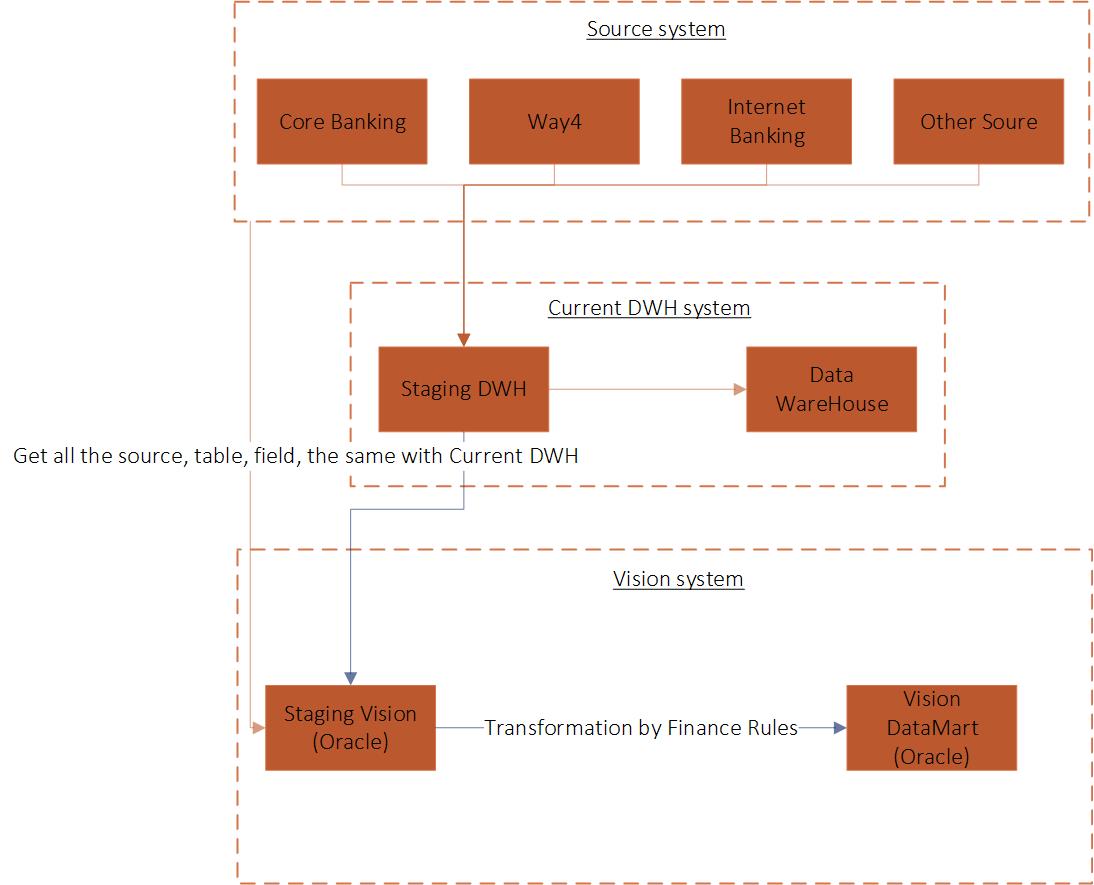
* 1. SECURITY REQUIREMENTS

The following are the security requirements provided by Maritime Bank for Vision BI Implementation:

|  |  |
| --- | --- |
|  | **Security** |
|  | **01 Authentication** |
| 1 | The Application must support centralized authentication (e.g. integrate with AD) |
| 2 | The Application must not use hard-coded IDs, passwords, connection strings or other sensitive authentication credentials. |
| 3 | Authentication credentials for accessing external services (external web service, database...) are encrypted. |
| 4 | Authentication credentials must be masked when entered at the login prompt and encrypted during transmission |
| 5 | Application support log-out function |
| 6 | The application must support to show Logon Banner "Access is for MSB authorized staff only. If you are not authorised, do not attempt to logon. |
|  | **02 Authorization** |
| 1 | The application must support group-level security and permission that have capablities for group creation and right assignment |
| 2 | The application must support role-based access control and the authorization is processed at server site |
|  | **03 Session Management** |
| 1 | The application Mandatory session token be provided by the application framework |
| 2 | The application Mandatory session token be unpredictable or can't be guessable |
| 3 | Sessions must expire after Specifies period of inactivity (r.g. 15 minute) |
| 4 | Session id: -Must change on login to prevent session fixation. -Must destroy or re-create to ensure the old id Is invalid when users log out Store only in cookies, not others. The cookies are deleted after userr close browsers -The application will not support to allow user create specific session id -Create by strong algorithm, length at least 128 bits |
| 5 | The application should generate strong random anti-CSRFtokens unique to the user when accessing sensitive data, and that the application verifies the presence of this token with the proper value for the current user when processing these requests. |
|  | **04 Input & Output Validation** |
| 1 | Vision will be deployed on your intranet hence all incoming information into Vision should be from trusted sources. |
| 2 | The application must support to have input validation including checks for common escape sequences, files types (example: block execute files), files size. The application should utilize a "white list" for input validation |
| 3 | Input validation must be performed on the server-side. Client-side security checks must also be performed on the server-side |
| 4 | Queries to database must make use of parameterized queries or parameterized stored procedures. The application does not allow users create directly enquire to database |
| 5 | The application must not connect to a database as a privileged user. For example, the SA accounts in SQL Server or SYSTEM account in Oracle |
| 6 | The application must support to have output to be entity encoded to the appropriate content type (e.g. HTML encoding, JavaScript encoding, URL encoding, CSS encoding etc.) to prevent it from being treated as a script. |
| 7 | URL redirection must not accept direct input from the user. Additionally, the application must encode the URL redirection to unrecognizable format (e.g. hasted file location). |
|  | **05 Cryptography** |
| 1 | The application must support encryption keys to be protected during transit and while stores in file system |
| 2 | The application must support password to be hashed with a strong algorithm and an appropriate salt is used. |
| 3 | The application must support to have the key used to decrypt data must not be stored in the same location as data encrypted with the keys. |
| 4 | Password/ Encryption keys must not be disclosed to anyone who does not need access to them. |
|  | **06 Logging and Auditing** |
| 1 | The application must support the logging for all user and account activity |
| 2 | The log will not store sensitive information such as session ID, Passwords. |
|  | **07 Error Handling** |
| 1 | The application must support so all errors, exceptions, and other error condtions to be detected by the application and handled appropriately. |
| 2 | The application must not display detailed error messages to the user |
| 3 | The application must display a genric web page or generic error message when an error condition occurs. |
| 4 | The application must not log application-specific sensitive data that could assist an attacker, including user's session identifiers and personal or sensitive information. |
|  | **08 Data Protection** |
| 1 | The application must support to verify all forms containing sensitive information have disabled client  side caching, including autocomplete features. |
| 2 | The application must support so all sensitive data is sent to the sever in the HTTP message body using HTTP/TLS (i.e. URL parameters are never used to send sensitive data) |
|  | **09 HTTP Security** |
| 1 | All communication between the user browser and the server, server and 3rd party application must be encryption (HTTP,SFTP...) with strong algorithm |
| 2 | The application must accepts only a defined set of HTTP request methods, such as GET and POST and unused methods are explicity blocked. |
|  | **10 Account Management** |
| 1 | The application must support system administrator to manage account such as assign/adjust role, create /delete account. |
| 2 | The application must support to generate and automatic send report about access role/permission to specific email address for periodic review |
|  | **11 Vulnerability Managemnt** |
| 1 | Application must use the latest/up-to date version of component such as framework, plugin |
|  | **12 Application Partitioning** |
| 1 | The application must support to have the Back-end be seperated from Front-end Website |

1. SOLUTION ARCHITECTURE
   1. Data architecture





The above architecture is provided based on the requirement from MSB architecture team.

Note: All financial data will goto histoical data and operational data will goto Vision Master Tables.

The ETL segment gets the feed ONLY from the IBM BDM System and not from any other source systems through DBLINK. The information which flows in contains one time baseline of customer, accounts, financial & static information of finances, deposits, etc. The data is then loaded into the Vision staging tables and after validation it flows into the Vision Data Warehouse.

The master requirements for Vision BI Implementation for the scope is summarized as an excel file embedded below.



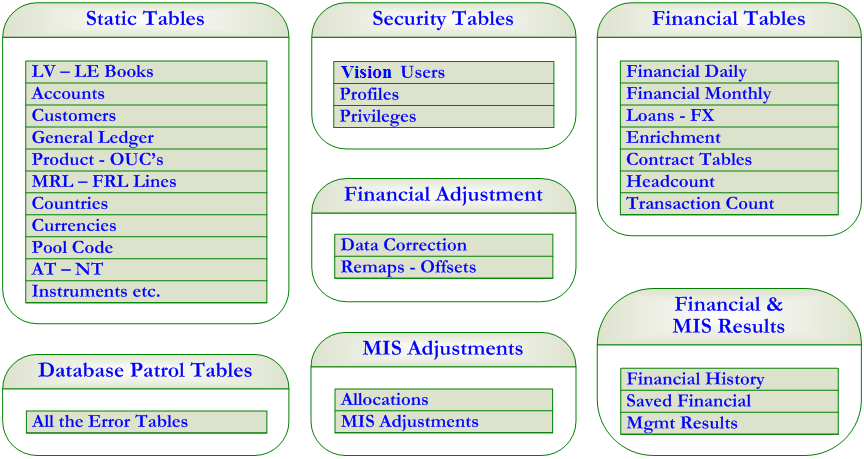
MSB Bank's team need to update the IBM BDM data on daily basis to serve its purpose of facilitating the data requirements to Vision BI Application. In order to do this, data from the core banking and other operational system needs to be extracted and moved into the IBM BDM Staging and then to IBM BDM Atomic through IBM ETL Data stage.

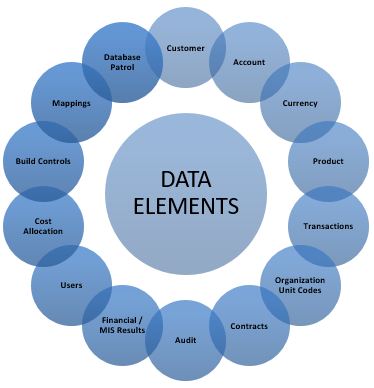
Vision maps the data from IBM BDM Atomic to Vision staging tables and provides the opportunity to enrich that data. Based on business rules and logic, these values can be populated by default; or specific values can also be defined as the requirement may be. Vision can also maintain a history and audit trail of all changes to the data loaded in the data warehouse.

As the load phase interacts with a database, the constraints defined in the database schema — as well as in triggers activated upon data load — apply (for example, uniqueness, referential integrity, mandatory fields), which also contribute to the overall data quality performance of the ETL process.

NOTE: IBM BDM remains the single source of information for Vision reporting for Finance & Business Units.

**VISION MAIN TABLE COMPONENTS**





The process of transforming the data procured thru ETLs into a the form of reports/dashboards/charts that is useful for the management is achieved thru a set of programs, frequently referred to as “Builds” or “Build Programs” maintained in Build Control tables. All the mapping structure for financial & management reporting are maintained in Mapping tables

* 1. security design

The objective is to explain how the Vision handles security aspects throughout the application. Basically the security issues are addressed in the following manner.

* Vision Application System Security
* System Security using third party software’s like SSL

## 

## APPLICATION FIREWALL

* Packet Filtering Firewall
* Will allow only HTTP/HTTPS traffic coming from the Content Server Static IP
* Will Disable PING/TELNET/FTP & other applications
* Will allows access to only specific destination Ports/Addresses
* Provide auto alerts via email in case of security breach
* Maintain Logs of all packet
* Generally installed on a separate Server in the data center

## SECURED SOCKET LAYER (SSL)

Vision System can be configured to support SSL solutions. This ensures that communications is secured and encrypted.

## 

## BROWSER

The thin client approach is used for Vision System with all applications generating HTML pages. This enables Vision to be independent of the types of the browser being used.

## 

## FIREWALL & PROXY

Access from behind a Firewall or Proxy Server will not hinder the access to Vision since Vision uses standard http and https traffic.

## 

## VISION SYSTEM – SECURED WEB SERVER

Vision applications are developed using Java spring and HTML5. This means that the Secured Web Server for Vision System can run on multiple platforms and Web Servers Security Mechanism

## 

## ENCRYPTION

All communications will use secured channel using SSL (Secured Socket Layer).

## 

## APPLICATION SECURITY MECHANISM IN VISION

The following mechanisms are implemented in Vision application to enhance security apart from system security.

* User Password are checked against Active directory for the MSB Bank’s users
* User will be automatically deactivated on x number of unsuccessful login attempts based on configuration which can be managed by Vision Administrator.
* User will be informed of last successful login when accessing the system. This will alert the user to unauthorized access, if any.
* Session will turn inactive, if the session is idle. The idle time is cross checked based on the idle-time set for the application
* Same user cannot log more than once; if the user session is already established with application.

## 

## ACCESS RIGHTS BY MODULE / USER

This application security feature fixes the rights for each users of the hosted application at module level. By fixing this access rights, the administrator can restrict or allow users certain modules.

## 

## AUDIT TRAILS

All module level and activity level audit is available.

## 

## AUTO LOGOFF

If the user does not use the application for certain time interval, the session on which the Vision was opened will be logged off automatically. System Administrator through systems settings can parameterize the auto log off.

## 

## SESSION ID & SESSION TRACKING IN VISION

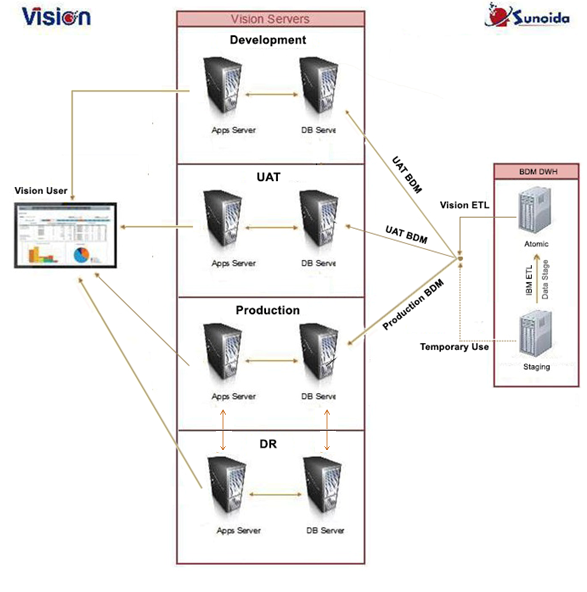
When a user logs into Vision, Vision application generates a session ID. This Session ID is generated using the Java session API. This session ID is added to the URL.

The session ID generated is guaranteed to be unique. Whenever a request is made, either through the GET or POST method, the session ID is appended to the URL. In this way the server is able to keep track of the user in a more secured way. When the user closes the application the session ID and information pertaining to the session ID is removed from the server.

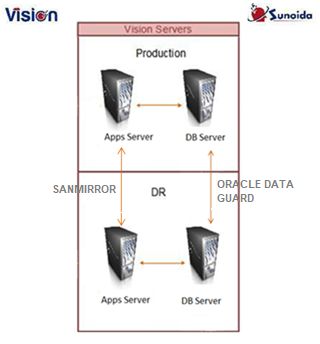
In Vision, when a user logs in, the session ID is stored in a data structure. Every access to the server will be authenticated using this session ID and the Login User ID. If there is any attempt by the user to change either the session ID and the Login User ID, Vision is able to detect and deny any further access to the system.

Apart from this, Vision comes with the Time-Out feature, which automatically terminates a session after a predefined period of inactivity. If the time interval between two requests is larger than the pre-defined period, Vision automatically logs out the user.

* 1. deployment architecture
     1. SYSTEM ARCHITECTURE



* + 1. DATA BACKUP & RECOVERY



Note: Any Patch updates to Application Server between Production & DR to be carried out through SANMIRROR. Vision Application Servers can run in Oracle RAC set-up to balance the load after Go-Live.

* + 1. SERVER SIZING & DATA STORAGE

|  |  |
| --- | --- |
| DATABASE SERVER – HARDWARE / OS – VIRTUAL SERVER SPECIFICATIONS | |
| Production Environment | * **CPU**: 16 Core Processors * **RAM:** 128 GB RAM * **Hard Disk:** 6TB Usable Hard Disk |
| Recovery Environment | * **CPU**: 8 Core Processors * **RAM:** 128 GB RAM * **Hard Disk:** 6TB Usable Hard Disk |
| APPLICATION SERVER – HARDWARE / OS – VIRTUAL SERVER SPECIFICATION | |
| Production Environment | * **CPU:** 16 Core Processors * **RAM:** 64 GB RAM * **Hard Disk:** 500 GB Usable Hard Disk |
| Recovery Environment | * **CPU:** 8 Core Processors * **RAM:** 64 GB RAM * **Hard Disk:** 500 GB Usable Hard Disk |

* + 1. REQUIREMENTS ON SOFTWARE

The requirements on software are summarized below.

|  |  |
| --- | --- |
| Database Server | * Oracle DBEE 12cR2 12.2.0.1 * C Compilers for the Operating System * OS: Linux (64 bit) |
| Application Server | * OS: Linux (64 bit) * Apache Tomcat 8 * Java 1.7 |

* + 1. REQUIREMENTS OF WORKSTATION

The client workstation will provide the gateway to Vision components. In order to successfully meet the implementation approach, workstations must be properly equipped to operate in an environment that will include the use of Vision Suite. The end user will be working with Vision using the web browser.

Windows 7 Professional / Windows 10 Enterprise 32-bit or 64-bit OS with support for all standard support applications (MS Office 2000 & above, Browsers – Google Chrome Version 22.0.1229 & above, Mozilla Firefox Version 25.0.1 & above, Internet Explorer 10 & above, all the Adobe Reader Version).

* + 1. REQUIREMENTS ON MONITORING

Vision extracts data from data sources (detailed under Data Architecture section) on a daily basis and hence provides business critical reports for various users and departments on a daily basis which enables key stakeholders to take timely and accurate policy decisions.

Vision has an internal check feature which alerts business / IT users based on deviations during regular processes like EOD & EOM, when the process aborts. Also, Vision alerts in automation of reports via mail scheduler engine.

On a daily basis, Vision application monitors EOD logs, ETL logs and plug transaction for reconciliation (CBS financials should match Vision financials) as part of regular sanity check.

* 1. constraints
     1. ABILITY OF EXTENSION

The target hardware specification detailed in server sizing above is based on number of customers (2 million), accounts (4 million), branches including transaction offices & sub transaction offices (4000) & transaction volume (3 million per day). Based on last 6 months data utilization (600 GB), Sunoida estimate 6 TB HDD for the period of 5 years. Currently, the total number of users is 200, however, Vision can support up to 1000 users with the above specified RAM and as well as the number of reports.

In future, based on data volume growth and users, the above estimate can be extended.

* + 1. DATA INTEGRITY

Database Patrol is a facility to validate and highlight data errors on the data fed into Vision from the IBM BDM Atomic to ensure the integrity of the information reported from the system.

The DB Patrol process comprises of a number of builds or modules each associated with different tables or set of tables. E.g.: monthly customers, contracts, monthly accounts, monthly financials etc. The DB Patrol builds is a standalone, so that they can be run at any time by the business. However it is important to note that these builds will also form part of the daily and monthly handoff processes.

The user can choose which tables to validate i.e. elect to include/exclude validations on tables as necessary, dependent on when the process is run. For example, some tables are only produced monthly, so would not form part of a DB Patrol that is run, say, in the third week of the month. The DB Patrol processes will populate an error table for each table that has been validated, highlighting all the errors associated with each record in the table. The error tables should be refreshed each time the DB Patrol process is run for the given table being validated. This means that the error tables will always contain only the most up to date validation issues i.e. those that existed at the time the DB Patrol function was run.

It should be noted that the DB Patrol processes are distinct from the initial handoff validations. The validations that form part of the handoff process are performed in order to ensure that the data coming from the source GL can pass the table triggers and therefore be successfully loaded where possible. These validations are processed for the individual country and LE\_Book combinations as the handoff data for these books is received, in order to cater for different timings of data feeds for each set of books from the source GLs. The handoff validations will populate the same error tables as the DB Patrol process, highlighting all the errors associated with each record in the table. The error tables will be refreshed each time the handoff process is run for the given country, LE\_Book and table being validated. Once the handoffs are loaded the DB Patrol processes, can then be run as required by the user. These builds will comprise more stringent validations, than those performed in the handoff and loading process. Most of the DB Patrol builds is incremental.

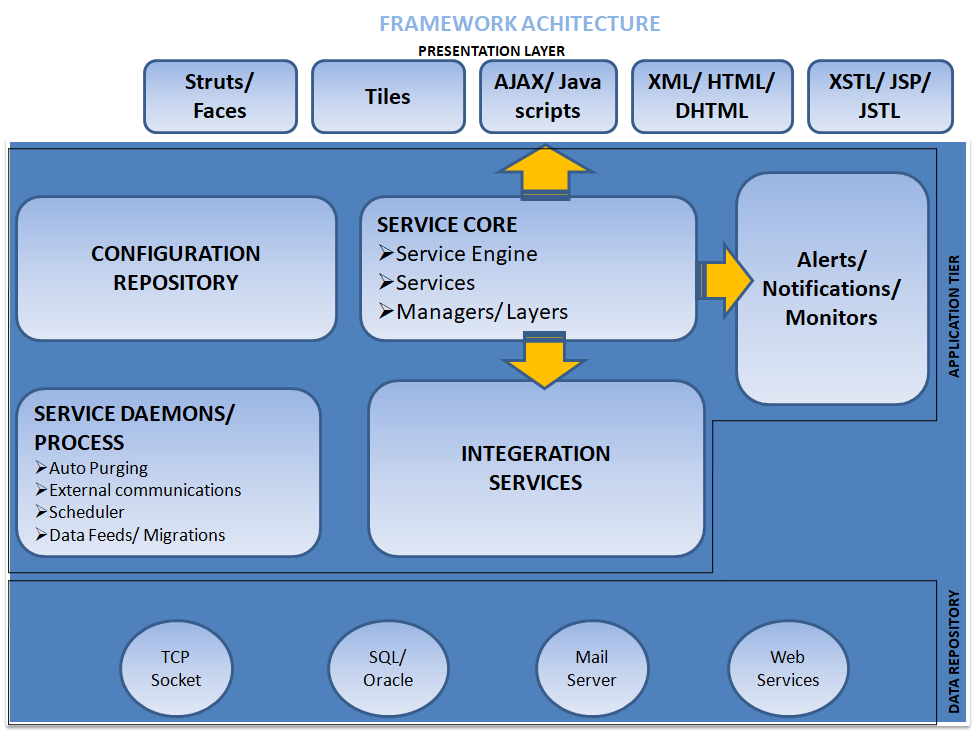
The second component of the DP Patrol function relates to error correction and revalidation processes. There are several ways in which the errors generated by the DB Patrol function can be corrected.

The third part of the DB Patrol function covers the revalidation of the key static tables, such as GL\_Mappings, Customer Enrichment Mappings, Transaction Code Mappings and all those mapping tables associated with the Contracts tables.

The system will assign the “User” to the Maker field and MV: DBP\_Verifier to the “Verifier” field for the records modified by the DBP Screens.

* 1. VISION FRAMEWORK OVERVIEW

The figure below provides a detailed technical overview of each components of the proposed solution framework:



## VISION TECHNOLOGY

Vision proposes an application framework based on n-tier architecture using J2EE set of technologies.

Vision uses Lightweight Directory Access Protocol (LDAP), a directory service protocol that runs on a layer above the TCP/ IP stack. The LDAP directory service is based on a client-server model which is used to access and manage directory information. It reads and edits directories over IP networks and runs directly over TCP/IP using simple string formats for data transfer.

Vision has been built based on the popular Service Oriented Architecture (SOA) complying with J2EE framework. SOA separates three distinct forms of functionality within applications. The main advantage of SOA is the Separation of Business Layer, Presentation Logic and Persistent Layer. Vision is built not alone using this framework, but also using all latest techniques & methodologies available in J2EE Technologies.

Vision is divided into following three major tiers:

1. Presentation Tier

2. Navigation & Workflow Tier

3. Integration Tier a. Middleware Integration b. Database Integration

## 

## PRESENTATION TIER

This tier is responsible to handle the presentation part of the user. Based on the origin of request (be it Web Browser, Auto Client or Web Services), this tier will paint the corresponding screens to the user on the required format. This tier of the product uses following technologies:

Java Spring, Struts & Tiles, JSP, XSLT, XML, CSS, HTML 5, Java Scripts, JPG Images

Vision Application is implemented on Java Virtual Machine (JVM), which analyzes the byte code, interprets the code, and executes it. The compiler compiles the Java file into a Java .class file, then that .class file is input into the JVM, which loads and executes the Vision class file.

JDK is a superset of JRE, and contains everything that is in JRE, plus tools such as the compilers and debuggers necessary for developing Vision applets and applications. JRE provides the libraries, the Java Virtual Machine (JVM) and other components to run applets and application.

## 

## NAVIGATION & WORKFLOW TIER

The tier is responsible to handle the workflow, business logic & navigation rules of the systems. Based on the rules defined, this tier of the product uses various re-usable components available in the Service Engine, Service Manager & processes the response to be provided to the customer. This tier of the product uses following technologies:

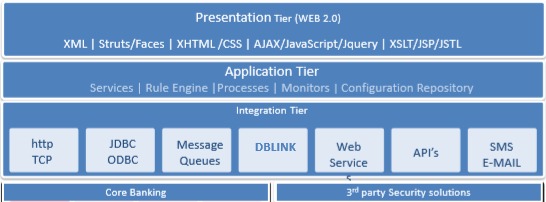
Java Spring, Struts, Java Beans, XML, XSL, DTD, XPATH

## 

## INTEGRATION TIER

This tier is responsible for handling the integration of Vision with IBM BDM (single data source) at the MSB bank. The rules for integration, flow, list of inputs & their formats, list of outputs & their expected formats are defined in this tier. This tier offers very high flexibility to the MSB bank in terms of Data Integrating Vision with their Middleware, Database, Host Systems using different mechanisms, etc. This tier uses following technologies:

JDBC, XSL, XML, XBRL, XSQL, DOM, JSSE, JCE, JDNI, etc.



The benefits of using database links is that they allow users to access another user's objects in a remote database so that they are bounded by the privilege set of the object owner. In other words, a local user can access a link to a remote database without having to be a user on the remote database.

## SALIENT FEATURES

The following section highlights the salient features of the proposed solution:

* Accessible from browser or automated client or web services
* Business rules, process flow, etc are configured as rules in the product source, resulting in minimal implementation
* There are predefined banking metadata fields which are mapped against various dimensions and measures from IBM BDM on which the business rules are applied based on best industry practice reducing Implementation turn-around-time
* Offers a variety of Security features both on the application level & system level such as:

1. Application Level Data Encryption based on Triple DES or SHA
2. System Level Data Encryption based on 128-Bit SSL Encryption
3. Application Level Data Encoding using MD-5 Encoding Mechanism
4. Integration capability with 3rd Party Security Solutions such as SSL

Extensive data validation which includes: Range Validation, Special Characters Validation, Cross-Size Scripting, SQL Injection Validation, Format Validation, Data Subset Validation & Business Logic Validation

* 1. architectural risk

(To Be Updated by Martime Bank Architecture Team).

1. ARCHITECT EVALUATION

(To Be Updated by Martime Bank Architecture Team).

1. APPENDIX

|  |  |  |  |
| --- | --- | --- | --- |
| **GL** | General Ledger | **UAT** | User Acceptance Testing |
| **FRL** | Financial Reporting Lines | **DR** | Disaster Recovery |
| **MRL** | Management Reporting Lines | **CBS** | Core Banking System |
| **OUC** | Organization Unit Codes | **SIBS** | Silverlake Integrated Banking System |
| **SBU** | Strategic Business Unit | **ERP** | Enterprise Resource Planning |
| **LE / LV** | Legal Entity / Legal Vehicle | **AD** | Active Direcory |
| **MIS** | Management Information Systems | **EOD** | End-of-day |
| **FTP** | Fund Transfer Pricing | **EOM** | End-of-month |
| **ETL** | Extract, Transform and Load | **DWH** | Data Warehouse |
| **KPI** | Key Performance Indicator | **LDAP** | Lightweight Directory Access Protocol |
| **FX** | Forex | **TLS** | Transport Layer Security |
| **PDS** | Pre-deployment Study | **DBLINK** | Database Link |
| **HTTP** | Hyper Text Transfer Protocol | **PING** | Packet Internet Groper |
| **SFTP** | Secure File Transfer Protocol | **TELNET** | Telecommunication Network |
| **SSL** | Secured Socket Layer | **J2EE** | Java 2 Exterprise Edition |
| **URL** | Uniform Resource Locator | **JVM** | Java Virtual Machine |
| **CSS** | Cascading Style Sheets | **JRE** | Java Runtime Environment |
| **HTML** | Hyper Text Markup Language | **JDBC** | Java Database Connectivity |
| **API** | Application Programming Interface | **JSSE** | Java Secure Scoket Extention |
| **DOM** | Document Object Model | **JCE** | Java Cryptography Extension |
| **DTD** | Document Type Definition | **JDBI** | Java Database Interface |
| **TCP/IP** | Transfer Control Protocol / Internet Protocol | **XBRL** | Extensible Business Reporting Language |
| **CSRF** | Cross Site Request Forgery | **XSL** | Extensible Style Sheet Language |
| **XSLT** | Extensible Style Sheet Language Template | **XPATH** | XML Path Lanuage |
| **GL** | General Ledger | **UAT** | User Acceptance Testing |
| **FRL** | Financial Reporting Lines | **DR** | Disaster Recovery |
| **MRL** | Management Reporting Lines | **CBS** | Core Banking System |
| **OUC** | Organization Unit Codes | **SIBS** | Silverlake Integrated Banking System |
| **SBU** | Strategic Business Unit | **ERP** | Enterprise Resource Planning |
| **LE / LV** | Legal Entity / Legal Vehicle | **AD** | Active Direcory |
| **MIS** | Management Information Systems | **EOD** | End-of-day |
| **FTP** | Fund Transfer Pricing | **EOM** | End-of-month |
| **ETL** | Extract, Transform and Load | **DWH** | Data Warehouse |
| **KPI** | Key Performance Indicator | **LDAP** | Lightweight Directory Access Protocol |
| **FX** | Forex | **TLS** | Transport Layer Security |
| **PDS** | Pre-deployment Study | **DBLINK** | Database Link |
| **HTTP** | Hyper Text Transfer Protocol | **PING** | Packet Internet Groper |
| **SFTP** | Secure File Transfer Protocol | **TELNET** | Telecommunication Network |
| **SSL** | Secured Socket Layer | **J2EE** | Java 2 Exterprise Edition |
| **URL** | Uniform Resource Locator | **JVM** | Java Virtual Machine |
| **CSS** | Cascading Style Sheets | **JRE** | Java Runtime Environment |
| **HTML** | Hyper Text Markup Language | **JDBC** | Java Database Connectivity |
| **API** | Application Programming Interface | **JSSE** | Java Secure Scoket Extention |
| **DOM** | Document Object Model | **JCE** | Java Cryptography Extension |
| **DTD** | Document Type Definition | **JDBI** | Java Database Interface |
| **TCP/IP** | Transfer Control Protocol / Internet Protocol | **XBRL** | Extensible Business Reporting Language |
| **CSRF** | Cross Site Request Forgery | **XSL** | Extensible Style Sheet Language |
| **XSLT** | Extensible Style Sheet Language Template | **XPATH** | XML Path Lanuage |