Considering retail domain in South Asian region which mainly involves with the majority of the daily lives of the people in the particular region. Because most of the people are not white collars compared to Western regions in other parts of the world. With regard to being the buyer as well as the seller and other related tasks for the same are usually done by same group of people. In developed countries most of the tasks are automated and minimum to no involvement of the people to the retail domain. Scheduled request for purchases when limit reaches predefined values, automated logistic information tracking, self-checkout points at supermarket, etc. This is the exact model or the modern best practices that this region requires to be implemented. This can be achieved by simply bringing the technology in to action.

As per the objectives which needs to be achieved by states, practices modern best practices based on international relations and imposed regulations from the organizations operates in the region as it can be easily distributed among region as low-cost ERP systems are available from best vendors from the world. Oracle Applications are best suit to any type of domain where it customizable to retail even.

Proposed solution is the architecture of Service Orientation where commonly used and easy to implement with minimum knowledge of Software Engineering. Integration architecture implemented in Oracle Application Express (APEX) where core SCM to Financial system uses Oracle E Business Suit. Once it is in operation DevOps road map has been in place for to minimize manual support tasks. Going forward new trends such as Artificial Intelligence can be easily adopted because the DevOps road map expandable in this solution as Oracle Applications are capable of. Meantime tools used as free\open source\subscribed can be developed on-promise based on the Oracle Application availability on the same and researching on future releases of marketplace apps.

There were range of solutions but most effective solution was to use a currently successfully running Point of Sales product integrating to Oracle SCM and Financial modules where most of the Sri Lankan market of Oracle Consultants are SCM and Finance qualified and experience in the same compared to implement the Oracle retail module which is not that much common in the region and cost-wise and other tasks wise feasible recommended by Oracle reginal experts. Other solutions were e-Wis Point of Sales system integrating to Oracle Application using file based approach. This is a straightforward approach but traditional for the current automation disruption era a.k.a. Industry 4.0 where less expandable to automations compared to chosen solution. Main factors considered were resources, cost and popularity. But there can be other factors like robustness for the importance of transactions, transparency as it has to be internationally recognized, and scalability for next era, deprecating traditional approaches to use latest technologies comply with for the ease of support and maintenance can be mainly highlighted. Selecting this solution based not only the suitability but also the previous technical expertise related to core functionalities in areas of wide knowledge of APIs and type of integrations\gateways available in Oracle Business Suit. Meantime the academics are highly assessed make the decision of voting on this solution. Taking about the rationale new technologies like SOA leads traditional approaches because high rate of improvements to the technology is released by the proprietary owners as it is highly growing and more attention of the open source community to other funding organizations’ investments. The challenges faces includes:

* deprecating of the previous APIs
* compatibility of the technology releases to the Oracle E Business Suit patches
* align the data models to process models adhering the formats such as JSON with respect to REST APIs
* data transfer security as it uses HTTP with SSL which can be easily breakdown compared to specific sockets programs
* network failures and identify those as it is occurred
* volume of data load that can be transferred without any corruptions and easily recovered using ERP system functionality

In order to overcome above challenges most effective approach is using Oracle Fusion Cloud applications and tolls where full-edged and which available with more functionalities that are easy to adopt and use in the support and maintenance. Considering the scale of the project and technical and non-technical factors used to make the decision of this implementation of Information System, free and open source tools used.

* Oracle Apiray for documentation and API management
* Running Microsoft Team Foundation Server testing machines\engines with new test cases and practice a good continuous improvements and continuous delivery using build and release tools available
* Oracle APEX REST API definition and mapping tool options available in APEX online IDE
* Network interfaces such as firewall, Windows Server interface
* Oracle Database level PL\SQL code structures to handle data corruptions and recover them

Project inception happened due to recorded discrepancies in company financial accounts where management decision of implementing a vendor specific Point of Sales system with procurement functions integrated to Oracle applications.

There were 13 endpoints according to the documentation which mainly focuses on GL, and Procurement to Payment. Project ran for planned 9 months but incurred delays due to requested modifications to the business process. Main changes were outlet\warehouse purchases has to be done from outlet\warehouse not from back office, document approval hierarchy definition, automate goods receiving, and carry out payments in more effecciently. Apart from that special considerations has to be made due to current affairs in the company like welfare, promotions\loyalty etc and non IT issues like legal, ethical, environmental, health and safety, welfare consideration and risks.

Analysis

Legacy system was observed in couple of times to capture the requirements. Conducted user guide based trainings mainly concentrating on maintenance and development and data conversions. Initiated AIM documentations from requested business process changes, requirements definition for Oracle applications based on current and change of business processes. After that user guides prepared for further analysis to support system design. Back office teams were divided into several groups and work with them to align business processes to Oracle applications where each user stayed till his\her responsible function is described in P2P cycle. Then this function explained with every possible type of using it in Oracle applications. After that user practically involved in according to testing scripts prepared. Finally the management gives the sign off whether this function can be used, requires changes, or replicate it in another system. Meantime each comment is noted down and request My Oracle Support in case of extended help beyond the Oracle EBS documentation.

Design

Based on analyzed business process changes and future of vendor specific ERP system design highlighted use of proposed 13 endpoints which will be integrated Oracle applications whenever there is system improvements continues during next 10 years.

* System design selected document creation interfaces instead of APIs because SOA.
* More weight on modularizing the logic due to frequent requested changes which later will be changed to more structured.
* Application level scheduling not database jobs to use application level to support DevOps roadmap.

For an example implement SOA to 13 endpoints can be done with Integrated SOA Gateway (ISG) shipped with Oracle applications where Oracle APEX services for Web Services temporary used for low level code design. At this point company hosted APEX web services in a cloud environment and it is connected to PBSA POS via LAN\WAN. This separation is quite obvious to use in future to a separate node for standard ISG. Same code design created can be used for ISG custom code or enabling a standard API. Further the same code can be structured into Oracle Fusion cloud where no PL\SQL used.

Development

Extensions and customizations made in Oracle application

* Web ADI
* Custom number sequences
* Customized filtering
* Display third-party information
* Integrated SOA Gateway API creation

Mainly focused not to go for additional OAF pages or Oracle forms at first place but to add new fields to existing and so on. In worst case Oracle APEX mini applications were developed when Web ADIs and other type of same features cannot be used. Then ISG directory is analyzed to choose the best API that can be mapped to procurement, frequent master data conversion (not backlog). Oracle workflow events used to receive and consume payloads from PBSA POS.

Testing

Validating the work in below tests identified based on the system Oracle EBS application architecture with minimum interaction with code changes not going for schema creation for custom data base objects even it is allowed.

* Regression testing – Each code design is as follows:
* Connection establishment
* Getting a new OAuth token
* Retrieve data from specific host
* Send data to specific host
* Persist data in temporary tables
* Validate data in PL\SQL SPs
* Execute APIs or run interface programs
* Integration testing
* Receive and send data rate is validated against expected
* Stress testing
* Working 10 hosts were replicated up to real company scenarios and scheduled the syncing process as designed.

Testing phase validate the work by tech leads and testing were carried out recoded in AIM test scripts. After every module related work is done integration testing is carried out and this is automated in MS Team Foundation. And authorization related tasks were tested related to constrains in Oracle applications configuration in below sub modules.

Constrains and applied on resources

* Using user responsibility
* User management options available
* Employee positions\hierarchies

Risk management

* Integration testing taken into account when communicate and acknowledge the data transfer endpoints specified by analysts. Production network used for this integration is LAN where testing instances currently running and use for future use is WAN connected to internet. Data transfer is to be highly concerned at both of this production and testing purposes before and after the implantation in order to comply with standard data transfer protocols and interchange between the respective instances.
* Because there are many ways of functioning the PBSA POS system. As a result of that the data they send and receive may vary even their documentation defines. PBSA POS documentation for procurement is providing high level functioning tasks and to mitigate the risks we have listed and fixed the steps of using it. Same time Oracle procurement functions has to be restricted to align with receiving and sending data to PBSA POS. And requires attention on redundant functions and information available on both integrated third-party and Oracle applications.
* Compatibility of custom code segments having a high risk when Oracle EBS is merge with Oracle fusion cloud in future estimated in 2030. In order to mitigate the code design is made more structurally rewritable.

QA

* Quality assurance techniques at each stage focuses on workflow, processes, practical guide using screens etc and test data for verification purpose
* Test scripts were designed in LSL and SW systems the interface with back office procurement and finance teams, end users at warehouses\outlets and who are handling minor tasks like excess goods, waste goods, rearranging the packages and related let overs which are not recorded in systems as critical functions.
* When development related extreme programming is used
* Issues to code review even when automated

Implementation

• Dot-matrix printer architecture for direct printing

* SSO logins from MS windows and Limited user licenses
* Data transfer protocols - SSL certificates and Oracle Wallet
* Log management - Papertrail™
* Code quality tools - codacy open source tool
* API documentations
* Code repositories
* Team foundation server\Azure devops server

For an example printer architecture used Oracle Linux printer service “lpr” and it is not compatible with currently using printer where there is no Linux drivers available. Since “lpr” is working as per the requirement of direct printing team and management did not initiated a driver development instead to be more friendly with MS Windows OS a custom batch file is executed to create a temporary file in Windows file system used by prospective users. Then OS where EBS installed namely Oracle Linux 7 accessing the temporary file. This the suggested workaround for MS windows server\MS printer server.

Limited Oracle user licenses were manipulated to map with available windows logins therefore same windows user or that users position\role will be used by Oracle EBS. In case of an audit or any other related investigation, a data dump can be extracted related to this configuration. This is more logical with static IP of PCs using the Oracle EBS.

At this release sprint most of the DevOps functions are integrated to cloud tools available later this can be developed or fully purchased.

Maintenance and devlopment

DevOps roadmap OS as kubernet and docker only with OS with application, networking, application monitoring, resource monitoring, code repository and version control, log management, event\incident\problem management, API documentation and running through code quality tools.

Example 1: Adding credit sales transactions to Oracle EBS. This can be achieved based on the time constraints. As this can be developed in rapid application environment like Oracle APEX or JET, develop a new Oracle Form, adding a new page in OAF using EBS features, use a standard page with page customizations to align with requested module. Suppose this is an APEX application developing to fulfil the requirement and business process. First Oracle APEX is installed in a Docker running Oracle Linux 7. Set network rules to communicate with endpoint(s). Since this is an inbound to the application, design code to consume the payload. Decide the frequency of retrieving data and attach the URI to a data base object using APEX features. Design the user interface to take any actions on the credit sale data. Adding a file base data upload feature in case of high data volume. Next use database links to retrieve temporary credit sales data from Oracle EBS to execute Integrated SOA Gateway (ISG) API for AR invoice and receipt. After ISG API is success the return will clear the temporary data. New changes were tested and application is extracted from docker environment and migrate to production APEX application. Network rules including firewall configurations has to be implemented in production environment. Later when decided no actions to be taken data this APEX application can be replace with Oracle EBS workflow events and consume the endpoint payload to trigger the above ISG API.

Example 2: Adding feature supplier return based on a purchase order – stop current automated processes, insert test data into tables, verify test data with PBSA POS supplier return event in real time, start identifying dependencies and code design, develop the module in a separate dev repository (PL\SQL code, customizations and extracted files), merge the dev to test repository, manually run process related to supplier return, check and verify new changes, move the code to production and update the documentation which is automated with the code level comments, running through code quality, start automated processes, verify the production changes. After the changes were made applications, resources and logs were closely monitored in case of an issue, automated alert will be generated and respective remediation may notified.

Project executed with teams divided into technical experts and domain experts with system functional skills. While running in this phase due to high risks and costs project tasks involved in many demonstrations to training sessions to capture specific requirements align with future system improvements.

Project closure planned in cutover with current master and transaction data conversion, providing operational reports, management reports, and final accounts more accurately.

Non IT issues occurred in legal context when considering country’s governing and financial units and regulations fulfilling to changes in the implementing system. As per current situation this is more straightforward but going back to 10 years back may be a different scenario as a result of civil war visions on liberation and merging two governing units. Legal issues like reginal trending for procurement and sales may affect this system implementation for company’s budgeting purpose which is a crucial sub part of financial accounts which required in initial analysis. Security is the main point emphasized in legal issues.

Ethical issues on distribute knowledge of work to clerical staff where manual system and Oracle EBS both systems are running on. Because file management, document approval, hospitality related tasks needs to aware with slight computer literacy.

Environments where outlets and warehouses located is a main factor to be concerned reaching them and spend special working hours in case of system installation and maintenance.

Restricted or purchasing agreements and GAAP finance literacy highly required for non IT risks mitigation as procurement tasks are depend on the same.

For health and safety server rooms needs to be with good standards with facilitated terminals access locations, and daily use of back office user PCs.

Manual tasks directly affecting like backlog entry due to IT issues and indirectly affecting delegating of responsibilities involved with Oracle EBS can be welfare. Staff in system monitor avoiding tedious tasks which replacing the same manual tasks.

When executing the project methodology used for this system development is more straightforward which is more traditional. It is used in future decisions making to decide improvements to the system. And making decisions on financials and technology made easy and professional issues eliminated and concentrate on people who interface as follows:

At analysis stage fact finding techniques used with customized Oracle reports, gap analysis, user demonstrations, and user trainings. This analysis is done throughout couple of phases. Project budgeting and to decide technologies are done covering financial and technical responsibilities. Different type of professionals mainly IT staff and Accountants made their contributions on this analysis stage.

At design phase more focus on Oracle application architectures to be maintained with minimum customizations following Oracle developer user guidance. The developer efforts will be lower and therefore it is obvious financials planning will not change. System customizations and database designs are done using sqldeveloper tool. And IT staff was key role players in system design especially in network design.

At development phase there were new modules to be licensed and third party tools to be purchased. In order to minimize the costs development of Web ADIs, more complex logics were coded to stop developing new Oracle forms, OAF pages, APEX applications therefore being more technical cut down expenses.

Testing phase is also completed by the developer which is automated with MS Team Foundation and excessive effort made on complete cycle testing. In some cases test cases failed with a new scenario in the procurement to payment cycle and code changes based on Github issues made to end up with successful testing.