International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075, Volume-2, Issue-6, May 2013

Automation of Business Process by Optimization of Data Extraction and Loading

Mohan.G, Madhu.M

Abstract—***An Automobile industry is transitioning to a new electronic enterprise product lifecycle management (PLM) environment using the WindchillPDMLink 10.0 application from Windchill 5.0. The development teams were using software called EIS, which does not have proper workflow. Even though new software exists the users were using the legacy EIS database. Data migration between EIS and WindchillPDMLink 10.0 has to be performed parallel. This involved a mechanism of downloading the data from EIS database and upload into Windchill database. In order to extract the data from EIS Database, SQL Scripting language is used .The data to be retrieved is specified in the script itself and then execution is carried out. The output obtained is saved as .csv file format and when this file gets executed in Windchill server, data gets loaded in the Windchill database. If the output obtained from EIS database exceeds the limit of character in SQL work sheet, then the remaining output gets stored in another .csv file. Complication starts while merging files. This data migration effort required considerably longer time than was expected due to an unexpectedly high percentage of errors in the extracted configurations. Although several iterations of extraction scripts were developed to reduce the number of errors, a significant percentage of configurations continued to display problems.***

Hence to optimize the data extraction methodology as well as to overcome the drawbacks of the EIS data extraction and loading, the system named as "Optimization of Data Extraction and Loading" is designed to improve quality and to reduce time of the data extraction and loading between databases.

Index Terms— Data migration, PLM upgradation, Product Lifecycle Management(PLM), Windchill.

I. INTRODUCTION

The present trend in automobile component manufacturing business emphasizes the need to customize products to the needs of the industry and deliver new products at a shorter lead time than that in the earlier years; in addition, high volume high variety manufacture has become necessary, to ensure growth in business. Hence, it has become necessary for organizations to manage multi-national customers and suppliers, through a multi-company and multi-location organization and company-wide integration of design and manufacturing processes; for this purpose, the business houses, now, has to establish profitable consortia with suppliers and customers.

An automobile industry is the leading manufacturer of auto-electrical products in India, supplying starter motors, alternators, distributors, wipers and headlamps to automobile manufacturing giants.

An automobile industry is a multi-product multi-location company with around 600 varieties of products and around 50 Original Equipment Manufacturing customers and having a strong dealership and service network, of around 700 outlets in India. The company has been certified for QS-9000 systems and has also implemented several JIT initiatives in their manufacturing lines.

The automobile industry Engineering Environment defines a model for an engineering design and communications system that will enhance project collaboration throughout the Enterprise. It will contain a comprehensive set of engineering design and analysis tools with integrated data repositories which are linked together and made accessible through a computer network. This approach facilitates concurrent engineering by enabling the seamless, workflow driven, parallel flow of digital models and information across a secure distributed networking environment with the enforcement of need-to-know user access.

In a global product development environment, more and more companies are implementing distributed engineering processes in order to shorten the time to market and merge different core competences.

An automobile industry currently manages its data for auto electrical parts with two software, EIS and WindchillPDMLink 5.0. EIS has been in place as a data repository since 2000. WindchillPDMLink was selected on the basis of its enhanced capability to manage all forms of product data. WindchillPDMLink, unlike EIS, is fundamentally based upon industrial configuration management principles with workflows and processes which will support SNL's transition to an integrated product lifecycle environment for all product related data.

On a study of several competing PDM systems in the World, which incorporate similar vision in their systems, Windchill offered by M/s Parametric Technology Corporation, was evaluated as the most suitable for An automobile industry. Windchill provides tools required to leverage the Internet/Intranet technologies for product development collaboration, for accelerating the introduction of products to market at lower cost, in addition to facilitating the introduction of a Virtual Enterprise and integrating it with Concurrent Engineering.

An automobile industry is a four decade-old company with standardized systems, Windchill had to be customized to a large extent through the use of multiple business objects to suit the existing and proposed re-engineered process of New Product Development and Change Management.

A WindchillPDMLink 10.0 production system was installed in 2013 January, which began the user transition from EIS to the new PLM solution. The Windchill 5.0 environment was used primarily for design process prove-in (workflows, team roles, access controls) and management of Pro/E MCAD files that had not been managed in EIS. Other functional areas such as non-CAD documents, change management, and product structure were intended to be rolled out in later production phases.

Manuscript received May, 2013.

Mr.G.Mohan, Department of Production Engineering, PSG College of Technology, Peelamedu, Coimbatore-641004, South India.

Asst.Prof.M.Madhu, Department of Production Engineering, PSG College of Technology, Peelamedu, Coimbatore-641004, South India.

Intention of the automobile industry was to migrate relevant legacy data from these two PLM systems into a new WIindchillPDMLink 10.0 system and use this new system to realize the following benefits:

i. Consolidation of data into a single environment.
ii. PLM tool integration with PRO/ENGINEER, the NSE CAD tool of choice.
iii. Establishment of a true enterprise PLM tool.
iv. Support for formal change processes based upon CMII fundamentals.
v. Supports product structure with all product related data; this can potentially replace the company's EBOM system.

II. EARLIER TECHNOLOGY USED IN THE AUTOMOBILE INDUSTRY

In order to extract the data from EIS Database SQL Scripting language is used. The data to be retrieved is specified in the script itself and then execution is carried out. The output obtained is saved as .csv file and when this file gets executed in Windchill server, data gets loaded in the Windchill database. If the output obtained from EIS database exceeds the limit of character in SQL work sheet, then the remaining output gets stored in another .csv file. Complication starts while merging files. If we try to extract data from EIS database and reload the data into Windchill database being unaware that it is already present in Windchill database, the system prompt error only at the loading phase. This data migration effort required considerably longer time than was expected due to an unexpectedly high percentage of errors in the extracted configurations. Although several iterations of extraction scripts were developed to reduce the number of errors, a significant percentage of configurations continued to display problems.

In this paper a new system is generated to optimize the data extraction and loading methodology to reduce time for data migration and cost.

III. RELATED WORK

Modernizing legacy systems is one of the most challenging problems we often face when engineering information systems with new technologies emerging and application domains evolving, legacy systems need to be migrated into new systems at some point, to support enhanced functionality and re-engineered business models. Data migration, as a fundamental aspect of projects on modernizing legacy systems, has been recognized to be a difficult task that may result in failed projects as a whole. Industry survey results reveal that the data migration market is rapidly growing and business companies annually invest billions of dollars in data migration tasks; nevertheless, only 16% of projects successfully accomplish their data migration tasks (i.e., being delivered on time and on budget)— 64% of data migration projects failed to be delivered on time and 37% were over-budget. A main reason for time and budget overrun is the lack of a well-defined methodology that can help handle the complexity of data migration tasks.

In general data migration is the process of moving data from legacy data sources of a legacy system into new data sources of a target system, in which legacy and new systems have different data structures. There are several issues that may considerably complicate this process. First, legacy systems often have a number of heterogeneous data sources designed by using different data modeling tools. This requires a thorough understanding of legacy data sources from various aspects, such as explicit or implicit data constraints, interrelationships across different data sources, and data availability.

Second, legacy systems may have inaccurate, incomplete, duplicate or inconsistent data. On the other side, new systems often require additional semantic constraints on data after being migrated. Thus, bringing the quality of data up to the standard of new systems can be costly and time-consuming. Obstacles in the process of product data exchange among heterogeneous PDM systems include data inconsistencies between senders and receivers, heterogeneity of data, and loss of data quality. These problems are caused by the fact that people who are responsible for creating data generally stick to the former system and are unfamiliar with the methods of data sharing and re-creating data. Thus, in spite of some recent attempts to consolidate distributed product data, only a limited form of data exchange is available.

A project of the German automotive industry called Product Data Technology and Communication in an OEM and Supplier Network (PDTnet) developed a uniform PDM interface based on a neutral data model for PDM data . This PDM interface, called OpenPDM, is a PDM bridge that facilitates the horizontal interfacing of several PDM systems through a Web client, thereby providing users with a single view of product data in a distributed environment. However, OpenPDM is limited to single systems and is not applicable to a virtual view because it is hard to keep track of who has received what data and when; hence, collaboration is difficult. Another significant drawback of OpenPDM is the positioning of the unification in the client, which means that the virtual view can only be used for presentations. There is no chance of applying additional modules of business logic to the unified view. Nowachi and Lukas proposed a PDM federation interface in which a higher degree of flexibility can be achieved in the integration of product data by approaching the side of the server. Through the introduction of a single virtual PDM system, the concept of a PDM federation provides a method of receiving client queries and processing them in several concrete PDM systems. This approach conceals the internal complexity of the federation and offers a uniform view and access to the data. Furthermore, the virtual PDM layer does not replicate the product data but is responsible for distributing queries and collating partial results to produce an overall result based on the distribution model. However, given that a uniform data model for applying the virtual PDM system is under development, there are still a number of problems in the general use of the commercial PDM system. Stouffs et al. provides a framework for representing product data based on a standardized syntax for the data exchange. It defines primitive data types that can be combined using formal compositional operators to form more complex data types. The resulting canonical representation allows comparing, mapping, and translating different product models. Morris et al. described a case study and solution of an IBM research project (called Hedwig) to investigate creating robust solutions for PLM. They focused on several research issues, including information federation, data mapping, synchronization, and web services connections. They described a working system that allows access to several heterogeneous PDM systems that are used in the automotive

and aerospace industries. Besides researching the integration of several PDM systems, researchers are examining approaches to the exchange of product data between two PDM systems by means of the interface called Standard for the Exchange of Product Model Data (STEP, ISO 10303). Goellnitz et al. proposed a method of integrating PTC's WINDCHILL with Dassault Systems' ENOVIA VPM through a PDM backbone that includes a STEP PDM processor. To exchange product data between SIEMENS Teamcenter and PTC's WINDCHILL, the US Army developed, for both PDM systems, an interface adaptor that supports the STEP application protocol (AP) termed AP214 CC6 (where CC refers to conformance classes).

IV. DATA MIGRATION

Migration projects are initiated due to a number of technical and non-technical reasons. Among the technical motivations for migrating a legacy system to a modern platform is the obsolescence of the technology or programming language in use.

Even though new software exists the users were using the legacy EIS database. Due to the upgradation of WindchillPDMLink 10.0, Data migration between EIS and Windchill10.0 has to be performed parallel.

The actual cost of migrating the data for these projects is frequently much greater than was initially anticipated. Large data migration projects often fall into the vicious cycle of "Code, Load & Explode". The symptoms of Code, Load & Explode will sound familiar to most implementation teams: develop programming logic - run a test load – the load fails - discover that data assumptions were incorrect and unexpected data quality issues exist – react by cleansing data and changing specifications – delay to adjust programming – re-test the load – the load fails - find additional incorrect data assumptions and additional unexpected data quality problems . . . react again . . . delay again . . . react again. . . delay again . . .over and over.

In all too many data migration projects, the vicious cycle of Code, Load & Explode goes on, and on, and on**,** increasing the cost beyond anyone's worst imagination.

V. THE THREE PRIMARY PROBLEMS IN DATA MIGRATION

A. Lack of Data Knowledge

Data migration projects frequently fail due to a lack of understanding of the data in the legacy system. Documentation for legacy systems is often incomplete, out of date, or missing entirely. Assumed relationships of data are often inaccurate. Most organizations assume that they understand the structure and content of the data in their systems, so they feel the data migration project will be a trivial part of the new system implementation. Unfortunately, most organizations don't learn that their assumptions are incorrect until much effort and expense has been wasted.

B. Data Quality Problems

Data quality problems in legacy systems are often the cause of major delays and cost overruns in data migration projects. All too often data migration projects must be placed on "hold" while unanticipated data cleansing projects are completed. In many cases data quality problems are not discovered until the new system is failing.

C. Lack of Flexibility for Specification Changes

The inability to quickly react to specification changes is the cause of major delays and cost overruns in many data migration projects. It is estimated that as much as 90% of the specifications initially provided for data migration projects change significantly during the life of the project; and over 25% of the specifications will change more than once (in many cases several times) before the project is complete. Further, many of the changes don't occur until the late stages of the data migration project, just prior to the "go live" date. The inability to react to changes quickly and efficiently can jeopardize the entire implementation project.

Without the ability to solve these "Three Primary Problems", data migration projects will continue to experience the syndrome of Code, Load & Explode.

VI. DATA EXTRACTION AND FORMAT REQUIREMENTS

Requirements analysis is the first stage in the systems engineering process and software development process. Requirements analysis is critical to the success of a development project. Requirements must be actionable, measurable, testable, related to identified business needs or opportunities, and defined to a level of detail sufficient for System design. Requirements can be functional and non-functional.

Data extraction and format requirements of Automobile industry are mentioned below:

i. A consolidated dump of all the data files will be provided before migration. This is very important for any Data migration projects. It will reduce the problem of data loss in the legacy systems.
ii. All data files to be provided shall be in readable text or excel file format. Every data files is converted into .csv file.
iii. The nomenclature for all attributes in WINDCHILL shall be decided before migration.
iv. All calculated and derived attributes shall be identified before migration.
v. The data files shall be separated as Model and Part data files and shall be named accordingly.
vi. The data files in excel format shall have the item # as the first column and all the corresponding attributes in the following columns.
vii. The data files extracted from EIS shall be without duplicate records. Earlier technology used in automobile industry will create duplicate records while merging of files.
viii. If linkages or product structure has to be built then all required information like parent (assembly) part and child (constituent) parts information shall be provided explicitly in the data files.
ix. No documents (CAD and NON CAD) shall be migrated as part of this migration.

A Study was conducted based on the requirement of the automobile industry. It shows that based on the user entry of data extraction data duplication was occurred. A System with minimal manual intervention was needed to reduce data migration period from Legacy Database.

VII. USER INTERFACE

An important aspect of system development is a viable user interface. It is necessary to achieve a productive relationship with the user as devote to the requirements of the system. Hence a user interface requirement is important for the system development. Most research indicates that as the number of user increases, the probability of system failure increases without close analysis / user interface.

The user interface requirements concerned to reduce manual intervention are as follows:

i. HTML makes the screen more vivid and navigation is made easy.
ii. It has to provide user with meaningful and user- friendly screens.
iii. It has to display only that information that is relevant to the current context.
iv. It has to produce meaningful error message.
v. Minimize the number of input actions required by the user.
vi. Easier to print report and the software can handle any number of records.

VIII. CONNECTING TO THE DATABASE

The first thing to establish a connection with the database involves, two steps :

i. Loading the driver: This involves loading of JDBC-ODBC bridge driver.
ii. Making the connection: This involves connection of appropriate database to the database.The connection returned by the method 'Driver Manager. Get Connection' is an open connection used to create JDBC statements that pass SQL statements to the database.

IX. OBJECT ORIENTED ANALYSIS AND DESIGN APPROACH

In order to develop a software, Object Oriented Analysis (OOA) was done. OOA helps designers identify the detailed requirements of their software, rather than how the software should be structured or implemented. It aims to describe the existing system and how it operates, and how the software system should interact with it.

Use case analysis:

Use case analysis is done to capture the functional requirements of the system by describing the different ways in which an actor interacts with the system.

Use case diagram for the current scenario is given in the figure (1).

This use case involves two actors namely:

1. Administrator and
2. PLM users.

Administrator: They are the ones, who is going to extract the data from the database and load the data into the database. They are the responsible for maintaining the data, which is stored in the database. They can view the data that is extracted from the database and the data to be loaded into database.

PLM User: They are the one, who is going to view the data.

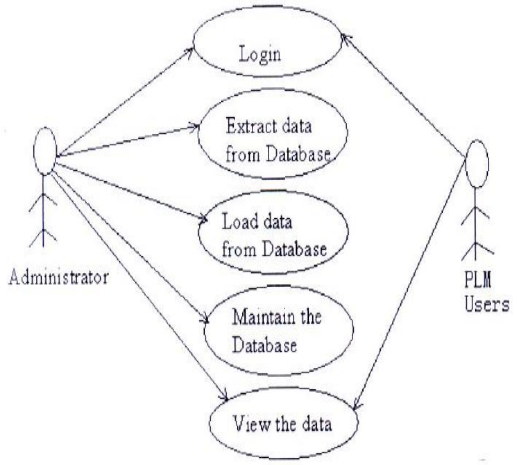


Fig 1 :Use case diagram

Class Diagrams can be viewed from three perspectives:

1. Conceptual model: The diagram represents the concepts in the domain under study. In the analysis stage, this model is drawn.
2. Specification model: Here we are looking at the interfaces of the software. When working with the software, this model is focused on.
3. Implementation model: When illustrating a particular implementation technique, use this perspective.

Class diagrams, illustrating relationships between classes for this use case realization is shown in the figure(2).

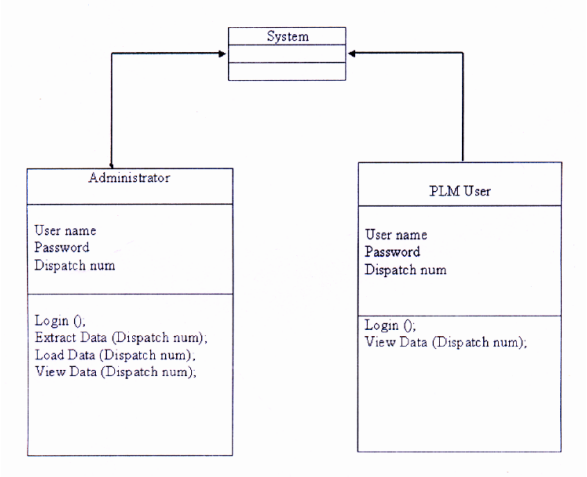


Fig 2: Class diagram

The function used in this paper is clearly identified in class diagram with attributes. The function used in the class diagram shown in figure (2) is an example only. Apart from these functions there are many function was used to develop the system. Without the class diagram the development of system will take long time to finish.

Sequence diagram:

Next step in the developing system is formulation of sequence diagram.

A sequence diagram is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use

case realizations in the Logical View of the system under development.

Interaction between user with EIS Database and WINDCHILL Database is clearly defined in sequence diagram.

These are horizontal arrows with the message name written above them. Solid arrows with full heads are synchronous calls, solid arrows with stick heads are asynchronous calls and dashed arrows with stick heads are return messages.

User interface developed must have Login page for authentication purpose. PLM User and Administrator was available throughout the project time.

Sequence diagram for data migration work is shown in the figure (3).

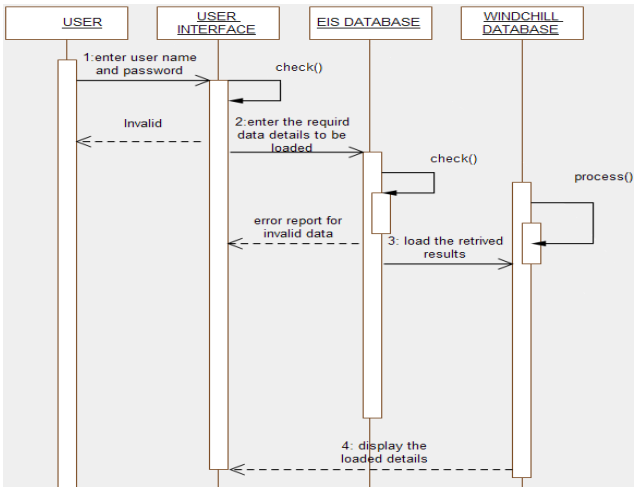


Fig 3: Sequence diagram

Logical Data design :

The logical data design describes the inputs, outputs, databases and procedures. It reviews the current physical system and its data flow content, volumes, frequencies etc., The logical data design for data migration work is shown in the figure (4).

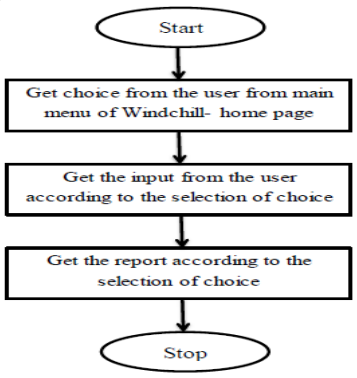


Fig 4: Logical data design

X. WINDCHILL CUSTOMIZATION

Customization of WINDCHILL is required to fetch the data from EIS database using java server pages. WINDCHILL is a three-tier JAVA Runtime Architecture that can be customized in many ways to meet our customization requirements in addition to the existing Out Of The Box (OOTB) functionality. There are basically two ways to customize WINDCHILL based on the level of complexity of customization.

1. Lightweight Customization
2. Heavyweight Customization

In our project Heavyweight Customization was done to add a new user interface in WINDCHILL software to migrate the data from EIS database.

Heavyweight customization involves use of client and server technologies require extensive WINDCHILL API coding.

XI. PROPOSED METHODOLOGY

In order to optimize the data extraction methodology as well as to overcome the drawbacks of the EIS data extraction and loading, the system named as "Optimization of Data Extraction and Loading" is designed to improve quality and to reduce time of the data extraction and loading between databases.

Proposed methodology is shown in the figure (5).

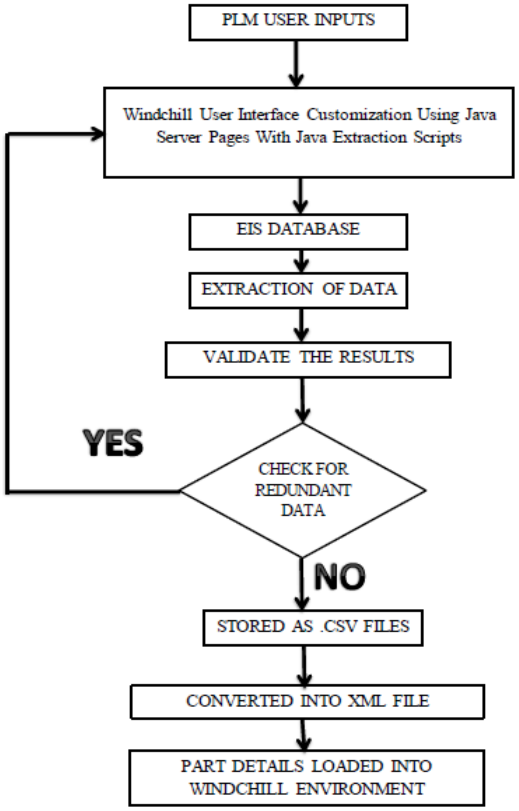


Fig 5: Proposed methodology

New user interface which is known as heavy weight customization was created in WINDCHILL PDM LINK 10.0 to get the user entry for data extraction with JAVA extraction scripts to extract the data from the EIS database.

User has to select which data to be migrated using the customized user interface.

XII. DATA MIGRATION REHEARSALS

A rigorous testing program, performed before moving data from legacy system to a target database, will ensure that the final migration process will be well understood and predictable. Failure to perform a rigorous testing program is risky and may lead to unpredictable results. Therefore, as much testing as possible should be

completed before migrating to a target database. Such preparation testing must include the following types of tests: migration, minimal, functionality, integration, performance, and stress.

After migrating a test database, it should be used to ensure that our existing applications operate properly with the target database.

Data from each system is first extracted using custom JAVA extraction scripts and then loaded into the Pro/INTRALINK system where it is validated to ensure no errors or loss of data. If significant errors or missing data files are identified, the source of the problems are identified, the data extraction scripts are modified, and another data extraction is performed.

Once data is validated to a level deemed acceptable, the data is loaded from Pro/INTRALINK into the PDMLink 10.0 PLM system using a WINDCHILL migratory tool that takes the Pro/INTRALINK data and maps data types, attributes, states, versions and other characteristics to the PDMLink 10.0 equivalents.

A series of rehearsals were designed to test and refine the custom JAVA scripts that are used to extract data from the legacy PLM systems. The process for the data extraction and validation used were as follows:

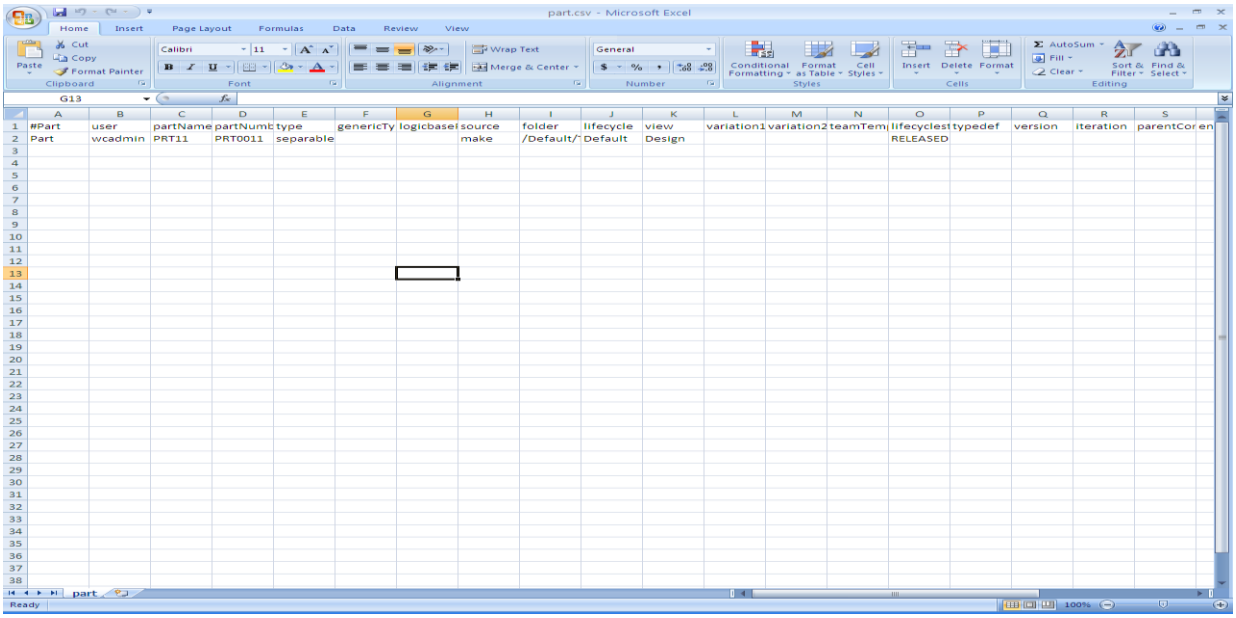
- Develop a data extraction script.
- Perform a data extraction rehearsal.
- Validate extracted data using PTC's ModelCHECK application. ModelCHECK provided Useful information regarding model quality and data integrity, but results were not used to refine extraction scripts.

At least 3 full rehearsals to be planned in order to ensure that by the time the last rehearsal is executed, the process runs unhindered and also fits within the defined and targeted Go-Live window.

When the determined number of rehearsals has not given the confidence on the process and is not meeting the go-live window, it is better to invest in another rehearsal than risking executing the Production Cut-over.

On an average, nearly 60-70% of the effort for Migrating is taken up by the Rehearsal 1 as the most number of issues are uncovered during the same and need to be resolved.

The extracted results were stored as .csv file as shown below. The details of CSV files are shown in the figure (6).



#Part	user	partName	partNumb	type	genericTy	logicbasef	source	folder	lifecycle	view	variation1	variation2	teamTem	lifecycles	typedef	version	iteration	parentCon
Part	wcadmin	PRT11	PRT0011	separable			make	/Default/	Default	Design				RELEASED				

Fig 6: Extracted csv file from EIS database.

Comma-separated values (CSV) file stores tabular data (numbers and text) in plain-text form. Plain text means that the file is a sequence of characters, with no data that has to be interpreted instead, as binary numbers.

CSV file is a common, relatively simple file format that is widely supported by consumer, business, and scientific applications. Among its most common uses is moving tabular data between programs that natively operate on incompatible (often proprietary and/or undocumented) formats. This works because so many programs support some variation of CSV at least as an alternative import/export format.

For example, a user may need to transfer information from a database program that stores data in a proprietary format, to a spreadsheet that uses a completely different format. The database program most likely can export its data as "CSV"; the exported CSV file can then be imported by the spreadsheet Program.

XIII. CONVERTING CSV FILES TO XML FORMAT FILES

Before converting the CSV files, the CSV file is updated to match the definitions in the csvmapfile.txt file. The files must contain all of the data required (for example, containers) before they are converted with the CSV2XML utility.

WINDCHILL environment consists of a set of containers that hold all of the administrative areas (known as domains), rules, and data that make up the context from which WINDCHILL users work.

Data created using the CSV file format which is then used with the conversion utility to convert it into XML file, as it is easier to create the data and less error-prone in the CSV format.

Conversion of CSV to XML is done using WINDCHILL utility. The converted xml file is given in the figure (7).

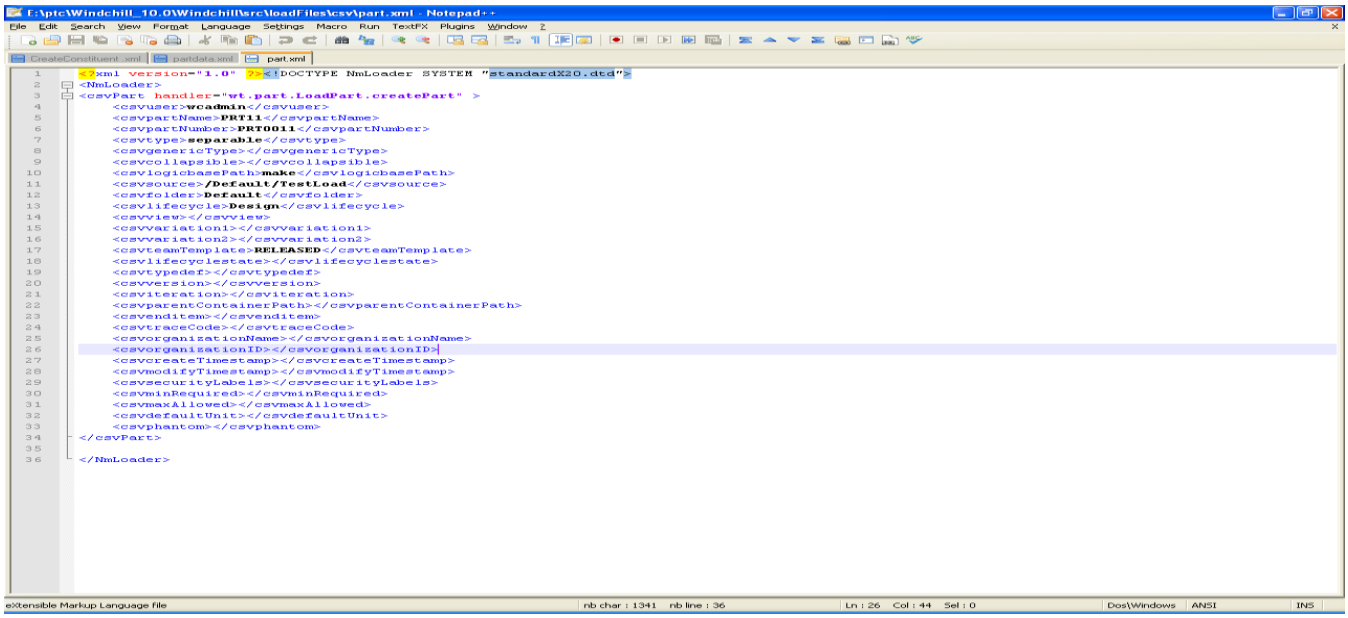


Fig 7: XML file of extracted data from EIS database.

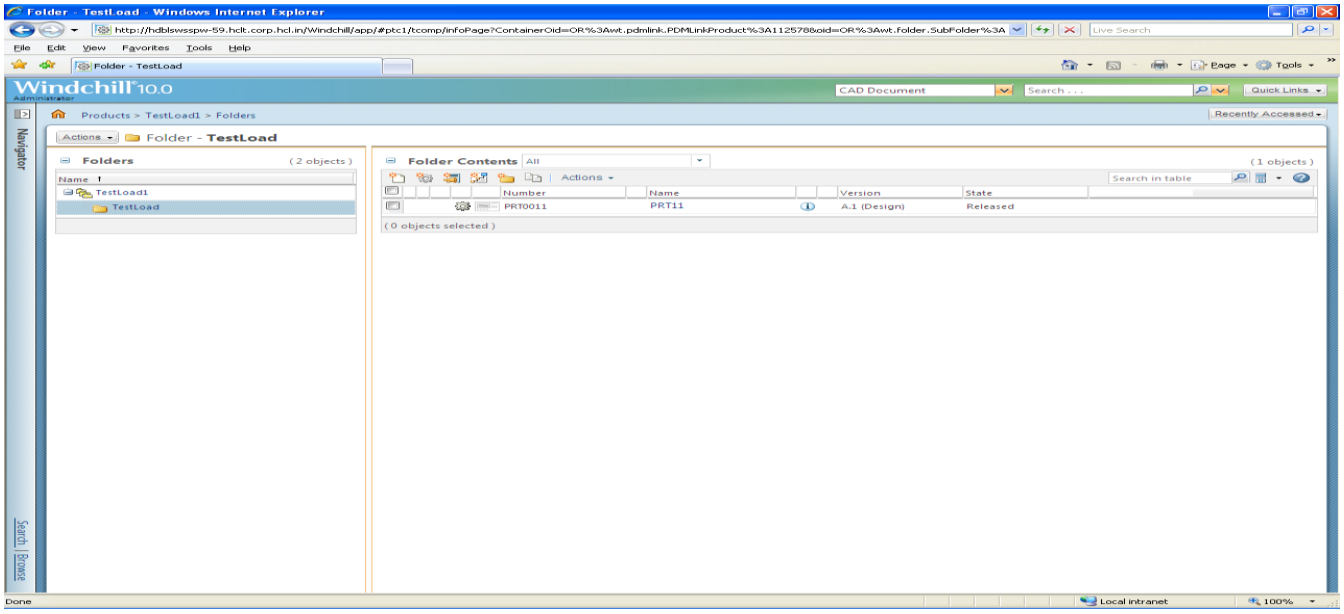


Fig 8: Loaded part in WindchillPDMlLink10.0.

XIV. LOAD XML FILE INTO WINDCHILLPDMLINK 10.0

Converted XML file is then loaded in WINDCHILL environment using Windchill shell. The loaded part in WINDCHILL is shown in the figure (8). The details of the data of the part which is migrated from EIS database to WINDHILL is shown in figure (9).

The data loaded consists of existing part detais. The same method is followed to add assembly part details. In assembly parts, the child parts which is loaded before is added as constituent parts .

The existing parts added to assembly parts is checked for redundancy. The assembly part details loaded in WINCHILL is checked after it is loaded in WINDCHILLPDMLINK 10.0

Automobile industry was using EIS software for more than 12 years. The database to be migrated consists of large number of data. The above step is followed to migrate the data from EIS database into WINDCHILLPDMLINK 10.0 database.

A comparision of Previous method of data migration is done with the newly customized system . The newly optimised system of data extraction and loading reduces the calculated time for previous method of data migration to 55 %

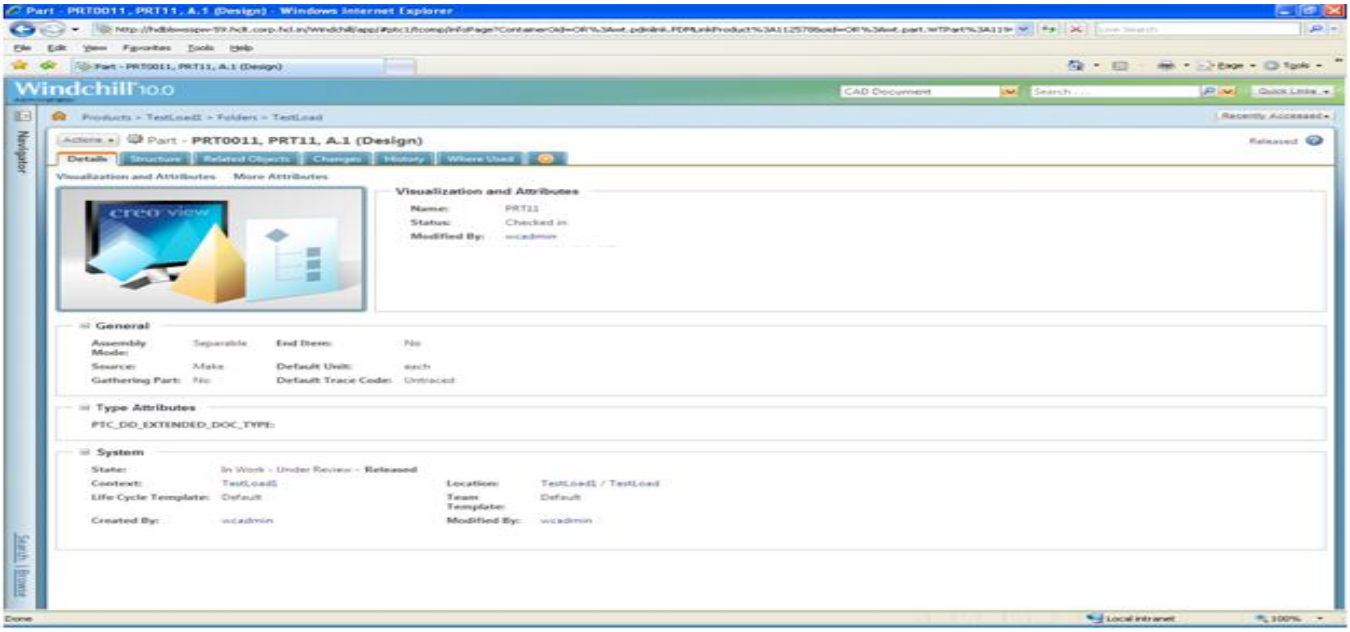


Fig 9:Details of Loaded part in WindchillPDMLink10.0.

XV. CONCLUSION

We have customized the WINDCHILLPDMLINK 10.0 user interface to get the inputs from users and to generate meaningful error on wrong inputs from users. Data from each system is first extracted using custom JAVA extraction scripts and then loaded into the Pro/INTRALINK system where it is validated to ensure no errors or loss of data. If significant errors or missing data files are identified, the source of the problems are identified, the data extraction scripts are modified, and another data extraction is performed.

The extracted data without error is converted into CSV file and then converted into XML file using utilities in WINDCHILL. The XML file is then loaded into WINDCHILLPDMLINK 10.0. The assembly data also created by adding existing part data as its constituent part.

Likewise, we have migrated more than 12 years of data from EIS database into WINDCHILLPDMLINK 10.0 with minimal manual intervention.

The newly optimised system of data extraction and loading reduces the calculated time for previous method of data migration to 55 %.

Thus Data extraction and loading process from legacy system is optimized with minimum manual intervention.

REFERENCES

[1] Mariano Ceccato, Thomas Roy Dean, Paolo Tonella, "Recovering structured data types from a legacy data model with overlays , Science direct Information and Software Technology 51 (2009) 1454–1468, 10 May 2009.

[2] J. Bisbal, D. Lawless, B. Wu, J. Grimson, "Legacy information systems": issues and directions, Software, IEEE 16 (5) (1999) 103–111, doi:10.1109/52.795108.

[3] Rapid Application Development (RAD) for Data Migration — White Paper Solutions by Premier International, Applaud_White_Paper.pdf 2004.

[4] A Roadmap to Data Migration Success-White Paper by SAP.

[5] M. Aboulsamh, J. Davies, Specification and verification of model-driven data migration, Model and Data Engineering (2011) 214–225.

[6] A.V. Aho, J.D. Ullman, Universality of data retrieval languages, in: Proceedings of Principles of Programming Languages, ACM, 1979, pp. 110–119.

[7] Farouk Belkadi, Nadège Troussier, Collaboration based on product lifecycles interoperability for extended enterprise, Int J Interact Des Manuf (2010).

[8] Hongjun Lu and Ling Feng , Integrating database and world wide web technologies , Baltzer Science Publishers , (1998) 73–86.

[9] Jesus Bisbal and Jane grimson, "Database sampling with functional dependencies", science direct Information and software technology, 43(2001) 607-615.

[10] C. Ou-Yang and M. C. Cheng,"Developing a PDM/MRP integration framework to evaluate the influence of engineering change on inventory scrap cost", Int J Adv Manuf Technol (2003)22:161–174 DOI 10.1007/s00170-002-1454-4.

[11] Hong-Taek Ju, Mi-Joung Choi and James W. Hong, "EWS-Based Management Application Interface and Integration Mechanisms for Web-Based Element Management", *Journal of Network and Systems Management, Vol. 9, No. 1, 2001*1064-7570/ 01/ 0300-0031.

[12] Gregory Follen , Chan Kim, Isaac Lopez , Janche Sang And Scott Townsend, "A CORBA-Based Distributed Component Environment for Wrapping and Coupling Legacy Scientific Codes" , Kluwer Academic Publishers. Manufactured in The Netherlands, Cluster Computing 5, 277–285, 2002.

[13] Andres S. Orrego · Gregory E. Mundy, A study of software reuse in NASA legacy systems Innovations System Software Engineering (2007) 3:167–180 DOI 10.1007/s11334-007-0027.

[14] Takahiro Hara, Kaname Harumoto,Masahiko Tsukamoto, Member, and Shojiro Nishio, "Database Migration:A New Architecture for Transaction Processing in Broadband Networks", IEEE transactions on knowledge and data engineering, vol. 10, no. 5, september/october 1998 839.

[15] Katarina Grolinger AND Miriam A.M. Capretz, "A unit test approach for database schema evolution" ,Science direct Information and Software Technology 53 (2011) 159–170.

Mr.G.Mohan, scholar of M.E. Product Design and Commerce, Department of Production Engineering, PSG College of Technology, Peelamedu, Coimbatore-641004, South India.

Asst. Prof. M.Madhu, Production Engineering Department, PSG College of Technology, Peelamedu, Coimbatore-641004, South India.