Legacy Systems Migration - A Method and its Tool-kit Framework

Praveen Shaik and Prof Y.P.Singh
Sunrise University,Alwar,Rajasthan

Abstract—**problems posed by mission-critical legacy systems - brittleness, inflexibility, isolation, non-extensibility, lack of openness etc. - are well known, but practical solutions have been slow to emerge. Each system has its peculiarities and uniqueness.. This paper presents an approach to mission-critical legacy system migration : the Butterfly Methodology, its data migration engine and supporting tool-kit framework. Data migration is the primary focus of the Butterfly methodology,. The fundamental premise of the Butterfly methodology is to question the need for parallel operation of the legacy and target systems during migration. Much of the complexity of the current migration methodologies is eliminated by removing this interoperation assumption.**

Key Word: (Data-Access Allocator (DAA , Data Migration, intropolation, Butterfly Methodology , Screen Scraping, SI System Interface UI User Interface M Application Module Legacy Data / Database Service Legacy Data / Database Service Target Data Target Data Legacy System Understanding)

I. Introduction

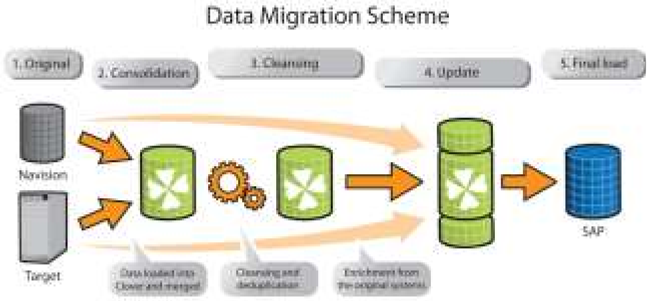
THE system engineering process normally involves phases of requirement definition, system design, subsystem development, system integration, installation, evolution and finally system decommissioning [22]. Within this process, system evolution plays, during the whole lifetime of the developed systems, a very important role in maintaining system performance and enhancing it to meet new requirements. The widespread use of computer technology over several decades has resulted in some large, complex systems which have evolved to a state where they significantly resist further modification and evolution. Such systems are termed Legacy Systems [3]. These systems typically form the backbone of information flow within an organisation and are normally mission-critical : if one of these systems stops working the business will generally grind to a halt. Thus for many organisations, decommissioning these systems is not an option [22]. An alternative solution is legacy system migration which has become an important research and practical issue, for both the system engineering and database areas ([3], [22], [11] and [27]).

Legacy system migration is concerned with developing a target system which retains the functionality and, importantly, data of the original legacy system [3] but which can be easily maintained and adapted to meet future business requirements. There is an urgent need to provide a strategy which will allow the migration of the systems to new platforms and architectures and to provide tools and methodologies to support such a strategy.

II. Related work Legacy system migration

Encompasses many research areas. A single migration project could, quite legitimately, address the areas of reverse engineering, business reengineering, schema mapping and translation, data transformation, application development, human computer-interaction and testing. Due to space limitations, only a brief outline of research directly related to legacy system migration will be presented. Current practical solutions mainly adopt what is referred to as “Wrapping”. Wrapping involves surrounding existing data, individual programs, application systems, and interfaces to give a legacy system a ‘new and improved’ look or improve operations [26]. The most popular implementation of this is “Screen Scraping”. Screen Scraping is the process of replacing the character based front ends of legacy systems with a client based graphical user interface. Introducing a graphical interface does not address many of the serious problems posed by legacy systems. At best it reduces training costs for new employees and allows an interface to the legacy system be placed on the desktop.This is unacceptable when considering migration of mission-critical legacy systems. Ganti and Brayman [11] propose general guidelines for migrating legacy systems to a distributed environment. Using these guidelines, the business is first examined and the business processes found are reengineered as required. Legacy information systems are linked with these processes to determine which systems have data and business logic of value in the new target environment. A set of processes are selected and the associated legacy systems are analysed. New applications are then developed to fit these processes. These guidelines are not really suitable for use in migrating a missioncritical legacy system. Only the data and its structure are used in the decentralised target system. Mention is made of retaining logic encoded in applications but it appears that the legacy systems will be discarded and replaced with new applications.

This approach recognises that legacy system migration should cause as little disruption to the current business environment as possible. However, it is unclear how the cut-over to the new, separately developed, target system will be handled. In their Chicken Little Methodology Brodie and Stonebraker ([2], [3]) propose an 11 step generic migration strategy employing complex gateways, shown in Figure 1.



(Figure-1 Data Migration Scheme)

In this method the legacy and target information systems are operated in parallel throughout the migration. Initially the target information system is very small, perhaps only one application with a very small database. However as the migration progresses the target system will grow in size until it performs all the functionality of the legacy system which can then be retired. During the migration, the legacy and target information systems interoperate to form the operational mission-critical information system.

Step 1 : Incrementally analyse the legacy information system

Step 2 : Incrementally decompose the legacy information system structure

Step 3 : Incrementally design the target interfaces

Step 4 : Incrementally design the target applications

Step 5 : Incrementally design the target database

Step 6 : Incrementally install the target environment

Step 7 : Incrementally create and install the necessary gateways

Step 8 : Incrementally migrate the legacy databases

Step 9 : Incrementally migrate the legacy applications

Step 10 : Incrementally migrate the legacy interfaces

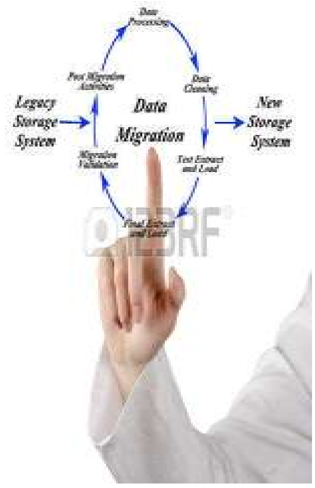
Step 11 : Incrementally cut over to the target information system.

III. The Butterfly methodology

The Butterfly methodology is being developed as part of the MILESTONE project, a collaborative project involving Trinity College Dublin, Broadcom Éireann Research, Telecom Éireann, and Ericssons. MILESTONE aims to provide a migration methodology and a generic supporting tool-kit for the methodology to aid migration engineers in the process of migrating legacy information systems to target systems. The project began in July, 1996 and will finish in June, 1998. A trial legacy system migration following the Butterfly methodology is currently being planed and results will be available in the future. The objective of the Butterfly methodology is to migrate a mission-critical legacy system to a target system. Different from Chicken little, the Butterfly methodology eliminates, during the migration, the need for system users to simultaneously access both the legacy and target systems, and therefore, eliminates the need of interoperation between these two (heterogeneous) information systems. It is very important to bear in mind that, using the Butterfly methodology, the target system will not be in production while the legacy system is being migrated. The legacy system will remain in full production during the whole migration process. There will never be a case where live data is stored, at the same time, in both the legacy and target systems.

3.1 Overview Legacy system migration:can be a very expensive procedure which carries a definite risk of failure. In order to perform a successful migration, a sound model of the migration process is obviously needed. Currently, however, no general model exists. MILESTONE's considers that migration consists of five major tasks:

1) Justification; 2) Legacy System Understanding; 3) Target System Development; 4) Migration 5) Testing.



(Figure-2 Data Migration Cycle)

Within each task, general software/system engineering techniques can be applied. Testing plays a very important role in all tasks. Justification involves an intensive study to quantify the risk and benefits and fully justify the redevelopment of the legacy system involved. Legacy System Understanding mainly involves reverse engineering of the legacy system.

- Phase 0: Prepare for migration. Once the decision to migrate a legacy system has been made, the next stage is to prepare everything for the migration. Although many issues essential to a migration project must be clarified at this stage, for example management, organizational issues or budget, the Butterfly methodology focuses only on the technical issues. Among these issues the Butterfly methodology considers user requirements for migration and target system determination to be most important. The main activities within this phase are listed in Figure 5. Obviously these activities can only succeed through intensive co-operation among the legacy system experts, migration engineers and users.

• Phase 1: Understand the semantics of the legacy system and develop the target data schema(s). The activities identified within this "reverse engineering"

IV. Research Methodology

It consists the followings phasis.

Phase 0: Prepare for migration

0.1 Getting the migration preliminary requirements;

0.1.1Determining user requirements;

0.1.2 Determining benchmarks for measurement of migration success; 0.2 Determining the target architecture;

0.3 Preparing the target hardware system;

Phase 1: Understand the semantics of the legacy system and develop the target data schema(s)

1.1 Understanding the legacy interfaces, identifying redundancies and determining the function of the target interfaces;

1.2 Understanding the legacy applications, identifying redundancies and determining the function of the target applications;

1.3 Understanding the legacy data; identifying redundancies and determining to-be-migrated data; (optional)

1.4 Identifying and understanding interactions with other systems;

1.5 Finalising the migration requirements;

1.6 Developing the Data-Access-Allocator (DAA);

1.7 Developing the target data schemas and determining mapping rules. Figure 5 Migration Activities in Phase 0 Figure 6 Migration Activities in Phases.

Phase 0: Prepare for migration.

Phase 1: Understand the semantics of the legacy system and develop the target data schema(s).

Phase 2: Build up a Sample Datastore, based upon the Target SampleData, in the target system

Phase 3: migrate all the components (except for data) of the legacy system to the target architecture.

Phase 4: Gradually migrate the legacy data into the target system and train users in target system

Phase 5: Cut-over to the completed target system. Figure 4 Six Phases of the Butterfly methodology

V. The Butterfly methodology

The Butterfly methodology utline supporting tool-kit Research into software tools has been ongoing for decades. Numerous tools have already been developed to assist in many stages of migration and research is still ongoing ([1], [4], [5], [6], [7], [12], [15], [17], [20], [21], [28]). However no single tool can completely automate any single phase, let alone migration as a whole. The aim of MILESTONE is not to develop an integrated tool-kit for migration as a whole. Due to time and resource limitations.

5.1 Project support As for any large systems development project, migration using the Butterfly methodology will require the support of tools to manage the project schedule and resources. A conventional project management tool (e.g. [10], [14], [20]) may be used to support the management of the migration. A project repository will be used to store information needed to support tools used, and people involved, in all phases of migration.

5.2 Justification tools The requirement for justification tools in migration is also similar to that for any systems development project. It is however more crucial that the risks and benefits of a migration be clearly understood as a failed migration project can result in an unusable legacy and an unusable target system. Obviously estimation of risk and benefit involves a degree of intuition based on experience, and knowledge of the individual organisation, and cannot be completely automated. However, once a plan has been formulated, cost and schedule uncertainty can be estimated using a risk analysis tool (e.g. [5], [6], [20]).

5.3 Legacy system understanding tools Legacy system understanding is a core part of migration. Information retrieved or produced in this phase affects tasks in all other phases. Tasks to be performed include reconstructing system documentation, identifying and extracting the legacy data schema, identifying reusable components and redundancies and investigating how the legacy system interacts with other systems and resources. Much research has focussed on this area and numerous tools exist to assist in this process, although many may not be sufficiently mature for use in real migration efforts (e.g. [1], [4], [7], [12], [15], [17], [21], [28]).

5.4 Target system development tools Any appropriate system development tools can be used to support the target system development. Numerous tools exist to assist in this task (e.g. [8], [13], [18], [25], [24]). There are also some tasks specific only to migration, such as schema mapping, for which tool support would be invaluable. Mapping tools will therefore be used to build the new target data schema from the legacy data schema (e.g. [7], [9], [16], [19

5.5 Testing tools: Testing is an essential part of migration using the Butterfly methodology and is an ongoing process in all phases. Apart from the general testing environment needed for any system development, some particular requirements exist for migration testing. One important aspect of migration testing is to ensure that there are no unexpected inconsistencies between the critical functionality of the legacy system and its replacement. Unfortunately, to date, no specific migration testing environment exists. It is beyond the main focus of MILESTONE to develop such an environment. However, because it is important to use "legacy" data to test the target system i.e. actual data which will be used in the target system, an application/tool for sample data generation will be introduced in the MILESTONE toolkit.

5.6 Migration tools The critical part of a migration project is the cut over from the legacy system to the target system. An important migration requirement is to cause as little disruption to the business environment as possible. The most essential and difficult migration process is that of migrating the mission-critical legacy data. Using the Butterfly methodology, two tools are used to control this process : Data Access Allocator (DAA) and a Data Transformation Tool (Chrysaliser). These tools will be application specific and may have to individually constructed for each migration effort.

VI. CONCLUSIONS AND FUTURE WORK

Presently MILESTONE's approach to the problem of legacy system migration has been presented. The migration process as a whole is a very complex procedure encompassing many different fields of research. The focus of discussion was thus necessarily limited. The proposed Butterfly methodology applies to the whole process of legacy system migration with the main focus specifically on the migration of legacy data in a missioncritical environment. The Butterfly methodology is a simple, safe, and open approach to this problem. It represents a departure from current thinking on how legacy systems as a whole can be migrated to new architectures.

The main difference between the Butterfly methodology and other proposed migration methodologies is that the Butterfly methodology is a gateway-free approach. It eliminates, during migration, the need for system users to simultaneously access both the legacy and target systems.

REFERENCES

[1] Bachmann, 'A CASE for Reverse Engineering', Datamation, pp. 49-56, July 1988

[2] M. Brodie and M. Stonebraker, 'DARWIN: On the Incremental Migration of Legacy Information Systems', Technical Report TR-022-10-92-165 GTE Labs Inc., http://info.gte.com/ftp/doc/tech-reports/tech-reports.html, March 1993

[3] M. Brodie and M. Stonebraker, 'Migrating Legacy Systems: Gateways, Interfaces and the Incremental Approach', Morgan Kaufmann 1995

[4] D. N. Chin and A. Quilici, 'DECODE: A Co-operative Program Understanding Environment', Journal of Software Maintenance 8(1), pp. 3-34, 1996.

[5] Concept Sales Ltd., 'Dependency Modelling Tool (DMT)', http://www.bucks.net/concept/dmthome.html, May 1997

[6] C/S Solutions Ltd., 'Risk+', http://cssolution.com/riskov. htm, May 1997

[7] Computer Science Department - The University of Namur, 'DB-MAIN: A R&D Programme in Database Applications Engineering and Case Technology', http://www.info.fundp .ac.be/~dbm/, February 1996

[8] Dynamics Research Coporation, 'VisualMagic', http://www.visualmagic.com, May 1997

[9] Evolutionary Technologies International, 'ETI-EXTRACT Tool Suite', http://www.evtch.com, May 1997

[10] Fresnelsoft, 'ReFind', http://www.fresnelsoft.com, May 1997

[11] N. Ganti& W. Brayman, 'Transition of Legacy Systems to a Distributed Architecture', John Wiley, 1995

[12] Z-Y Liu, M. Ballantyne and L. Seward, 'An Assistant for Re-Engineering Legacy Systems', Proc. 6th Innovative Applications of Artificial Intelligence Conf. pp 95-102, Seattle, WA, http://www.spo.eds.com:80/eds/papers/ asstreeng.html, August 1994

[13] Magna Software Corporation, 'Magna X Application Generator', http://www.magna.com, May 1997

[14] Microsoft Corporation, 'Microsoft Project', http://www.microsoft.com/project, May 1997

[15] Dr. H. A. Muller, 'Understanding Software Systems Using Reverse Engineering Technologies Research & Practice', Tutorial presented at Int. Conf. on Software Engineering 18, http://tara.uvic.ca/UVicRevTut/UVicRevTut.html, March 1996

[16] B. Narasimhan, S. B. Navathe and S. Jayaraman, 'On Mapping ER and Relational models into OO schema', Proc. 12th Int'l Conf. on the entity-relationship Approach ER'93 (LNCS 823) pp. 402-413

[17] J. Q. Ning, A. Engberts and W. Kozaczynski, 'Automatic Support for Legacy Code Understanding', In Communications of the ACM, 37(5), pp. 50-57, May 1994

[18] Object Domain Systems, 'Object Domain Object Oriented Analysis and Design Tool', http://www.object-domain. com, May 1997

[19] Ontos Inc., 'OIS Development Tools', http://www.ontos.com, May 1997

[20] Quantitative Software Management(QSM), 'SLIM (Software Life-Cycle Model)',http://www.qsm.com, May 1997

[21] Dr. H. Muller, 'RIGI Project - An Extensible System for Retargetable Reverse Engineering', University of Victoria, Canada, http://tara.uvic.ca, November 1996

[22] I. Sommerville, Software Engineering Environments, 5thEd., Addison-Wesley, 1995.

[23] F. N. Springsteel, 'Object-based schema integration for heterogeneous databases : a logical approach', Proc. 4th Int'l Conf. of database and expert systems Applications DEXA 93 (LNCS 720) pp. 166-180

[24] Structured Technology Group, 'AxiomSys, AxiomDsn', http://www.stgcase.com, May 1997

[25] S. Stobart, 'CASE Tool Home Page', http://www.osiris.sunderland.ac.uk/sst/casehome.html, May 1997

[26] Systems Techniques Inc., 'Wrapping Legacy Systems for Reuse : Repackaging vs. Rebuilding', http://www.systecinc.com/white/wplist.html

[27] S. R. Tilley and D. B. Smith, 'Perspectives on Legacy System Reengineering', http://www.sei.cmu.edu/~reengineering/lsyree

[28] K. Wong, S. Tilley, H. Muller, M. Storey, 'Structural Redocumentation: A Case Study', IEEE Software, pp. 46-53, January 1995.

About the Author:

1. Mrs Praveen Begum. is a Research scholar and He is also pursuing Ph.D. Degree from the Sun Rise University, Alwar,Rajasthan.. She has published 3-4 Research papers on the the legacy Technology .She also attended a handful of conferences . She is persuing the Research on the fellowships from Maulana Azad Education Foundation , New-Delhi.

2. Prof. Y.P. Singh, Professor in Electronics & Communication Engineering, Sunrise University, He has worked as Director, Somany (P.G.) Institute of Technology and Management , Rewari, Haryana . He has also worked about 29 years as Lecturer, Dean of academics & Principal in many Engineering institutions and organization. He has also served with Training and Technical Deptt. Govt.Of Delhi, almost for 17 years. He has about 162 research paper published in National and 364 papers published in international journals in his credit. He has been selected and awarded by Govt. of Delhi as "Best Technical Teacher-2004". He has been conferred Outstanding Teacher Award 2012 and 2013 respectively.

His Research work has been adjudged as outstanding work. He has submitted his 2 patents and are in process. He has also has been awarded as "Star of Asia Award " an international award , by Global achievers foundation. He has been awarded the Best Technical Director (Research) in Haryana in 2014, and Best Technical Director (Research) in Delhi 2015 . He is also an expert and Master Trainer for the Teachers, empanelled by SCERT/NCERT. He is also the guide of research scholar for almost dozen of Universities. Beside He is the foundation member of LI-Fi constorium in India.