# Industrial Perception of Legacy Software System and their Modernization

*B. V. Batlajery*

*R. Khadka*

*A. M. Saeidi*

*S. Jansen*

*J. Hage*

Technical Report UU-CS-2014-004
Feburary 2014

Department of Information and Computing Sciences
Utrecht University, Utrecht, The Netherlands
www.cs.uu.nl

ISSN: 0924-3275

Department of Information and Computing Sciences
Utrecht University
P.O. Box 80.089
3508 TB Utrecht
The Netherlands

# Industrial Perception of Legacy Software Systems and Their Modernization

Belfrit V. Batlajery, Ravi Khadka, Amir M. Saeidi, Slinger Jansen, and Jurrian Hage
Utrecht University, Utrecht, The Netherlands
{b.v.batlajery, r.khadka, a.m.saeidi, slinger.jansen, j.hage}@uu.nl

Abstract

*Research in legacy system modernization has traditionally focused on its technical challenges, and takes the standpoint that legacy systems are obsolete systems, yet they are crucial for an organization's operation. Nonetheless, it remains unclear whether practitioners in the industry also share this perception. This paper describes the outcome of an exploratory study in which 26 practitioners from the industry are interviewed on what makes a software system a legacy system, what the main drivers are leading to the modernization of such systems, and what challenges are faced during the modernization process. The findings of the interviews have been validated by means of a survey with 198 respondents. As it turns out, practitioners value their legacy systems highly, the challenges they face are not just technical, but also include business and organizational aspects.*

## 1. Introduction

After three decades of legacy modernization research, it is surprising to find that legacy systems are still in daily operation. A legacy system is any software system that significantly resists modification but are business critical, and hence, their failure can have serious impact on the business [1]. Most of these systems were developed years ago, and they are continually evolving. With changing requirements frequent, modifications within these legacy systems have resulted in unstructured source code, which is difficult and costly to maintain. Furthermore, the knowledge about those legacy systems are scarce as the original programmers leave the company or retire, and up-to-date documentation is usually lacking [2]. These issues have been recognized by the software engineering community and a plethora of legacy modernization approaches have been proposed, primarily, to reduce maintenance costs, and to increase flexibility. Despite the issues of legacy systems and acclaimed benefits of legacy modernization, technology consulting firms estimate that 180-200 billion lines of legacy code are still in active use for daily business [2]. This discrepancy has triggered us to investigate legacy systems and their modernization from an industrial perspective. We aim at identifying what are good characteristics of legacy systems, what bad characteristics of legacy systems are the drivers for legacy modernization and what challenges are faced in industry while modernizing legacy systems.

We set up our research as an explorative study with an aim at discovering new perspectives and insights about legacy systems in industry, instead of starting out with predefined hypotheses and testing them. Therefore, we used grounded theory [3], which is increasingly popular in software engineering research [4], to conduct and analyze semi-structured interviews with 26 practitioners from industry with an experience in legacy systems and legacy modernization. The findings of the interviews were validated through a separate structured survey with 198 respondents.

This report has the following contributions:

1. We document the industrial perception of legacy systems and their modernization.
2. We identify the perceived benefits of the legacy systems, drivers of modernization, and challenges that the industry faces during modernization.
3. We report the perceptional differences of and about legacy systems between the industry and academia.

The paper is structured as follows. In Section 2, we present the related work; in Section 3, the research method is detailed, in particular the GT approach; in Section 4, research design is explained; in Section 5, the research execution is presented; in Section 6 the research findings are explained; in Section 7, the validation using the results of the survey is presented with some statistical details used. Finally, the report concludes in Section 9.

## 2. Related Work

Brodie & Stonebraker [5] in their book describe legacy systems as ``*any systems that cannot be modified to adapt to constantly changing business requirements and their failure can have a serious impact on business*." and reported various characteristics of the legacy systems such as mission critical, hard to maintain, inflexible and brittle. In [6], Bennett described legacy system as ``large software system that we don't know how to cope to cope with but that are vital to our organization". Bisbal et al. [1] enlisted several problems of legacy systems such as legacy systems run on obsolete hardware, expensive to maintain, lack of documentation and understanding of system, difficult to extend and integrate with other systems. From the aforementioned definitions and characteristics of legacy systems, it is evident that legacy systems are presented as a serious problem in academia. The current study, in contrast, takes different approach to identify the characteristics that still keep them operational in industry. Nevertheless, the study also explores various issues of the legacy systems that complement the issues identified in the academia. In fact, we explicitly identified overlaps between the issues of legacy systems in academia and in industry such as high maintenance cost, lack of resources, achieve flexibility. In this research, we have presented those issues as drivers for legacy modernization.

Within 2 decades, a plethora of legacy modernization approaches have been reported. We refer to systematic literature studies [7-9] for further details. With respect to legacy modernization challenges, Brodie [10] enlisted various technical challenges of legacy modernization that are more influenced by academic research experiences. Van Deursen et al. [11] presented an overview of techniques to facilitate legacy modernization and the issues of modernization, particularly aiming at identifying objects. The legacy modernization research and the challenges identified in the academia are largely technology oriented. They provide different techniques/methods to facilitate legacy modernization and point out various challenges faced in the course of applying those techniques/methods. Our research not only identified various business issues, but also confirmed the technical observations made by researchers.

## 3. Research Method

The objective of this research is to explore how legacy system is perceived in industry and revisit legacy modernization from industrial perspective. The result will be compared with academia perspective. Two research methods are applied in this research, being interview and survey. By doing interview, the author can understand the meaning or nature of experience of persons with legacy systems and legacy modernization. It helps the author to find out what practitioners think about legacy systems and how do they perform legacy modernization. On the other hand, survey is used as a triangulation method to validate the findings derived from interview process.

The overall research model is depicted in Figure 1.

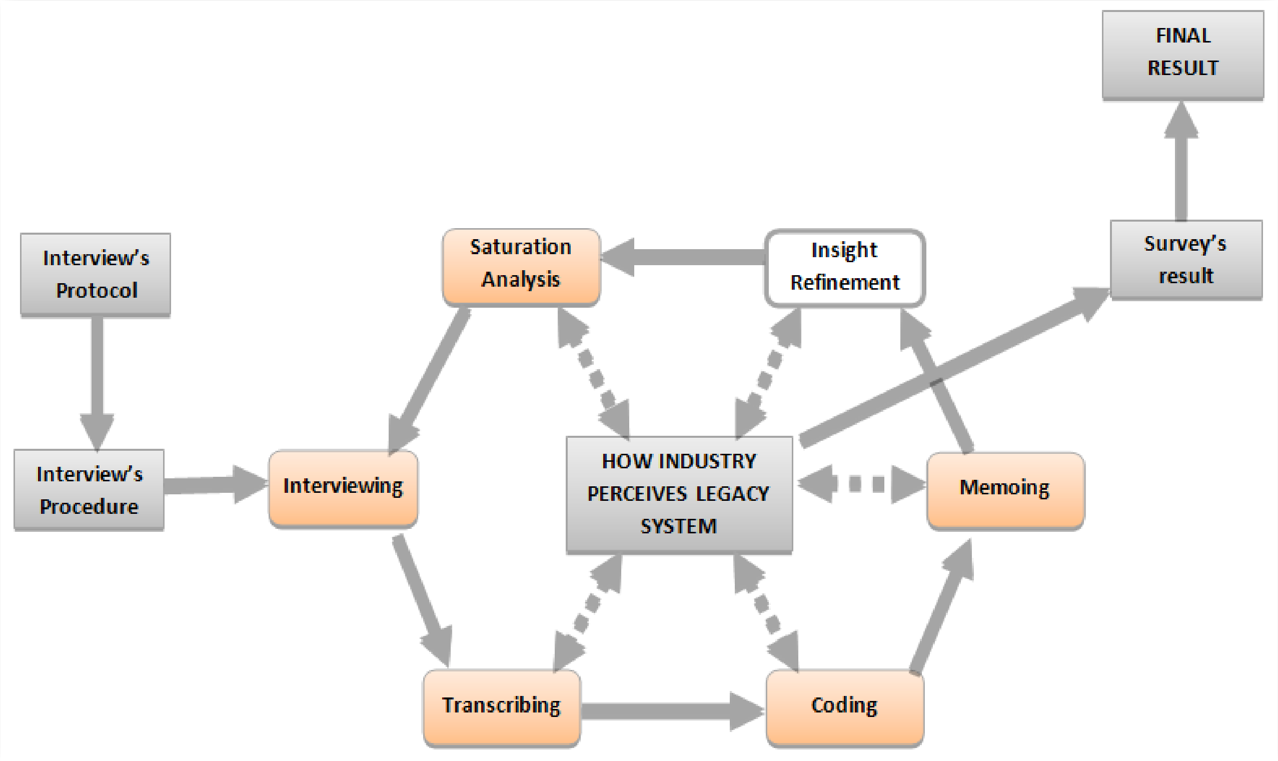


Figure 1. The research model

In this research, we aim at exploring new perspectives and insights about legacy systems. Thus, this research is adopts exploratory research using Grounded Theory (GT) as qualitative research through the process of interview. The interview is performed to collect opinions or impressions about legacy systems and legacy modernization from professional in industry. The form of interview session is semi-structured interview. We construct multiple questions based on his knowledge, literature study, survey´s fact, and input from experts in order to get better quality of information from the participants.

Prior to conducting interviews, an interview protocol is developed that provides information about the interview. The interview protocol is then sent to all the interviewees beforehand. In the interview protocol, the research motivation is introduced briefly and then question regarding the information regarding the interviewee is stated. Some representative questions about legacy system and legacy modernization are also stated in the protocol.

To apply Grounded Theory method, data received from interview were processed into several steps: (i) transcribing, (ii) coding, (iii) memoing, (iv) insight refinement, and (v) saturation analysis. By using Grounded theory, we try to get the new perspective and insight by exploring the data received from interview to grow his insight.

First step in GT is conducting the interview and record it. The interviews were performed in face-to-face meeting (direct meeting) at the office of participants or in the some places where both participant (interviewee) and researcher (interviewer) agree with. The interviewer(s) and the interviewee(s) will have a conversation in English for about 60-90 minutes depends on information flowing between them.

After the interview is performed, the recorded voice was transcribed word by word. The transcription then was cut into smaller units (in sentences or paragraphs) and was categorized based on key characteristic. The process is the called coding. The coding processes are not only the process of labeling, but also the process of leading the raw data to the idea and link them together. From coding the author can also identify the pattern such as similarity, difference, frequency, sequence, correspondence, or causation. The codes are organized hierarchically into concepts and at the end will be grouped into categories.

After the process of coding, the process of memoing is performed. Memoing is the process of writing down narratives explaining the ideas of the evolving theory. We transcribed the recorded interviews and read the transcribed documents carefully on what our participants said and try to understand what does it means. After that we try to present our idea on why does that matter in our research. The objective of memoing is to demonstrate to the author himself, provide his interpretation, express his position, and explain his decision during the process of research. Upon reaching the saturation stage- a condition that there is no additional information is being found with new interviews.

We used an online survey to triangulate the finding of the interview, to which 198 responded. The objective of the survey is to solicit information about the legacy systems and the legacy modernization in order to validate the finding derived from the interview sessions.

Our survey contains 17 questions that capture the demographic background of the participants, questions about legacy systems and also about legacy modernization. However, there are some logics in the questions to make the survey simple. These logics will allow respondents to skip the questions which are not applicable to their experience. All of them can be completed in approximately 10 minutes and was designed accordingly.

## 4. Research Design

In total, there were 23 interviews with 26 participants who can provide relevant information on legacy systems and legacy modernization from various organizations across the Netherlands. We started the current research with a series of interviews conducted with 26 practitioners (identified as P1-P26 in this paper), each lasting 1-2 hours. The informants were selected based on two criteria: (i) have experience with legacy systems, and (ii) have experience with legacy modernization projects. The informants were identified opportunistically, via industrial collaborators, followed by snowball samplingkitchenham2002principles, in which the first generation informants help to identify other informants fulfilling the criteria. In total, 23 interview sessions were performed. In three of the interview sessions, the interview was conducted with two informants from the same organization, and additionally, two practitioners were from a same company, reducing the sample size of the participating organizations to 22. The sample is arguably broad to represent the professional software engineering, including diversity among the domain of the industries and variations among the informant's roles and experiences. Table 1 depicts the details of the informants with the domain of their company. The company sizes ranged from consulting firms to global corporations such as IBM, Deloitte, Capgemini. The variation among the informant's roles is also broad, ranging from software developers, to system analysts, consultants, software architects, business architects, research and development managers, and Chief Information Officers (CIOs). The experiences of the informants range from 5 years to 43 years, with >19 years as an average experience of the sample and cumulatively, the informants have >490 years of experiences in information technology. Additionally, the data totaled 25 hours of recorded data.

Table 1. Details of the informants

| Participant | Type of industry | Position | Experiences |
|---|---|---|---|
| P1 | Banking | Business Architect | 24 |
| P2 | Information technology and Services | Application Innovation services/ Bus App modernization | 33 |
| P3 | Airlines / Aviation | Project Manager | 18 |
| P4 ,P5 | Government | ICT Manager and Information Architect | 12 and 15 |
| P6 | Consultancy | SOA/ Integration Consultant | 5 |
| P7,P8 | Software house | Manager Technology & Quality, and Manager Research and Development | 35 and 15 |
| P9 | Security and investigation | Manager Research and Development | 10 |
| P10 | Management consulting | Management Consultant | 43 |
| P11 | Information technology and Services | Chief Technology Officer Benelux | 27 |
| P12 | Information technology and Services | Software Architect / Consultant | 15 |
| P13 | Food and dairy | Head of Office of the CIO | 17 |
| P14 | Poultry | Director of IT | 20 |
| P15 | Insurance | CIO | 24 |
| P16 | Flower auction | Manager I&A Service, Infrastructure and Applications | 10 |
| P17 | Banking | Business Information Manager | 5 |
| P18 | Software house | ICT & Logistics Consultant | 13 |
| P19 | Software house | CTO | 15 |
| P20,P21 | Information technology and Services | Manager Business & Product Development and Service architect | 5 and 15 |
| P22 | Insurance | Director Operations | 36 |
| P23 | Machinery | Manager Besturingen (Control) | 18 |
| P24 | Consultancy | Senior manager | 23 |
| P25 | Government Administration | IT Architect Middleware and Integration | 10 |
| P26 | Government Administration | Lead architect | 15 |

We conducted semi-structured interviews in person in English; these were recorded. Prior to the interview session, informants were provided with an interview protocol with sample questions to be discussed in the interview sessions. The interview session consists of three categories of questions (i) about characteristics of legacy systems, (ii) drivers for legacy modernization, and (iii) challenges faced during modernization. Afterwards, the recorded interviews were transcribed and each interview transcript was analyzed through coding: a process of breaking up the interviews into smaller coherent units, and adding codes to these units. Subsequently, a process of writing down narratives that explain the ideas of the evolving theory, known as memoing, was used to develop the coding. These coherent units represent key characteristics of the interview being analyzed. Later, the codes were organized into concepts, which in turn were grouped into categories. When the interviews being analyzed progressively provided answers similar to the earlier ones, a saturation stage was observed. To confirm the saturation stage, we conducted two more interviews and found that the analysis resulted in similar responses to the earlier ones. We used Nvivo 10[1] as an instrumentation tool to facilitate the interview analysis process.

The second and final phase of this research aimed at validating the results. We adopted a structured survey as a data triangulation process to validate the findings of the interview results. A data triangulation process, using more than one data source or collecting the same data at different occasions, is typically used to increase/decrease confidence in a finding by confirming or contradicting evidence and helps to improve

[1] www.qsrinternational.com/

validity of the findings of an empirical study such as this. The survey[2] was announced via mailing lists, social media such as Twitter, LinkedIn, Facebook, and via personal referrals. Listing 1 depicts the groups that we used in Linked in to spread our survey.

Listing 1. LinkedIn groups used for distributing survey.

- IEEE-SCAM
- Legacy migration
- Legacy modernization
- TOGAF
- Mainframe Migration
- Amadeus alumni
- Shell Australia IT Network
- Software Improvement Group
- TCS - Tata Consultancy Services
- Application Modernization and APM
- Enterprise Legacy Modernization (ELM)
- Mainframe Experts Network
- Cognizant Technology Solutions
- Legacy AS400 I-series IBM i Modernization
- Centrum Wiskunde & Informatica - Employees & Alumni of CWI and Mathematisch Centrum
- Master Business Informatics - Utrecht University
- IBM Enterprise Modernization Professionals
- Shell Information Technology International

## 5. Research Execution

In the following section, we detail the execution of the GT method that we have used in this study. Following the GT process, we started with transcribing the recorded interview and then iteratively started coding. These coding were then grouped into subcategories and categories. In total we have 3 categories comprising of 45 codes. In the following section, we provide the detail of the resulting codes with short description.

### 5.1 Resulting collections of codes

The process of coding was executed after the first interview has been transcribed. Quotes and codes are built up freely during the time depends on the comments given by participants in their interview sessions. The coding system in this research comprises 3 top-level categories, 25 subordinate concepts and 1-5 basic codes per concepts giving a total 44 codes. Each code has its own name and short description for further explanation about the code. The details of the research codes are presented in Appendix A and the respective code with quotes of the interviewees are presented in Appendix B.

## 6. Research Finding

The primary findings included the followings:

### 6.1 Legacy systems

The interview sessions usually started by asking a definition of a legacy system from the informants. Most of the informants agreed that legacy systems are “old” systems. Despite being old systems, the informants pointed out that legacy systems are “core” systems that have been proven to work correctly in a production environment for decades. P1 provided his opinion about “old” and “core” system as: “Most of the legacy systems are older than 20-30 years...[] Most of the systems of the legacy systems are the core system.“ P11 agreed with P1 by stating: “It [Legacy system] is an old system; ... a lot of legacy system is the core system.”

Interestingly, most of the informants related legacy systems as systems which that do not ?t with the future IT strategy of the organization. P19 expressed this as: “My definition of a legacy system is systems and technologies that do not belong to your strategic technology goals.“ To investigate further, the informants were asked if programming language is a determining factor for a system being legacy, we obtained a mixed opinion. More than half of the informants do not agree that the programming language is a determining factor for a system to be legacy, while the rest were in agreement. Such a

[2] http://servicifi.wordpress.com/legacy-survey/

mixed opinion is also observed from the results of the survey. Around 50% of the respondents agreed that there programming languages do determine if a system is legacy.

6.2 Perceived benefits of legacy systems
The perceived benefits of the legacy systems included (i) business critical, (ii) proven technology, (iii) reliable system, and (iv) performance.

6.3 Drivers for modernization
The drivers of the modernization included (i)high maintenance costs, (ii) lack of knowledge, (iii) to remain agile to change, and (iv) prone to failures.

6.4 Challenges of the modernization
The following challenges were identified: (i) Complex system architecture, (ii) Difficult to extract and prioritize business logic, (iii) lack of knowledge, (iv) data migration, (v) resistance from organization, (vi) time constraints to finish modernization projects and (vii) addressing soft factors of modernization.

## 7. Validation

The results of the survey[3] are presented in diagrammatic representation below. Note that out of 198 respondents, only 176 have indicated that they have experiences with legacy systems. Thus, the result represents the views of 176 valid respondents.

### 7.2.1. Demographics data about the participants

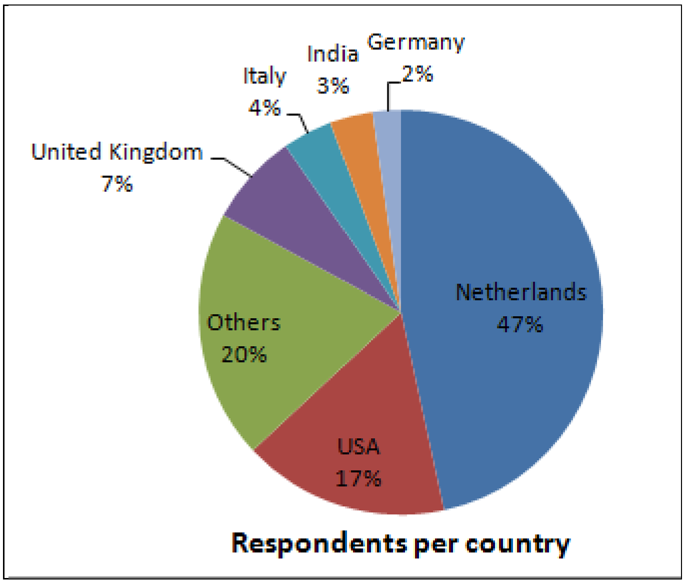


Figure 2. Demographic information about the respondents

7.2.2. Profession of the respondents

[3] The survey is available at http://servicifi.wordpress.com/legacy-survey/

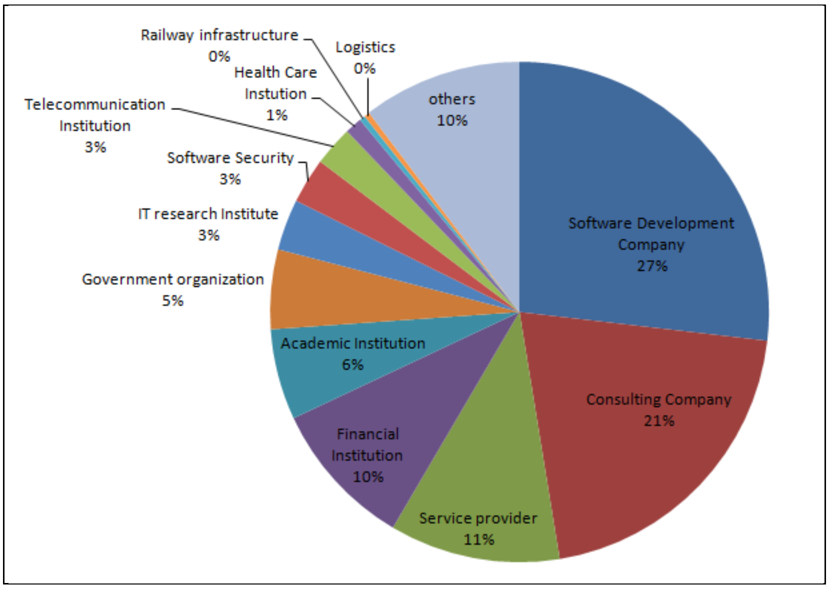


Figure 3. Profession of the respondents

### 7.2.3 Roles of the respondents

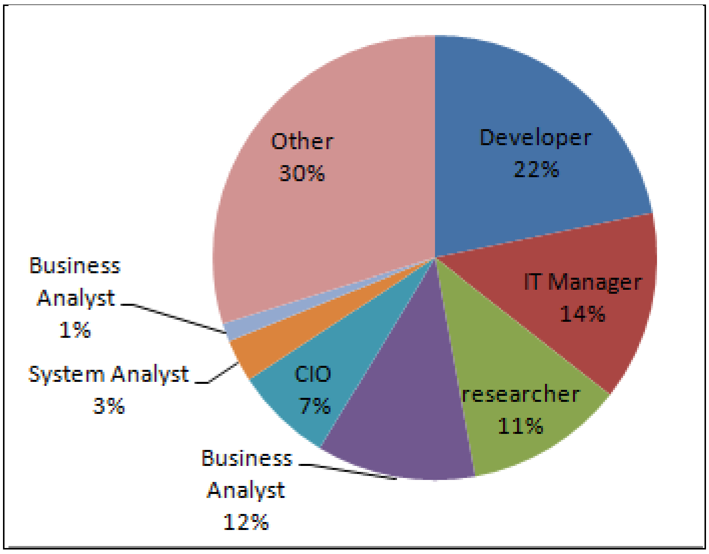


Figure 4. Roles of the respondents

### 7.2.4 Perceived benefits of the legacy systems

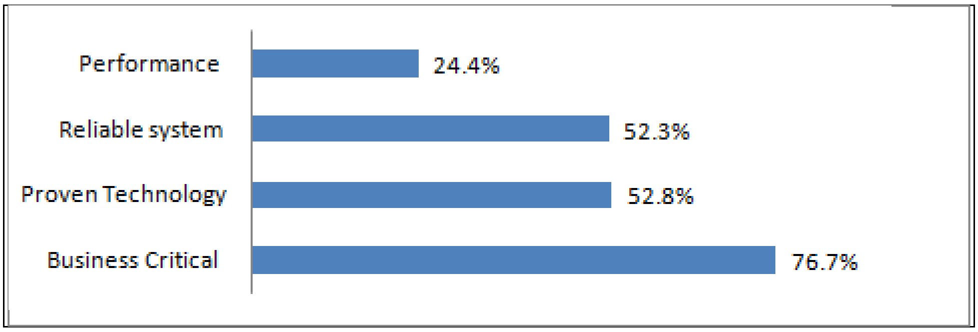
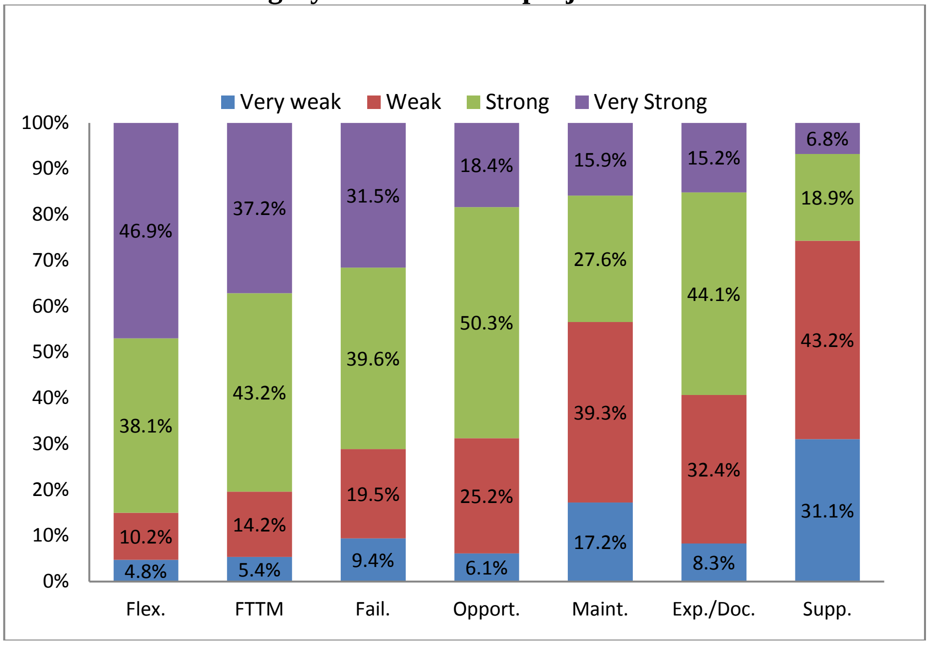


Figure 5. Perceive benefits of the legacy systems as per the respondents

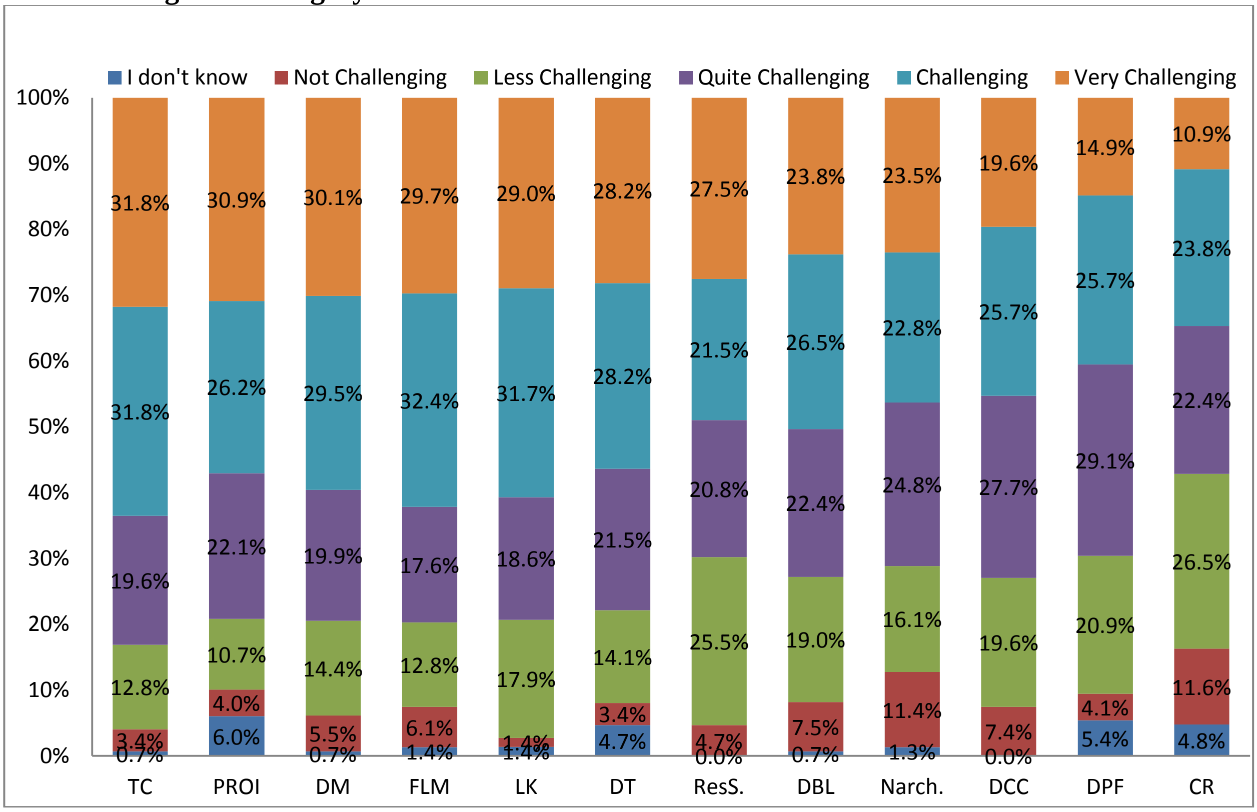
### 7.2.4 Drivers of the legacy modernization project



*Legends: Flex.:-Become flexible to change; FTTM:-Faster time-to-market; Fail.:-Prone to failure; Opport.:-Create business opportunities via mergers/acquisitions; Maint.:-High cost of maintenance; Exp./Doc.:-Lack of experts/documentation; Supp.:-Lack of suppliers/vendors*

Figure 6. Drivers of the modernization indicated by the respondents

### 7.2.5 Challenges of the legacy modernization



*Legends TC: Time constraint to finish modernization; PROI: Predicting ROI; DM: Data Migration; FLM: Funding modernization project; LK: Lack of knowledge; DT: Difficult to test; ResS.: Resistance from staffs; DBL: Difficult to extract business logic; Narch.: Non-evolvable system architecture; DCC: Difficult to communicate the consequences; DPF: Difficult to prioritize the functionality; CR: Cultural resistance from organization*

Figure 7. Challenges faced by the respondents in legacy modernization

### 7.2.6. Programming languages

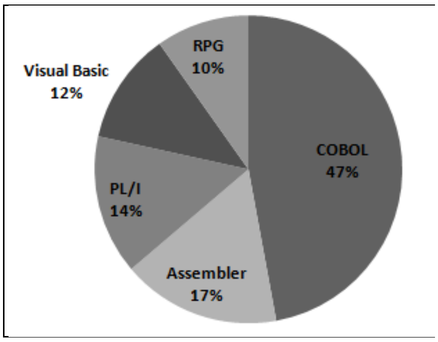


Figure 8. Legacy language as perceived by the respondents

### 7.2.7 Statistical analysis

The chi-square test was conducted to see if there is any associated between job type and the respondents who indicated programming language determines if a system is legacy. However, The analysis with (ρ= 2.9), which is far less than the critical value (α= 12.59), revealed that such an association does not exist.

## Crosstabs

**Case Processing Summary**

| | Cases | | | | | |
|---|---|---|---|---|---|---|
| | Valid | | Missing | | Total | |
| | N | Percent | N | Percent | N | Percent |
| Job Type * Do you consider the programming language as one of the factors to decide if a system is legacy? | 161 | 91.5% | 15 | 8.5% | 176 | 100.0% |

**Job Type * Do you consider the programming language as one of the factors to decide if a system is legacy?**

| | | | Do you consider the programming language as one of the factors to decide if a system is legacy? | | Total |
|---|---|---|---|---|---|
| | | | 1 | 2 | |
| Job Type | | Count | 26 | 23 | 49 |
| | | Expected Count | 26.5 | 22.5 | 49.0 |
| | Business Analyst | Count | 5 | 3 | 8 |
| | | Expected Count | 4.3 | 3.7 | 8.0 |
| | CIO | Count | 6 | 7 | 13 |
| | | Expected Count | 7.0 | 6.0 | 13.0 |
| | Developer | Count | 20 | 20 | 40 |
| | | Expected Count | 21.6 | 18.4 | 40.0 |
| | IT Manager | Count | 14 | 6 | 20 |
| | | Expected Count | 10.8 | 9.2 | 20.0 |
| | Researcher | Count | 7 | 6 | 13 |
| | | Expected Count | 7.0 | 6.0 | 13.0 |

|  |  |  |  |  |  |
|---|---|---|---|---|---|
|  | Software Architect | Count | 0 | 1 | 1 |
|  |  | Expected Count | .5 | .5 | 1.0 |
|  | System Analyst | Count | 9 | 8 | 17 |
|  |  | Expected Count | 9.2 | 7.8 | 17.0 |
| Total |  | Count | 87 | 74 | 161 |
|  |  | Expected Count | 87.0 | 74.0 | 161.0 |

**Chi-Square Tests**

|  | Value | df | Asymp. Sig. (2-sided) |
|---|---|---|---|
| Pearson Chi-Square | 4.073[a] | 7 | .771 |
| Likelihood Ratio | 4.529 | 7 | .717 |
| N of Valid Cases | 161 |  |  |

a. 4 cells (25.0%) have expected count less than 5. The minimum expected count is .46.

## 8. Conclusion

Although legacy systems and their modernization have been extensively researched, this paper attempts to address the relative absence of empirical studies of industrial perception of legacy systems and their modernization. To the best of our knowledge, this is the first attempt to empirically investigate the perceived benefits of legacy systems, problems associated with legacy system that initiate modernization, and the challenges faced during modernization from an industrial perspective. With the current status of the research some findings complement the academic literature, while some are new and surprising to the pervious observations made.

1. We document the industrial perception of legacy systems and their modernization.
2. We identify the perceived benefits of the legacy systems, drivers of modernization, and challenges that the industry faces during modernization.
3. We report the perceptional differences of and about legacy systems between the industry and academia.

As of future work, we aim to validate the findings considering a real world legacy modernization projects.

## References

1. Bisbal, J., et al., *Legacy information systems: Issues and directions.* Software, IEEE, 1999. **16**(5): p. 103-111.
2. Veerman, N., *Revitalizing modifiability of legacy assets.* Journal of Software Maintenance and Evolution: Research and Practice, 2004. **16**(4-5): p. 219-254.
3. Strauss, A. and J. Corbin, *Grounded theory methodology.* Handbook of qualitative research, 1994: p. 273-285.
4. Adolph, S., W. Hall, and P. Kruchten, *Using grounded theory to study the experience of software development.* Empirical Software Engineering, 2011. **16**(4): p. 487-513.
5. Brodie, M.L. and M. Stonebraker, *Migrating legacy systems: gateways, interfaces & the incremental approach*. 1995: Morgan Kaufmann Publishers Inc.
6. Bennett, K., *Legacy systems: coping with stress.* Software, IEEE, 1995. **12**(1): p. 19-23.
7. Comella-Dorda, S., et al., *A survey of legacy system modernization approaches*. 2000, DTIC Document.
8. Khadka, R., et al., eds. *Legacy to SOA Evolution: A Systematic Literature Review*. In AD Ionita, M. Litoiu, & G. Lewis (Eds.) Migrating Legacy Applications: Challenges in Service Oriented Architecture and Cloud Computing Environments. 2013, Citeseer.
9. Razavian, M. and P. Lago, *Understanding SOA migration using a conceptual framework.* Journal of Systems Integration, 2010. **1**(3): p. 33-44.
10. Brodie, M.L., *The promise of distributed computing and the challenges of legacy systems*, in *Advanced Database Systems*. 1992, Springer. p. 1-28.
11. Van Deursen, A., P. Klint, and C. Verhoef, *Research issues in the renovation of legacy systems*. 1999: Springer.

## Appendix A

# Interview Protocol

Throughout decade the evolution of technology information has reached at the point where modification, maintenance and development are complicated and difficult. The fact that many organizations still rely on legacy system has motivated this research to find out how legacy system are viewed from industrial perspective and their modernization towards a new platform such as service-oriented architecture, cloud computing, etc. The interview will focus on identifying Legacy System and their Modernization from industrial perspective.

The interview takes about one hour and some of the sample questions are presented below. The interview will be recorded. Furthermore, we assure you a complete confidentiality and if indicated, a report of our finding will be provided.

**Participant Background Information**

| | |
|---|---|
| Name | |
| Recent organization | |
| Current Position | |
| Primary Function | |
| Work Experience (yrs) | |
| Expertise | |
| Age | |

**Interview Questions**

The information you provide in this interview will be used mainly to construct a new insight from industrial perspective about legacy systems and modernization for research purpose. Below you will find some of the representative questions that will be discussed.

**Part I: Legacy System**

1 What is your definition of legacy system?
2 What are the important characteristics of legacy system?
3 What are the problems with legacy system?
4 What are the good things about legacy systems?
5 Can you give me the example (languages, application, etc) do you consider legacy and why?
6 How do you HANDLE the problems caused by Legacy Systems?
7 From all your applications, which Legacy System would you consider Modernizing first?
8 Could you give me an indication of how much it cost for maintaining legacy system? And could you elaborate more in what extend the money will be used for?

**Part II: Modernization**

1 What are METHODS you know for Modernization?
2 What are FACTORS / REASONS you know driver Modernization?
3 What are the CHALLANGES of doing Modernization? Can you rank them from (1 being not challenging; 5 being extremely challenging)
4 Please describe STRATEGIES or TACTICS they had employed to overcome the OBSTACLES / CHALLANGES?
5 During the Modernization, what FACTOR can determine SUCCESS / FAILURE of Modernization?
6 Could you give me an indication of how much it cost for legacy modernization? And could you elaborate more in what extend the money will be used for?
7 To what extent, the process of modernization is automated?

**Part III: Extra Question**

1. If your organization has used, is using, or is currently considering legacy system modernization practices or processes that have not been addressed in this survey, or you consider an improvement or innovation to legacy system modernization practices, please describe the process or methodology below.

**Participant's feedback**

| | |
|---|---|
| Please describe any unusual circumstances and/ or events that had any bearing on the interview such as interruptions, language difficulty, etc.: | |
| What do you think need to be improved in order to make the interview become better? | |
| Additional comments: | |
| Interviewer Qualifications | □ Knowledgeable<br>□ Structuring<br>□ Clear<br>□ Gentle<br>□ Sensitive<br>□ Open<br>□ Steering<br>□ Critical<br>□ remembering<br>□ Interpreting |

**Date :**

________________________

**Researcher / Interviewer**

________________________

**Participant / Interviewee**

## Appendix B

# Category 1. Characteristics

***Characteristic of the legacy systems***

### Concept 1.1. Maintainability.

*Degree of effectiveness and efficiency with which a product or system can be modified by the intended maintainers.*

| | | |
|---|---|---|
| [1.1.1] | Modifiability | Degree to which a product or system can be effectively and efficiently modified without introducing defects or degrading existing product quality. |

### Concept 1.2. Reliability.

*Degree to which a system, product or component performs specified functions under specified conditions for a specified period of time.*

| | | |
|---|---|---|
| [1.2.1] | Robust | Ability of a computer system to cope with errors during execution or the ability of an algorithm to continue to operate despite abnormalities in input, calculations, etc. |
| [1.2.2] | Stable system | Software is stable only if it does not have problems that cause it to stop working. A single instance of loss of availability or a system failure would indicate that the software is not stable. |

### Concept 1.3. Usability.

*Degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.*

| | | |
|---|---|---|
| [1.3.1] | User Interface Aesthetics | Degree to which a user interface enables pleasing and satisfying interaction for the user. |
| [1.3.2] | Learnability | Degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use. |

### Concept 1.4. Performance Efficiency.

*Performance relative to the amount of resources used under stated condition. Resources can include other software products, the software and hardware configuration of the system, and materials (e.g. print paper, storage media).*

| | | |
|---|---|---|
| [1.4.1] | Time Behavior | Degree to which the response and processing times and throughput rates of a product or system, when performing its functions, meet requirements |
| [1.4.2] | General Performance is good | Performance can be defined as the capability of a system to process a given amount of tasks in a determined time interval. |

### Concept 1.5. Functional Suitability.

*Degree to which a product or system provides functions that meet stated and implied needs when used under specified conditions.*

| | | |
|---|---|---|
| [1.5.1] | Functional Completeness | Degree to which the set of functions covers all the specified tasks and user objectives. |
| [1.5.2] | Functional Appropriateness | Degree to which the functions facilitate the accomplishment of specified tasks and objectives. |

### Concept 1.6. Compatibility.

*Degree to which a product, system or component can exchange information with other products, systems or components, and/or perform its required functions, while sharing the same hardware or software environment.*

| | | |
|---|---|---|
| [1.6.1] | Interoperability | Degree to which two or more systems, products or components can exchange information and use the information that has been exchanged. |
| [1.6.2] | Adaptability | Degree to which a product or system can effectively and efficiently be adapted for different or evolving hardware, software or other operational or usage environments. |

### Concept 1.7. Supportability.

*Support from supplier / vendor / provider for software or hardware in legacy system.*

| | | |
|---|---|---|
| [1.7.1] | Unsupported Supplier | Suppliers / vendors don't support their product anymore. For example, old versions of hardware / patches / updates are no longer available. |

### Concept 1.8. Flexibility.

*Degree to which a product or system can be used with effectiveness, efficiency, freedom from risk and satisfaction in contexts beyond those initially specified in the requirements.*

| | | |
|---|---|---|
| [1.8.1] | Not Flexible | Legacy systems are rigid and not flexible due to the fact that they are difficult and take longer time to implement new technology. |

### Concept 1.9. Proven Technology.

*Systems that have been demonstrated or verified without doubt to comply with business requirements for long period of time.*

| | | |
|---|---|---|
| [1.9.1] | Old system | Systems which are older than 10 year and was built with an outdated technology. |
| [1.9.2] | Availability | Degree to which a system, product or component is operational and accessible when required for use. |

### Concept 1.10. Risk.

*Risks identified in legacy system.*

| | | |
|---|---|---|
| [1.10.1] | Risk of running legacy system | Risks are increasing if company is still running legacy system, such as risk of continuity. |

### Concept 1.11. Architecture of the systems.

*The structure or structures of the system, which comprise software elements, the externally visible properties of those elements and the relationships among them.*

| | | |
|---|---|---|
| [1.11.1] | Large multiple systems | More than one system does the same functionality in which resulting large systems. Big in size of the system including enormous line of codes and connectivity among applications. It also measures the volume of the system. |
| [1.11.2] | Database issue | Problem with database in legacy system. Including Unstructured and inappropriate use of data in legacy system. |
| [1.11.3] | Back Office | System that does not interact directly with the customer. It is used to administer operations that are not related to any direct sales effort (such as a salesperson with a customer present) and interfaces that are not seen by consumers. |
| [1.11.4] | Complicated system | System in which contain of a lot and complex rules. Complexity determine structural characteristics by examine how object are interrelated. |
| [1.11.5] | Poor design and ecosystem | Legacy system is characterized by damaged structure of application and bad ecosystem. |

### Concept 1.12. Knowledge around the systems.

*The availability of knowledge around legacy system, such as documentation, programmer / developer, analyst.*

| | | |
|---|---|---|
| [1.12.1] | Knowledge become scarce | Condition in which there is not sufficient knowledge available around the system including lack of documentation and expert (programmer, analyst, etc.) in legacy system. |

### Concept 1.13. Business Strategy.

*Legacy system in regards to business strategy of organization.*

| | | |
|---|---|---|
| [1.13.1] | Doesn't fit with future strategy | Inability of legacy system to support business functionality. It is not a part of the organization's strategy because it is getting more difficult to in-line with business strategy. |
| [1.13.2] | Business Critical | Systems whose disruption or malfunctioning will cause a failure in business operations. The business critical systems contain vital features/functionalities and their failure can result on serious financial & legal problems, damages and other penalties. The systems also bring benefit or profit to organization. |

### Concept 1.14. Finance.

*Legacy system from financial point of view.*

| | | |
|---|---|---|
| [1.14.1] | Expensive in maintenance | The cost for maintenance and operating the legacy systems is perceived high. |

## Category 2. Challenges

***The difficulties that arise in legacy modernization project.***

### Concept 2.1 Data Migration.

*The process of transferring the data from the legacy system into the new system.*

| | | |
|---|---|---|
| [2.1.1] | Data Migration | The difficulties that arise in the process of transferring data between data storage systems, data formats or computer systems, including translating the data from one format into the other formats. |

### Concept 2.2 Lack of knowledge.

*Limited knowledge to assist the process of legacy modernization.*

| | | |
|---|---|---|
| [2.2.1] | Lack of resources | Lack of reference (e.g. documentation) and limited manpower to involve in legacy modernization. |

### Concept 2.3 Non evolvable Architecture.

*Obsolete architectural design of the system.*

| | | |
|---|---|---|
| [2.3.1] | Poor system architecture or infrastructure | Old fashion of building the system or obsolete IT landscape (E.g. monolith and, hardcoded, spaghetti architecture). |
| [2.3.2] | Difficult to test | Difficult to complete test case. Test case is a process of *verifying* and *validating* that a software application or program to meets the business and technical requirements that guided its design and development, and to works as expected. |

### Concept 2.4 Time constraints.

*Time restriction in the process of legacy modernization.*

| | | |
|---|---|---|
| [2.4.1] | Time constraints to finish legacy modernization | Limited time available to finish legacy modernization, including time management in every step of legacy modernization. |

### Concept 2.5 Difficult to extract and prioritize business logic.

*The challenge in govern the process of legacy modernization.*

| | | |
|---|---|---|
| [2.5.1] | Difficult to extract business rules or knowledge | Difficult to extract business rules or knowledge from legacy system. |
| [2.5.2] | Difficult to effectively prioritize the functionality for modernization | The difficulties to govern the process of modernization during long time period of time (e.g. prioritize new functionality, maintain the old and new system, etc.). |

### Concept 2.6 Resistance from Organization.

*The acceptance from the organization and the users of the legacy system.*

| | | |
|---|---|---|
| [2.6.1] | Cultural resistance in organization not to adapt new system | The ability of organization to adapt to a new culture during or after the process of modernization. |
| [2.6.2] | Resistance from the current users or maintainers in the organization | Uncooperative manner from current users or maintainers in an organization or behave of certain people which are oppose the process of legacy modernization |
| [2.6.3] | Communicate the reasons or consequences of modernization | Difficult to make people, who involve in legacy modernization, to accept the legacy modernization together with its consequences. |

### Concept 2.7 Funding.

*The difficulty in getting the business sponsor in the process of legacy modernization.*

| | | |
|---|---|---|
| [2.7.1] | Funding legacy modernization projects | Difficult to get money to support the legacy modernization project. |
| [4.2.2] | Predicting Return of Investment (ROI) of modernization | Difficult to create business case as a scenario in which has positive return on investment. |

## Category 3. Drivers

***Factors that provide impulse or motivation to do legacy modernization.***

### Concept 3.1. Become Flexible.

*Desire to get the flexible system.*

| | | |
|---|---|---|
| [3.1.1] | Agile to business requirements | Systems are not flexible enough to be modified or changed when new requirements are coming. |
| [3.1.2] | Create new business opportunity | Legacy modernization allows organization spread their business more widely. |
| [3.1.3] | Faster time to market | Quickly deliver the product or service to the customers. |

### Concept 3.2. High maintenance cost.

*Desire to reduce the cost of maintenance and operational of the legacy system.*

| | | |
|---|---|---|
| [3.2.1] | Reduce cost of maintenance and operation | The new systems produced by doing legacy modernization are expected to have low cost of operation and maintenance. |

### Concept 3.3. Prone to failure

*Vulnerable system.*

| | | |
|---|---|---|
| [3.3.1] | Prone to failure | Systems are vulnerable to be failed and organizations cannot afford if the systems failed. |

### Concept 3.4. Lack of knowledge.

*Factors that motivate organization to do legacy modernization from business point of view.*

| | | |
|---|---|---|
| [3.4.1] | Lack of supplier / vendor | Limited supplier or vendor to support changing business requirements |
| [3.4.2] | Lack of documentation / experts | The knowledge around the legacy system is becoming scarce |

## Appendix C

Quotations by the respondents

The process of coding is basically based on the result from transcribing the interviews. Below, we present the quotes and respective mapping to the respective codes from our participants during their interview sessions.

## Participant 1

| | | |
|---|---|---|
| **[1.3.2]** | ***Learnability*** | • *System when you get somebody new in it would take a lot of time when these people, if you can get them at all when these people have to learn how the system works and how they can maintain it.* |
| **[1.5.1]** | ***Functional completeness*** | • *The first evolve IT was looking at an automating tasks so it is a normal history of automation at first. Handling all kind of standardizes administrative stuff which can be easily formalized and automated.*<br>• *Like I said before the functionality of the system is  basic but old.* |
| **[1.5.2]** | ***Functional Appropriateness*** | • *A legacy system is a system that works. OK, it is there because it is there and it is old because it works and it works fine.* |
| **[1.9.1]** | ***Old system*** | • *Most of the legacy systems are older than 20-30 years* |
| **[1.2.2]** | ***Stable system*** | • *So they're well performing and they are very good secured and they're stable ok.* |
| **[1.4.2]** | ***General Performance is good*** | |
| **[1.6.1]** | ***Interoperability*** | • *They operate within an environment. If the systems stay the same, and the system surround the systems, the environment is evolving to new technologies you get a discrepancy and you get less possibility to properly interconnect with the older system.*<br>• *It will be become more difficult over the years to connect to your new systems mainly in front-end environment. Yeah… I think those are the bigger problems.*<br>• *If it isn't properly designed, then you will run into trouble faster.*<br>• *If you have ill-designed legacy system or old fashioned design as a monolith system, that might still be a challenge.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | |
| **[1.6.2]** | ***Adaptability*** | • *So, the problem of legacy system, I think are not in the systems themselves but in the supporting technology or its only current for a certain period.* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *People who understand technology, the old technology and the way the system has been evolved and sometimes it is not well documented, the knowledge about the system become scarce*<br>• *You become dependent on scarce set of skills and specific people and that can make them more difficult to maintain.*<br>• *A lack of documentation can be a problem. But that's with any language any system. It isn't properly documented.*<br>• *They knew they were getting pretty close to the border where you hit a brick wall and trying to get someone with knowledge on the system and to extend them or to amend them.*<br>• *It will be very difficult to get knowledgeable people.*<br>• *Experts are becoming more costly..[] it's not well documented, the knowledge about the system become scarce. And you become dependent on speciffic people and that can cause difficulty to maintain.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.1.1]** | ***Modifiability*** | • *They knew they were getting pretty close to the border where you hit a brick wall and trying to get someone with knowledge on the system and to extend them or to amend them* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *And they were not dissatisfied with the systems.*<br>• *I am satisfied with the way is working right now.* |
| **[1.13.2]** | ***Business Critical*** | • *Most of the systems of the legacy systems are the core system of the bank.* |
| **[1.11.1]** | ***Large system*** | • *it is basically still the administrative stuff which works in batch mount on large databases.* |
| **[1.2.1]** | ***Robust*** | |

|  |  |  |
|---|---|---|
| **[1.9.2]**<br>**[1.5.2]** | ***Availability***<br>***Functional Appropriateness*** | • *The system has been around for a long time and has been tuned to stability, robustness, availability and so on. So they're well performing and stable. Functionalities [Quality attributes] that count are stability, robustness, reliability and availability of this system".* |
| **[1.10.1]** | ***Risk of running legacy system*** | • *Risk is one of the other factors.*<br>• *Money is not a largest problem, risk is the largest problem. You know, they can afford these systems to failed only for 1 hour.*<br>• *It currently works but they worry if they continue it, can it continue long enough into the future?*<br>• *On the second level you might say that people start beginning to worry if it can continue this way.* |
| **[2.7.1]** | ***Funding the legacy modernization project*** | • *Now you see with this project, the funding is granted...so, business only gives small amount of money and next year small amount of money...and a little bit more... little bit more... ooo this guy is now achieving something..*<br>• *You have problems with 10% of the systems, which is used 50% of the time but the other 50%, is 90% of the code and it still works fine. So, how do you judge when is the right moment if it cost a lot in migrate? it is a tough question.* |
| **[1.8.1]** | ***Not flexible*** | • *So, we are able to more quickly adapt the certain components on our own.*<br>• *Increasing flexibility.* |
| **[3.3.1]** | ***Prone to failure*** | • *The other main objective is to get smaller, less complex components. So disentangle the logic and create more self-contain components.*<br>• *Because most of them were monolith...ok…so, I think if you looking at modernizing is not question of the language that has changed that much for the certain type processing but of course influent more on design and how to optimize things.*<br>• *We are revising the central role system who work tidily couple, so more or less monolith, decomposing them in more... more service oriented way.* |
| **[3.4.1]** | ***Lack of supplier / vendor*** | • *Also continuity, so the driver behind staying current with this technology is that you want to have continuity of your core system.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *That all depends on how the systems are maintained, have been maintained out of the core of long period. They are becoming more costly.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | • *The way they were programed, the way they were designed…ok…because most of them were monolith*<br>• *It can still be a problem of the design or the requirements specification. And of course if you have legacy system ill designed or old fashion designed as a monolith that might still be the reason for you to change.*<br>• *Because most of them were monolith...ok…so, I think if you looking at modernizing is not question of the language that has changed that much for the certain type processing but of course influent more on design and how to optimize things.* |
| **[2.3.1]** | ***Poor system architecture or infrastructure*** | • *Making the right decision in how to componentize the system.* |
| **[2.5.1]** | ***Difficult to extract business rules or knowledge*** | • *[Q] For example like a lot of people might not know what the inner functionality of legacy system because the developers are not there. [A] Yeah…yeah.* |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | • *They run out of budget..they run slightly out the time. [...] that's mainly to do with scarcity of people on the legacy system.* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *No...not that much... I think the people working on this type of things, they are always some people would like to staying that comfort zone, but I think by now is very clear that if you don't adapt to the environment and people go on without you.* |
| **[1.13.1]** | ***Doesn't fit with future strategy*** | • *If it still suit their needs in the future.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *How do you do the whole communication of culture of geographical area, how do you secure things being developed over there, how you check things, how you vitamin to your environment... a lot of stuff.* |

# Participant 2

|  |  |  |
|---|---|---|
| **[1.12.1]**<br>**[1.11.5]** | ***Knowledge become scarce***<br>***Poor quality of design and ecosystem*** | • *In production it is legacy. My experience is and...that...at the moment the system is built...yeah...it is well documented, it is often well structured. But at the moment it is going to production, you get the ad hoc changes on it and those kind of things, and what you see is that within...for some companies is within a year, for others is within 3 years, the documentation is outdated, the structure of the application is damaged...yeah...and the problems start.* |
| **[1.11.5]**<br>**[2.4.1]** | ***Poor quality of design and ecosystem***<br>***Time constraints to finish legacy modernization*** | • *IT is forced to come up with the quick and dirty solutions often and doesn't have time afterward to adjust it, to make it well suited in the design of the application itself.*<br>• *Time is sometimes an issue, budget is less an issue.*<br>• *If requirements...new requirements are put on the application, what you then see is that it is often quick and dirty ad hoc and those kind of things.*<br>• *Business wants something yesterday.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | • *Knowledge erosion is the biggest problem there.* |
| **[1.12.1]**<br>**[3.4.2]** | ***Knowledge become scarce***<br>***Lack of documentation / experts*** | • *If you still have the people who build it and maintain it, then it still works fine. But if they leave the company, then they end up in serious problems. And where you see that is often the pressure between business and IT.* |
| **[1.2.2]**<br>**[1.11.3]** | ***Stable system***<br>***Back Office*** | • *Legacy system is what you find in back office. Because back office processes are more stable than mid-office and front-office. So you have less business requirements on that popup.*<br>• *You also see is that because they are very knowledgeable people, the application itself doesn't suffer from many defects anymore. Over the years all the defects have been solved.* |
| **[1.4.2]** | ***General Performance is good*** | • *[Q] Already there...so performance is still good in legacy system. [A] Yeah.* |
| **[3.1.3]** | ***Faster time-to-market product*** | • *Putting new requirements, time to market is long and they wanted nowadays…what we find is that within business they come up with an idea, they want to have an initial estimate and time to market estimate within a few weeks and then average implemented in 3 months or something like that.* |
| **[1.8.1]** | ***Not Flexible*** | • *It's more that the business has the requirements which cannot be implemented quickly enough in the legacy.* |
| **[1.1.1]** | ***Modifiability*** | • *So, what you then often see is that the old legacy system which is the kind of backbone still stays there and that is used in is built in the midterm or more in the area of mid-office and front-office. Just don't touch the legacy application.* |
| **[1.5.2]** | ***Functional Appropriateness*** | • *It may be a creepy code but it runs and it runs without any failure.*<br>• *it is not the question of quality of service of the application.* |
| **[1.11.1]** | ***Multiple system*** | • *A lot of banks and insurances and ending up with 3 or 4 systems overlapping functionality. Because they cannot decommission the old one.* |
| **[1.13.2]** | ***Business Critical*** | • *So, one of the nice things about that is we also a lot of work at customers decommission really old application.* |
| **[1.7.1]** | ***Unsupported supplier*** | • *I mean what you trigger or also things like for instance a hardware. I have seen customers running the systems on the hardware that run out of support. And even hardware that if something is broken down they have to look on E-bay to find spare parts.*<br>• *So when your environment [legacy systems environment] runs out of support then it is really dying and if that's true then you are already late.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *Yeah...cost reduction.*<br>• *Cost of mainframe is perceived as high. And they want to migrate way to migrate away to mid-range systems.* |
| **[3.1.2]** | ***Create new business opportunity*** | • *People really want instant correct information. And that might drive some companies to...well...to do modernization...yes...to enable the legacy application to cope with those kind of information request.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | • *Very difficult is for instance in a new old client-server environment, power builder for instance...yeah...That is just a disaster to work on it, because it is very easy...a lot of logic is concentrated on the client side.*<br>• *And what you also nd is that one application is developed that it uses also the database of another one. And it's not through a normal interface but it's via back door to get some data over there*<br>• *They have a lot of legacy applications and it is hardcoded often in the code.* |
| **[2.2.1]** | ***Lack of resource (e.g. documentation, expertise, etc)*** | • *You need C level support for that and you have to involve business always. Because if you have impact on the business and especially when you look at large modernization it means that the business will need to be involved.*<br>• *The biggest challenge I find always is to get C level support for this kind of program.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *Get really commitment from that level...from that level to really set up the whole period. And not well…OK next year...o well...we have less budget...well we stop and we quit.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | • *If the company or that customer doesn't governs the modernization as one overall program, but looks at the application individually.* |
| **[1.3.2]** | ***Learnability*** | • *New one comes in, he has to learn about it...well...it is not something you can do in just a few months.* |
| **[1.9.1]** | ***Old system*** | • *So, if you look at applications in that category what you will find is that system runs for 20 years, 25 years.* |
| **[1.6.2]** | ***Adaptability*** | • *That is very hard to align that with a very old system of course.* |
| **[2.7.1]** | ***Funding legacy modernization*** | • *And what is the alternative? Yeah...is to rebuild the system? I can tell you, that is more expensive than restructuring it and making it* |

| | | |
|---|---|---|
| | ***projects*** | *maintainable.*<br>• *So they have not implemented that, and it is a lot of time money to still implement it, so they keep up running the old one.*<br>• *So you always have business case over there. And the moment that business can yeah...so it is not positive and for many companies if the Return on Investment is not receive one year and say we don't do it.*<br>• *They are always looking for a short Return on Investment. One you put the money in, they want to earn it back. And...But what you now see is last year I had customer who said I want to it but I want to earn it back in 6 months. If you have talk about a project of at least one year you cannot and never earn it back in 6 months. So, only if we get in more economic better times and we can stretch it a little bit because modernization is not something that you do in a few months. That will takes a year or 3 years or if there is very big program you can find yeah...*<br>• *Modernization in such a way that it is cost effective to do the job. It's not rebuilding a complete new...of a building...a complete a new system or turning the legacy system upside down or inside out. But just making those kind of changes that is cost effective.* |
| **[2.1.1]** | ***Data migration*** | • *And at the time they have a new system they want to migrate all the data from the legacy to the new one. And then they find that...oouww...but here we have something...ouww...we haven't had foreseen that...yeah...* |
| **[1.5.1]** | ***Functional completeness*** | • *It performs basic func...still support basic functionality of the business processes.* |
| **[1.4.1]** | ***Time behavior*** | • *But on the other hand, it does the job efficiently, it is fast and that is more less if talk about back office system, that is where combination are looking for ...yeah...reliable, fast…* |
| **[1.4.2]** | ***General Performance is good*** | |

# Participant 3

| | | |
|---|---|---|
| **[3.2.1]** | ***Reduce cost of maintenance and operation*** | • *[XXX] looked and said it cost a lot of money, a lot of danger. So last 10 years [XXX] migrated 5 big systems.* |
| **[1.1.1]** | ***Modifiability*** | • *So the technical drivers are old, difficult to maintain, the knowledge was not easy to get because PL/1 programmer is not easy to find. From business side, it becomes expensive to technically fulfill the business reason.* |
| **[1.9.1]** | ***Old system*** | |
| **[1.12.1]** | ***Knowledge become scarce*** | |
| **[1.14.1]** | ***Expensive in maintenance*** | |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.11.5]** | ***Poor design and ecosystem*** | • *It's monolith, everything in one source code and in one machine.* |
| **[1.4.2]** | ***General Performance is good*** | • *It's [Performance] Enormous. Enormous quick. So the old system itself didn't have [performance issues]...I don't think the performance is a problem."* |
| **[2.1.1]** | ***Data Migration*** | • *How we are going to migrate the data into the new system. Because it is operational it has to be real time accuracy. Since it is operational system, you have to do it right, one time. There is no re-tried, because if you start working with the new system and you find out that something is wrong with the database, it is hardly impossible to repair because at that point the whole situation is not up to date anymore, is not maintain anymore so it is not there anymore. It's a really big challenge. And also some of the data is inconvenience, it is not properly migrated.* |
| **[1.11.2]** | ***Database issue*** | • *Yeah…and that is the problem, because it does not have a referential data.* |

# Participant 4 and Participant 5

| | | |
|---|---|---|
| **[3.4.1]** | ***Lack of supplier / vendor*** | • *Legacy system is a system which service life has expired.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *Maintenance is very expensive...too expensive.*<br>• *The maintenance is cost higher.* |
| **[1.1.1]** | ***Modifiability*** | • *Not maintained that means that no functional updates or no security updates.* |
| **[1.9.1]** | ***Old system*** | • *That's obsolete technology.* |
| **[1.2.2]** | ***Stable system*** | • *Most of the time it is stable.* |
| **[1.3.2]** | ***Learnability*** | • *People are used to the system. They know how it works.* |
| **[1.9.2]** | ***Availability*** | • *Proven technology is often the reason why they are still in us* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *But the documentation of the old system is a primary* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[3.1.1]** | ***Agile to business requirements*** | • *I think in government, we have influence of politic.* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *[Q] If the organization wants to modernize their legacy system, there is certain people who try to not cooperate because they're an expert in legacy system. By the time when legacy is modernized, they might lose their job. Do you see kind of phenomena in organization? [A] Yes.* |
| **[1.6.1]** | ***Interoperability*** | • *[Q] Is it difficult to interconnect with...interconnect the application with your legacy system. [A] Yeap.. it is difficult. We try to standardize the interface.* |

# Participant 6

| | | |
|---|---|---|
| **[1.13.2]** | ***Business Critical*** | • *I think it is a core system, most of the time.*<br>• *Like the core transaction.* |
| **[1.8.1]** | ***Not flexible*** | • *They also have a lot of limitations on flexibility.*<br>• *Very rigid...let say if you want to change it like data structure, it is not clear for me what exactly the data structure is.*<br>• *Yeah... it is not flexible. It's a lot of hassle.*<br>• *But actually, like most companies now say but we want to be flexible.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | • *And I could see people are using at the...they had a legacy system but during time they also have complementary systems like workflow system. And using both and using all top to do little bit here, little bit there.* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *But still you can see they trust in it, because it is working...some guys said last week, it is working for 20 years now.* |
| **[1.5.1]** | ***Functional completeness*** | • *But for me is all about if you would define, it is about limitation. You see they have street field which is limited to certain characters.*<br>• *But it is not process thinking.*<br>• *It is only simple logic, this is simple operations.*<br>• *There is no a lot of …most of the time, there's not a lot of complex logic in it.*<br>• *So it is just you can like update, maybe the address of the client.* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *I think that's a lot of knowledge which is only in their head.*<br>• *Like if you're young like you and me who decides to start learning COBOL and working in it. It's also they know that system because they work with it for 10 years. If you want...what kind of guys for designer if you like end of 20ies, you just graduated, and nobody decides to start COBOL*<br>• *[Q] Is that means...Is that means that the maintenance of legacy system also expensive or not? [A] Yeah… it is... so and even though it is not expensive, it is hard...I think it is hard to find like skilled people or people who want to do it and who know that all that things.*<br>• *There is more IT, they also spend more in IT.* |
| **[1.14.1]** | ***Expensive in maintenance*** | |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.2.1]** | ***Robust*** | • *Yeah...just like what I said before, it is robust.*<br>• *I think these all UNIX systems. They do not crash often.* |
| **[1.9.2]** | ***Availability*** | • *Like the system they are online, they are...they always available and because there is no a lot of ...most of the time, there's not a lot of complex logic in it.* |
| **[1.1.1]** | ***Modifiability*** | • *Changes are hard.*<br>• *A lot struggle to change that, because all these logic about that is totally integrated into the system.* |
| **[1.3.1]** | ***User interface aesthetics*** | • *Interface...user interface sucks...user interface is not very...is not modern...you get this back…yeah this old screen with some data on it.* |
| **[1.3.2]** | ***Learnability*** | • *I tried to explain it last time to someone who was taking something over from me. It's like, I don't get it.* |
| **[3.1.1]** | ***Agile to business requirements*** | • *There was a new regulation, which says they had to make policy to be gender independent.*<br>• *They had also too much different customization from the policy...So, they're also bringing their policies pack to limited...was really complex*<br>• *That's an important one legal change.* |
| **[1.11.5]** | ***Poor quality of design and*** | • *Also the business logic, so...let say you want to determine what is the premium? that is all hard coded. That's especially hard coded, in* |

| | | |
|---|---|---|
| | ***ecosystem*** | *the logic to find it, to change it, it is very hard.* |
| **[1.11.2]** | ***Database issue*** | • *Data garbage.*<br>• *So the quality of the data.* |
| **[2.7.1]** | ***Funding legacy modernization projects*** | • *Should also have budget right like if he says we are going to do that, and then you can also put the money that way, because sometimes that's a financial system, financial department involve.*<br>• *Main thing is always budget, so they should be money.* |
| **[2.1.1]** | ***Data migration*** | • *I've seen very strange thing, it is just missing for maybe like 5% of the record some really important data it is just missing...hahaha...*<br>• *Like why don't we use our own, if we started after their own databases tables, and work like that, the project would be quickest so much.*<br>• *Could be like 40 small insurance companies which in the time like 20-30, merge to each other and merge again. Imagine every company have their own data, so they migrate their data at the time.* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *Because what is our need if we have a new system, which is working not with COBOL. Who is gonna [going to] need me anymore, so they ditch me after it [modernization] is done. So, why should I cooperate?"*<br>• *People all have kind of different motive not to work along.*<br>• *Like people should cooperate but how do you get them there for* |
| **[2.6.1]** | ***Cultural resistance in organization not to adapt new system*** | • *Live close to Rotterdam, i like my job because I can go to my work in 15 minutes, and then at the end of the day 15 minutes. I like my…the way my life is. I like my job, I like the department.* |
| **[1.10.1]** | ***Risk of running legacy system*** | • *I think the company will go out of market.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *Someone like...sometimes you see like make the decision, but he should be high up in the chain like if they say it…how do you call it...mandate...it should have mandate, the guy should…I think he should be on such a high position.* |
| **[1.11.1]** | ***Multiple system*** | • *Within history they first had this system and then a few years later they thought o yaa...now we need this system. So they have multiple systems for policies.* |
| **[1.9.1]** | ***Old system*** | • *The older system.*<br>• *It's like if they choose the system, they're going to use it for a like 10 years.* |
| **[3.2.1]** | ***Reduce cost of maintenance and operation*** | • *They wanna have rather like one person having just one user interface.* |
| **[2.2.1]** | ***Lack of resource (e.g. documentation, expertise, etc)*** | • *You need people, skilled people to make services.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *There's also misunderstanding.*<br>• *You get a lot blaming...sometimes we get the right people together to solve, just solve the problem, it is a big challenges.* |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | • *Just time...so if you want to change something we say like can you change it. O yeaah...and then...it also thing they say like yeah...we only have limited capacity, because they don't have guys, old guys doing it.*<br>• *First you have to change processes, you have to implement it. I think that takes some time. If you choose who is going to implement it? Who's going to test it? How you are going to migrate the data to the new system...bla..bla..bla..and then It's also maintenance about the time I said...implement it you want to...there is be maintained by some....There is also going to be like some minor changes during...but if they chose, they are going to use it for the coming 5 or 10 years.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *Like also problem this kind of guy that's not really good in communication.* |

# Participant 7 and Participant 8

| | | |
|---|---|---|
| **[1.9.1]** | ***Old system*** | • *So legacy system for me is an older system.* |
| **[1.11.1]** | ***Large system*** | • *On average is a large system because small systems don't impose to big problem.* |
| **[1.11.4]** | ***Complicated system*** | • *A lot of functionality, lot of business rule, business logic implemented and it is develop over the years to very mature degree.*<br>• *So there's always something with technical complexity, a lot of components, a lot of layers, lot of programs and interfaces interacting with each other.* |
| **[1.2.2]** | ***Stable system*** | • *And that basic functionality is stable for very long time because that's all for whole year, it is working that way.*<br>• *The functionality is stable...just let it stay as it is.* |
| **[1.4.2]** | ***General Performance is good*** | • *So the functionality on average is OK.* |
| **[1.1.1]** | ***Modifiability*** | • *But the problems is with the technical quality, the maintainability is for several reasons bad.*<br>• *You have encountered some problems with maintainability.*<br>• *Cannot maintain it efficiently because quite simple change takes so much time because no one knows exactly the system works.*<br>• *That's a big change for a system like that, and that kind of changes that is difficult for legacy system. So it impacts very much the existing functionality.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | • *You made adaptation on and adaptation on...and the internal technical structure is not very optimal as it was, just after you started developing the system.*<br>• *So you fix something and something else does not work anymore. So, by doing the changes and changes and changes, it gets the...less...less optimal the system.* |
| **[1.6.2]** | ***Adaptability*** | |
| **[1.12.1]** | ***Knowledge become scarce*** | • *The documentation is not...if any documentation is not actual. The knowledge of the people who built it is gone because people are going away to other job.*<br>• *Or I do not have the personal anymore who can do it. Or I don't have documentation and I do not even know how it should work the system.*<br>• *The problem is our COBOL application is that the people works on it are retiring. So they are leaving the company then...So our knowledge is going away about the system.*<br>• *Lack of knowledge about what it is going...what now effect…well being executed by the program and the coding. That can be the main challenge.*<br>• *[Q] Yea...yea....so do you think that to find people with the knowledge in COBOL is difficult now? [A] Not at the moment. But it will be in the future.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | • *In the time pressure. You do not have the time to do it good or to refactor the coding or there is a pressure because you do not have sufficient knowledge about how it should it works.*<br>• *Legacy modernization project lasts too long. We plan modernization for 3 years, and after 5 years we stop the whole modernization, and start it over.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | • *[Q] Yea…yea...so do you think that to find people with the knowledge in COBOL is difficult now? [A] Not at the moment. But it will be in the future.* |
| **[1.11.3]** | ***Back office*** | • *I think another characteristic of legacy system is that they are mainly back office system.* |
| **[3.1.1]** | ***Agile to business requirements*** | • *Law, can you give an example of what law maybe. This is from politics the change of the retire...what it is…retirement.*<br>• *But the problem is that legislation of pension fund is changing continuously.*<br>• *Definitely business requirement changing overtime.* |
| **[3.1.3]** | ***Faster time-to-market product*** | • *To launch quicker new product for the customers.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *So the driver that the maintenance of legacy system causes a lot of money, well it is only a problem when our customer will not pay anymore for it and we still have the obligation to keep it in the air for some years. That is another aspect of legacy system, you have a lot of customers. Together they are paying a lot of money to keep it in the air.*<br>• *So one of the main factor is the cost to do changes becoming higher and higher, compare to the change itself.*<br>• *The hardware on which the system runs for example mainframe becoming too expensive compare to cheaper hardware nowadays available.* |
| **[3.4.1]** | ***Lack of supplier / vendor*** | • *Yes...it is strengthens the lifecycle of the system.* |
| **[2.7.1]** | ***Funding legacy modernization projects*** | • *Budget we have for that... Time available... yeah...capacity.*<br>• *And it is not easy to migrate that to another system, to change to another system. It always, it requires a big efforts to change to another technology or to change the system to another system. So that's the reason why the old system keeps in use for longer and longer.* |
| **[1.7.1]** | ***Unsupported supplier*** | • *System is built with program languages, library and frameworks that no longer support or the suppler no longer support.* |
| **[1.5.1]** | ***Functional completeness*** | • *It is not technical automation. It's not embedded software for example.* |
| **[1.13.2]** | ***Business Critical*** | • *And in fact some of our legacy systems are the most profitable of our system. Not because the maintenance is done very efficiently but customer pay a lot of it to keep it in the air.* |
| **[1.3.2]** | ***Learnability*** | • *What I'm telling is sometimes you do not know anymore what the coding of the system supposed to do.* |

# Participant 9

| | | |
|---|---|---|
| **[1.6.2]** | ***Adaptability*** | - *[Q] So you..can you categorize UTMS as a legacy system or not, in your opinion? [A] Auuuhh...good question...I think no...but that's more ..well it is now 10 years old, 10 years ago. So could it be legacy but the technolgoy isn't legacy right? So the technology is still used in 4G networks well.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | - *Systems that are where the development environment is out-dated.*<br>- *Not only the development part, we have to think about the architecture of the system also. Because if people are bumping against architecture, changing the earlier architecture.* |
| **[1.9.1]** | ***Old system*** | - *The alarm receiving center is the heart of the applications that surround it…I think 30 years old or so...* |
| **[1.6.1]** | ***Interoperability*** | - *That communicates with, via X25 connction. So that's all old stuff and then you have to have a legacy ssytem because nothing else can talk that stuff again.*<br>- *If you integrate application to another system, if you link application...You can do it on different ways. You can say well...I only have low level API and I just presented data by myself. Or maybe the data is already presented by legacy system or whatever system.*<br>- *A lot of application running aroud it the thing to ofload it adn that kind of thing, but it is OK...but if you look at harmony, this application is called harmony from [ORTECH] and we have a lot of probelm with harmony.* |
| **[1.5.1]** | ***Functional completeness*** | - *So the problem you are having at that point is that…that you are being limited in functionality by the legacy system.*<br>- *It [performance] is very poor.* |
| **[1.1.1]** | ***Modifiability*** | - *The effort to build the functionality is so high that is not a good business case. So if for instance we have to put in another data field in the system and that takes 4 weeks to program then yeah...you probably say well...sorry the system does not support that functionality anymore...sorry...you have to...we have to start modernization.*<br>- *Because legacy system tend to have bad maintanable code.* |
| **[1.13.2]** | ***Business Critical*** | - *Legacy system in the core of business.* |
| **[1.14.1]** | ***Expensive in maintenance*** | - *The amount of people who know that legacy system or the environment is build are limited. So that also the reason why cost more money. And probably in time you will have a problem with update of may be platform OS or those kind of things.*<br>- *But reason they don't...that is a lot of money, because their framework does not support...they do something strange with enabling the web part of that application. And change that is a lot money.* |
| **[2.7.1]** | ***Funding legacy modernization projects*** | - *It is very expensive to change it...still running fine so that's not a problem.* |
| **[3.1.1]** | ***Agile to business requirements*** | - *And the screening at the gate and the screening at the gate just schipol the only airport that does that...it is needed because the passeger flows mixed. So arriving passengers and departing passengers meet. And that's the problem so schipol is going to...* |
| **[1.13.1]** | ***Doesn't fit with future strategy*** | - *Legacy system does not facilitate them doing their business.* |
| **[2.7.2]** | ***Predicting Return of Investment (ROI) of modernization*** | - *For filling the business case.* |
| **[1.11.4]** | ***Complicated system*** | - *Legacy system also tends to hide complexity of businesss process.* |
| **[2.2.1]** | ***Lack of resource (e.g. documentation, expertise, etc)*** | - *You do not want COBOL programmer, but what do you want is somebdoy who understands COBOL and can extract functionality in logical way from the legacy system.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | - *There is data on how much ICT project is always overbudget and takes to much time right? so there is some problems in our management concern in ICT project.* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | - *[Q] when you do modernization do you this kind of phenomena. Like people in your...maybe in your team, they are don't really cooperate with the modernization process because they are the experts in the legacy system? [A] Yeah..that happens..*<br>- *So, the most important thing is, I think to make sure that the people you have in your company dont...aren't link to a spesific system but have another goal, maybe maintain multiple systems or build some other applications whatever.* |
| **[2.7.2]** | ***Predicting Return of Investment (ROI) of modernization*** | - *And the amount of change is going through the legacy system was not really foreseen at the moment we built it, we build the application. So there's some limitations in the applicaton itself.* |
| **[1.11.1]** | ***Large system*** | - *We have a big system running.* |
| **[1.8.1]** | ***Not flexible*** | - *And if...it's legacy system behind it then I can imagine, that you are less flexible presenting the data than newer systems.* |
| **[1.12.1]** | ***Knowledge become scarce*** | - *They get a beard and that kind of thing.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | - *It's a long project.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | - *I do not have a real trustwothty data for them...OK it is going to crash on 12 of November 2013 and after that date your business is gone. So that is not a case.* |

# Participant 10

| | | |
|---|---|---|
| **[1.13.2]** | ***Business Critical*** | - *I think the major issue about Legacy system is they are valuable.*<br>- *They have valuable business value...great business value.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | - *They do not fit into the way people want to work and how infrastructure work of today* |
| **[1.13.1]** | ***Doesn't fit with future strategy*** | |
| **[1.1.1]** | ***Modifiability*** | - *They trouble maintaining it, adjusting it.*<br>- *I think it will be harder and harder to change the system.* |
| **[3.1.2]** | ***Create new business opportunity*** | - *[XXX] is improving on the customer experience.*<br>- *Customers do not like it anymore. So the human experience, the user interface which is old fashion.* |
| **[1.12.1]** | ***Knowledge become scarce*** | - *Then you saw that only very few people who could understand how you could do that and...So that is…the system is legacy, but still there.*<br>- *People don't know the rules anymore because they never use them because the systems do the work. So there is no business knowledge anymore in the business.*<br>- *Nobodies know all the rules anymore which are in the system.*<br>- *Like Javanese, the high Javanese language, the core language, very few people understand it. And so that's if you're company, you have a system built in such a language which is of course we hard to find people to help you to maintain it and adjust it, then it is a problem.*<br>- *I think the big problem that you cannot find people to understand them and understand the technology.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.11.1]** | ***Large system*** | - *This legacy system have enormous amount of very complicated business logic inside.* |
| **[1.14.1]** | ***Expensive in maintenance*** | - *200million for maintenance.* |
| **[2.7.2]** | ***Predicting Return of Investment (ROI) of modernization*** | - *It is a lot of money...it is enormous of money. But would it be cheaper to make new system?* |
| **[1.11.2]** | ***Database issue*** | - *The database issue. They have same issue with database management and they want to renew that.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | - *We should do it so quickly that there are no changes almost. Somehow we have to compromise a little.*<br>- *If people start to change the system or the requirements of the system while you do modernization.*<br>- *Because it looks appealing to do more than only modernization. If you go with your car to the shop and have it fixed, and say...Well maybe you should paint it another color.*<br>- *[Q] Yeah...you change the decision in the middle of process. [A] Yeah*<br>- *The big puzzle is how to do modernization of this system while keeping it maintained.* |
| **[2.1.1]** | ***Data migration*** | - *If you migrate the data from the legacy system to the new system. Often that's also a big issue...it is even more complex than the writing the code.*<br>- *They have all the issues about legacy and all the issues about new system at the same time. but that's because data migration is very complicated.* |
| **[1.11.1]** | ***Multiple system*** | - *So they do the new business in the new system, and old system.* |
| **[1.6.1]** | ***Interoperability*** | - *People are still have trouble making the interface of SOA, simple and transparent that are services is easily used. Yeah....it is a big challenge.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | - *I think top management doesn't understand the issue and they don't give budget for it [legacy modernization].* |

# Participant 11

| | | |
|---|---|---|
| **[1.9.1]** | ***Old system*** | • *There is a lot of legacy systems that are 20 30 years old already.*<br>• *It [Legacy system] is an old system* |
| **[1.13.2]** | ***Business Critical*** | • *So it is an old system, it is an obselete system maybe but It is very useful and has a business impact still and generates a lot of revenue for banking and their clients, because it is still a lot of legacy system in the core system.*<br>• *No...good is good enough let's remain the leacy environments. Because then they can incerase their profit*<br>• *a lot of legacy system is the core system* |
| **[1.9.2]** | ***Availability*** | • *Normally, because there are not many changes on legacy system availability, they are very available, they are available, and they are more less 24/7 up and running.* |
| **[1.4.1]** | ***Time behavior*** | • *It is also characteristic about response time, it is very fast because normally it is very simple small data messages, so there is not that a lot of everhead.* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *More or less not that many knowledge of that system in organization because the people who are knowlegdable about this kind of system are already gone. they are already left buildng you might say. And documentation, so the characteristic also from old legacy system that is the lack of documentation, because it was never docuemted in those days.*<br>• *We have 100thousand people working for [XXX] in India. That is the biggest part of [XXX] is in India. But there is a lot of knowledge around there. So we have built up education around legacy transformation in India itself.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.1.1]** | ***Modifiability*** | • *It is hard to maintain, because beside there is no real updated software anymore and it is also not supported by a lot of other organizsaitons* |
| **[1.7.1]** | ***Unsupported supplier*** | |
| **[1.14.1]** | ***Expensive in maintenance*** | • *That is maintenances and because the maintenance is not there and if it is then it is the cost of maintenance, the maintainance is very high. So that will increase the total cost of the owneship of the system.*<br>• *Lisence cost of the software. that is also very expensive.* |
| **[1.6.1]** | ***Interoperability*** | • *The second one is a system integration. Becuase old date system has most of more or less have old applicaiton program interfaces. So there is more or less a lack of knowledge about integrating legacy systmes by itself.*<br>• *An other issue or challenge could be a system conversion also from software perspective.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | • *No because the…especially for organiation who are 24/7 in the air. It is very difficult than to have a shodow system besides the normal system because it must be up and running.*<br>• *Becaue you are changining in a running environment. So you need somewhere, somehow you need to determine OK…what is the best moment of the change.* |
| **[2.2.1]** | ***Lack of resource (e.g. documentation, expertise, etc)*** | • *It is complex, because you have a lot of different expertise needed to modernize such as people who understand database environment and operating systems, middleware, enterpriser services bus of architecture. You also need people who understand business functionality to transform business functions from legacy systems to another system."* |
| **[1.13.1]** | ***Doesn't fit with future strategy*** | • *Another discussion, because then you are not be able to say this legacy system can run for 5 or 10 years so it will...if it is not work properly anymore, it will disrupt business model of the organization.* |
| **[1.5.1]** | ***Functional completeness*** | • *That the older system cannot deliver what the new system can deliver, because the new system has also the capability of ananlitic and that could be also predictive analytic, so you have more functionality in the current new environtments around data.*<br>• *So how do you make from data information and how can you make from information derive intelegent and how can you base on that intelegent can ake proper decision.* |
| **[2.1.1]** | ***Data migration*** | • *Now you always see also see a lot of unstructure data, that is more difficult because 75% of the generated data of today is unstructured. It's texts, voices, movies, pictures. So that is an adapt...That is more complex to transform into anthoer because it cannot be stored in an hirarical relational database.*<br>• *The main risk in modernization is that the data migration, which cannot be done perfectly. Errors are made and you have some risk that your new system is disturbed after modernization.* |
| **[2.6.1]** | ***Cultural resistance in organization not to adapt new system*** | • *Sometimes people do not like changes. Not only in the business organizations, but also in IT organization. So you need to persuade them for the need of transformation [modernization].*<br>• *Sometimes they start the transformatino, they forget to educated the people.*<br>• *Change that the organizaiton is ready for the change. Is the culture in the organizaiton, do they allow a change for that kind of different information.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *So you need to persuade them, the why and the how for the transformation.*<br>• *Communicate and infrom the people who are responsible for transforming, so in programitself, but even more important is to communicate and infrom the people outside the program.* |
| **[2.7.2]** | ***Predicting Return of Investment (ROI) of modernization*** | • *Creating a business case. are you capable to describe in financial perspective what are the revenue? what are the cost and what are the Return On Investment.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | • *The third one is of course transformaiton plan itself. What steps will you...do you need to prepare to govern the whole transformation plan by itself. So you need to very good project manager.*<br>• *How do you organize and how do you control all the transformation. Also take into acount availability.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | • *That has everything to do with I think the architectural technical complexity.* |

# Participant 12

| | | |
|---|---|---|
| **[1.7.1]** | ***Unsupported supplier*** | • *I would define a legacy system as a system which is constrained by hardware, specific hardware which is no longer fully supported or being very scarce.*<br>• *If the legacy system depends on specific hardware, then if you cannot stock up on that hardware, then you completely relying on its functionality. If it is break down, then you do not have a choice and it is too late.* |
| **[1.1.1]** | ***Modifiability*** | • *It maybe constrain by technology, for instance it is built on technology that is no longer maintained.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | • *Or it may rely on specific people because lack of documentation or complexity of the system.* |
| **[1.11.4]** | ***Complicated system*** | |
| **[1.12.1]** | ***Knowledge become scarce*** | • *Mostly I do not see a lot of documentation on that work*<br>• *Many companies now are very relying on very specific persons and they do not really like that because you cannot control when the person leave the company. And sometimes they even old enough to nearly retired.*<br>• *it can help if teach student what is the value of the legacy systems and how to work with that systems.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[2.7.1]** | ***Funding legacy modernization projects*** | • *If they modernize it now they have to completely reinvest.*<br>• *They mostly do not want to do that.so if they choose to completely rebuild, it will be major investment. Many man years project* |
| **[1.13.2]** | ***Business Critical*** | • *Core in the system where is very difficult to find out what are exactly other rules that are implemented.* |
| **[1.2.1]** | ***Robust*** | • *They have been around for many years and during this period they have been stabilized.* |
| **[1.5.1]** | ***Functional completeness*** | • *Simplicity it has, for instance user interface, very simplistic because it keeps out all the unnecessary stuff.*<br>• *And sometimes they simplicity of the chosen languages or technologies also make it very stable, because it is not possible to add all these kind of feature during the years.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *[Q] DO you think that the maintenance of the legacy system is expensive? [A] Yes I think so...*<br>• *The goals to remove those people from the payroll and of course they doesn't really help with their involvement in the project.*<br>• *It can be desire to reduce cost, for instance by moving to standard software.*<br>• *But if you can move to standard product, then it could be usually an advantage because the maintenance cost for standard product is usually lower."* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *It's really difficult, but you can only do it when it also part of the assignment. because sometimes they don't really feel a need to change the culture.*<br>• *Not difficult to find him but sometimes it is quite difficult to work with them.* |
| **[2.6.1]** | ***Cultural resistance in organization not to adapt new system*** | • *Sometimes it can be a bit difficult and to let that go and start working on new technology and new system.*<br>• *Sometimes they see legacy systems as their baby and they tend to know every aspect of it. Sometimes it is dicult to work with them while modernizing because they might not share their knowledge.* |
| **[2.3.2]** | ***Difficult to test*** | • *It is usually testing.*<br>• *Because that difficult also difficult to get your test case complete*<br>• *It can be diffcult to extract all the use cases to test. It also requires a lot of work to compare functionality from legacy system to the new application. It can be time consuming and difficult work.* |
| **[2.5.1]** | ***Difficult to extract business rules or knowledge*** | • *It's more challenge for…to get requirements very clear and complete.*<br>• *Help to improve the insight into internal procedure of this legacy system and not in the very technical way.*<br>• *So I think it [business rules extraction] can be really helpful to provide insight into the internal working of the system, to extract it to a* |

| | | |
|---|---|---|
| | | *human readable diagrams or documentation.*<br>• *The company or the project team has to extract exactly the internal functionality of this legacy application. It can be difficult to extract, to document, and to implement it [business logic] properly* |
| **[2.1.1]** | ***Data migration issue*** | • *A challenge can be to migrate the data. For instance to convert from specific encoding or data structures or specific storage types to a new database.*<br>• *During the process of migrating you have all kind of data quality issues.*<br>• *You want to migrate it completely with all its problems or do you want to improve on the quality as well? But if chose improve on the quality you may end up with the whole new project on your hand as well. Because improving data quality is...can very...well...be large project on its own.* |
| **[1.11.2]** | ***Database issue*** | • *Because this system, they have been running for many years they tend to be some dirty data inside it.* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *They can be satisfy with the system, but they can be very dissatisfy with the risk building up offer the years.* |
| **[1.10.1]** | ***Risk of running legacy system*** | |
| **[1.10.1]** | ***Risk of running legacy system*** | • *The risk can be a very strong driver.*<br>• *But for the insurance they really take care of the risk because they can't afford if their system goes down for like one hour.*<br>• *It can be to mitigate the risks, they see it as an operational risk when staying with the current.* |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | • *And it will take many months or sometimes even more than one or two years for the project of modernization to complete.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | • *Because if it takes that long the world is changes during the project. So you have to adjust to that. If you don't do you deliver, you always deliver late. Because you deliver on specification which 1,5 year old.*<br>• *And during that period they will still need the old system. Mostly you will see they will keep running their original system, sometimes even keep performing maintenance on the original system because of the project of rebuilt takes that long, they cannot afford to stop on maintenance.* |
| **[1.8.1]** | ***Not flexible*** | • *They expect more flexibility when thye want to change their busness process.*<br>• *They expect more cheaper and faster ways of implementing those changes.* |

# Participant 13

| | | |
|---|---|---|
| **[1.9.1]** | ***Old system*** | • *Legacy system for me is a combination of hardware and software which was buit long time ago.* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *It is usually built or installed or configured by someone or somebody who is not around anymore, so lacking knowledge.*<br>• *[Q] So knowledge scarscity is a problem in the future. [A] Yeas..yeass..yeas...*<br>• *The issue is that there is less knowledgeable people [experts] are available in the organization because the [knowledgeable] people of the system are already gone [left job]. The [other] characteristic of legacy systems is the lack of documentation."* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[2.7.1]** | ***Funding legacy modernization projects*** | • *It usually a lot of time and energy and money and research involve to get rid of it.*<br>• *So it is a lot of invest. It cost them a lot of time.*<br>• *Basically a lot of things have already said yeah...so it cost a lot of money and alot of time. basically you can only invest euro once.* |
| **[3.1.2]** | ***Create new business opportunity*** | • *OK now it is time to stop that and we going to move to the new world. Like merger or acuistion or reorganizaiton or something like that.*<br>• *Natural moment. so basically if something else happen then which forces you to invest in the application anyway, then migrate it to a new platform.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *So simply the skill is not available, the only way to by pass that is that to hire quite expensive, dedicated consultants.*<br>• *It is getitng more and more expensive because risk are getting bigger and also knowledge is getting scared.*<br>• *You have to hire somebody quite expensive to make your legacy world SEPA comply.*<br>• *Maintenance cost are getting higher because knowledge is getting scarce and risk of failure are growing.* |
| **[2.2.1]** | ***Lack of resource (e.g. documentation, expertise, etc)*** | • *You do involve the busenesss, the user, the owners, to test, to monitor, to see if the application is working.* |
| **[3.2.1]** | ***Reduce cost of maintenance and operation*** | • *We want to do move toward a standarized way of working in the entire company. So standarize from business process point of view.* |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | • *The business impact is always the bigger ones. An it is usually just sinple time and money.*<br>• *Your biggest problem is an availability of resources [documentation and experts] and availability of money and [to some] extent availability of time.* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *Usually, the people who use the system are satisfy with the system.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | • *Basically, what you need to have is you need to have strong governance model. You need to have one clear set of rules, who make a decisions and what the impact and magnitude the decision can be.*<br>• *Yeah...so you always have to make sure from ok...if I do a migration I always have to do lot of check and balances. I have to do yea...I have to do comply check...I have to do assessment, I have to do...I have to make sure that everything works, including the people.* |
| **[1.10.1]** | ***Risk of running legacy system*** | • *Basically the risks could be the legacy system falls down, our production process is a stopped.*<br>• *Continuity. That the biggest problem. Because it is old it will fall down.* |
| **[1.1.1]** | ***Modifiability*** | • *Becuase it is old and the knowledge is not available, no body can do the adjustment something like that.* |
| **[1.8.1]** | ***Not Flexible*** | • *Implement a changes coming from your environment like the SEPA project or couple of years ago the implementation of euro with stuff like that and you have one system, you can do it quite fast. If you have to do that same change in 20 or 25 systems, it is a hell of lot more work and therefore hell of a lot more cost.* |
| **[3.3.1]** | ***Prone to failure*** | • *From the cost avoided point of view to have your environment up to date and integrated and standarize as much as possible. In the end that's a chepest way run your IT.*<br>• *We have an old ERP system, old almost 10 years old. And it drives the production in the plant and also the logistic and warehouse and also the order towards the customers. If that system stops, the plant stops, the warehouse stops."* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *Somebody quite high at the top. somebody who can enforce decision where they need to be enforce. and sombody who can also oversee the impacat of going left - going right.* |
| **[2.3.2]** | ***Difficult to test*** | • *So there's always human work and alos always room for human error.*<br>• *No...well...testing sometimes. But that is it. For now and then we use every known...And we had use a script to for instance to deploy an image an applciaitons on a virtual server and stuff like that.* |

# Participant 14

| | | |
|---|---|---|
| **[1.9.1]** | ***Old system*** | • *Let's call it a system which is older.* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *You have less knowledge about.*<br>• *And you notice that not everything is documented properly.*<br>• *In meantime we ask now the people please document your maintenance procedure, please document your recovery procedure.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.13.1]** | ***Doesn't fit with future strategy*** | • *Does not fit let say your future roadmap.*<br>• *But you know you want to get rid of it in the future.* |
| **[1.13.2]** | ***Business Critical*** | • *Legacy for me is let say business critical.* |
| **[1.2.2]** | ***Stable system*** | • *Legacy by itself it is stable.* |
| **[1.4.2]** | ***General Performance is good*** | • *The performance is enough.* |
| **[1.10.1]** | ***Risk of running legacy system*** | • *So it is more about continuity of the legacy system where the risk is.*<br>• *Can mean the risk of higher cost, can mean risk of losing people that go away that you don't have knowledge.* |
| **[1.9.2]** | ***Availability*** | • *It has proven, it is stable but it is also has proven.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *if you look at the [maintenance] cost, I'm quite sure I can run a similar environment against lower cost, if I would use a standardized product. Let's say state of the art ERP environment...because I don't need somebody to maintain.*<br>• *So I'm quite sure I can make it cheaper but that means you have through 2 or 3 ERP implementation globally to implement a new ERP system and then take out ...let say... those IT staff. And get rid of the IT staff.* |
| **[2.7.2]** | ***Predicting Return of Investment (ROI) of modernization*** | • *Are you willing and then you have to calculate the risk of the how if it can be happen. And then you make a decision.* |

| | | |
|---|---|---|
| **[3.2.1]** | ***Reduce cost of maintenance and operation*** | • *Standardize means no customization anymore, if I want something extra to the system, either it must be configurable or the vendor deliver that needs of new functionality and its part of the new system.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | • *So one of the challenges is how do you migrate all your interfaces which are point to point instead of SOA.* |
| **[1.6.1]** | ***Interoperability*** | • *Now interfaces last year we are now developing more in the SOA way which means the new interfaces are easier to connect to the new ERP. But the old point to point that would be difficult how that can be solved.*<br>• *We have a huge set up of interfaces. ERP is the core of our company almost every system talk, synchronize or asynchronize with the system.* |
| **[3.1.3]** | ***Faster time-to-market product*** | • *Business driver is time to market.* |
| **[3.1.2]** | ***Create new business opportunity*** | • *What I see is our business opportunities and our legacy cannot meet the pace for make new development.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | • *So have a good project... program...how do you do the program management? How do you do the change management in project?*<br>• *Proper governance in those kinds of project is crucial in my opinion. And that should be at let say at the right level of the organization.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *And there is on one hand a helping with people, explaining to people, we want to different, explain why we want to do it?* |
| **[2.6.1]** | ***Cultural resistance in organization not to adapt new system*** | • *Because people have really get used to it that they cannot walk.*<br>• *I want to prevent that I develop something now here that does not going to work in UK or US. Or whatever the location. So the culture here is a people are used of just doing that bilateral.* |
| **[2.1.1]** | ***Data migration*** | • *So if your old system, one of the problems here we have is data quality.*<br>• *Strongly believe if you are doing migration of your legacy, you have to prepare a good data migration strategy.*<br>• *And preferably, as much as automated as possible.*<br>• *To speed up the process, but also for cost. I think for automatic data migration, especially when you go.*<br>• *If you are doing migration of your legacy, you have to prepare a good data migration strategy.* |
| **[2.5.1]** | ***Difficult to extract business rules or knowledge*** | • *Having good requirements.*<br>• *Requirement before and requirement after.* |
| **[1.7.1]** | ***Unsupported supplier*** | • *We have to ask the vendor please develop that part in your software and make it as a standard feature.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *Like the managerial part also about those bigger legacy system have also impacted.* |
| **[1.8.1]** | ***Not flexible*** | • *Can make it pluggable like plug and play by having SOA environment available* |

# Participant 15

| | | |
|---|---|---|
| **[1.8.1]** | ***Not flexible*** | • *Is it flexibele enough to be changed in order to support the future requirements? So that's flexibility.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | • *Is the technical architecture of this applications still up to date and still supported for the near future?* |
| **[2.1.1]** | ***Data migration*** | • *The larger system though, the larger back office systems, take some time, because you need to convert data.*<br>• *The data transformation cannot be done perfectly. Errors are made. so you have some risk that your operation is a little bit disturb after transformation.* |
| **[2.7.1]** | ***Funding legacy modernization projects*** | • *This is time consuming, and costly.*<br>• *Transformation of old system to a new system is costly since you need to test it very carefully. And testing is costly* |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | |
| **[2.3.2]** | ***Difficult to test*** | |
| **[1.6.1]** | ***Interoperability*** | • *Legacy system typicaly are old in a integration with the internet. They do not support integration with the internet.* |
| **[1.5.1]** | ***Functional completeness*** | • *[Q]So you expect your target system when you do modernization would be more beneficial than your legacy system?[A] Yes, absolutely.*<br>• *Storing data in a structured way, but not supportive in lot of calculations or validation of the data.* |
| **[2.2.1]** | ***Lack of resource (e.g. documentation, expertise, etc)*** | • *Really you need people who undrestand the old system in order to transfrom the data to the new system. And those people are usually quaite scarce.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *Data trasnformation is not a sexy subject.*<br>• *The good news is in this company we have make it joined target. So business and IT, top management have a join target in decreasing the number of applications.*<br>• *I believe but that is a success factor if you make it talk of the time by top management, it helps tremendously.* |
| **[3.3.1** | ***Prone to failure*** | • *So that on point of agility. I think having less applications. Make sure that cost probably run down, and make your company more agile and more easy to be change to the future requirements.*<br>• *If you have application landscape which is to complex, you need to simplify it, otherwise you are not agile to the future.*<br>• *We need to implemetn as an insurance company, quite a few legal requirements. If you dont...if you have a very complex application landscape, you need to implement all legal requieremetn in all system. If you have less system, tehn it easier and costless.*<br>• *It should be a driver, and if company are smart, they use it as a driver.* |
| **[3.1.1]** | ***Agile to business requirements*** | |
| **[1.11.4]** | ***Complicated system*** | |
| **[1.12.1]** | ***Knowledge become scarce*** | • *So then people start to scarce, but maybe not in the language, but more on the content on that applcaition.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.11.4]** | ***Complicated system*** | • *Since most of the products were very complicated, and therefore a quite a few point solutions.* |
| **[1.5.2]** | ***Functional appropriateness*** | • *Is it functional supporting the current requirement?* |
| **[1.2.1]** | ***Robust*** | • *So it is robust for the future.* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *So that can be a reaosn why peole are a little bit less reluctant.* |
| **[3.2.1]** | ***Reduce cost of maintenance and operation*** | • *Ideally you have standard interface between each of these compartment of each functionality. And SOA helps with that.* |
| **[3.3.1]** | ***Prone to failure*** | • *If you have separated functional areas, you can also separate maintenance. This is the way of managing complexity.* |

# Participant 16

| | | |
|---|---|---|
| **[3.4.1]** | ***Lack of supplier / vendor*** | • *End supported or end of life by the supplier.* |
| **[1.6.2]** | ***Adaptability*** | • *It's not compatible in our IT environment.* |
| **[1.11.1]** | ***Multiple system*** | • *The fact that lot of time we replace application but the old application is still running.* |
| **[2.7.1]** | ***Funding legacy modernization projects*** | • *Not enough budget.* |
| **[1.10.1]** | ***High risk of running legacy system*** | • *Big risk becasue you dont have any support of your supplier.*<br>• *No..no...because of the risk and unsupportable.*<br>• *If it going down, it is really going down. And it takes a lot of effort to get it up and make it runs again.* |
| **[1.2.2]** | ***Stable system*** | • *On legacy, we don't do any changes anymore so sometimes it is stable.* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *We still have people with knowledge of legacy system. But what we see is that the knwledge is walking out because people are moving to other job. So it is getting harder and harder to get the knowledge.*<br>• *Lack of knowledge.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[3.1.2]** | ***Create new business opportunity*** | • *But it is not good for my customers Resistance from the current users or maintainers in the organization.* |
| **[1.8.1]** | ***Not flexible*** | • *On of the other things why we want to get rid of the legacy because it is very…it is not flexible.*<br>• *It's regarding interoperability..hmm...it is not flexible...*<br>• *One other point is that my costumer wants flexibility.*<br>• *Legacy it is not flexible and the legacy systems are hidding some secret that we not always know.* |
| **[1.11.1]** | ***Large system*** | • *Because legcacy is rigid, is big.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *We now migrate it to dotNET and our lisence fee is drop for something like half.* |

| | | |
|---|---|---|
| **[2.2.1]** | ***Lack of resource (e.g. documentation, expertise, etc)*** | • *Sometimes we are the first example in the world* |
| **[2.6.1]** | ***Cultural resistance in organization not to adapt new system*** | • *But also the state of mind of the people.*<br>• *And the awareness with our end users that we have legacy and we have to get rid of it.* |
| **[2.1.1]** | ***Data migration*** | • *Mising data.* |
| **[1.1.1]** | ***Modifiability*** | • *I think the system is unrepairable in the future because of the lack of the parts of the machinery and because of the lack of the knowledge regarding of the software.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *I sometimes think that if you have dedicated team. Dedicated focus on get rid of legacy, maybe it would help.* |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | • *The timeframe to test.* |
| **[3.1.3]** | ***Faster time-to-market product*** | • *Short time to market.* |
| **[1.6.1]** | ***Interoperability*** | • *It is regarding interoperability…hmm...it is not flexible.* |

# Participant 17

| | | |
|---|---|---|
| **[1.9.1]** | ***Old system*** | • *It is old.* |
| **[1.7.1]** | ***Unsupported supplier*** | • There’s no patches. If suppliers stop their product, organization needs to find another way to keep supporting their system. |
| **[1.5.1]** | ***Functional completeness*** | • *It is not processing any more.* |
| **[3.4.1]** | ***Lack of supplier / vendor*** | • *Because you need support, you need update the patches, you need security patches, you need everything to get the stable situation* |
| **[1.11.1]** | ***Large system*** | • *That was a really big.* |
| **[1.9.2]** | ***Availability*** | • *It is proven technology most of the time.*<br>• *Most of the time it's [a] proven technology. AS400 is stable, it always works [24/7] and is quite good. So, it's proven technology and normally it's stable, [which] is a good thing."* |
| **[1.2.2]** | ***Stable system*** | • *It was stable, it works almost always...hahaha...But it was quite good.*<br>• *They have been working- why x it?"* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *Yeah...actually higher, because it is not a standard anymore.*<br>• *But the maintenance cost of the AS400 was sky high so we need to get rid of the system and really turn it out.*<br>• *[With legacy systems] the cost is getting higher because maintenance is getting more expensive, [then] maybe you should think of modernization.* |
| **[2.1.1]** | ***Data migration*** | • *So biggest problem was how to update those data? So how you get the good information and get the right information out of the system into the newer system and leave all the old system behind.*<br>• *Data migration is really difficult because legacy system doesn't support modern databases or doesn't have relation database model.* |
| **[3.1.2]** | ***Create new business opportunity*** | • *Bank is too big, you have to split it because otherwise the risk is too high if it is fall apart and the governance has to carry all the consequence.*<br>• *It was initiated by the change of ING. We have to change then start modernization. So there was in need for changing the modernization, there was a need for changing.* |
| **[1.6.1]** | ***Interoperability*** | • *The technical difficult was all the subsystem of the legacy system. So there was something for printing, there was something for reporting, there was some external connection for getting all the messages, there were so many system and connectivity around the legacy system.* |
| **[2.2.1]** | ***Lack of resource (e.g. documentation, expertise, etc)*** | • *If you are the first, I think you will have a risk.*<br>• *Biggest risk is if you go to the new system that hasn't any proven technology or it is really new.* |
| **[2.6.1]** | ***Cultural resistance in organization not to adapt new system*** | • *We also change the culture of people and how they should work and report and everything. I think it will take 3 years to get that kind of transformation.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *One of them is communication and get business people involve in the project. let them see here and know what's you are doing. So I think a best success factor is you know K-O-T-T-E-R.*<br>• *We did a lot of communication. Several time a week we post everything need, what a new strategy was. And that was about communication.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *To convince them and get the migration and accept the new system. I think that's the hard part of it.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | • *Also the business strategy itself change several time during the migration. So we also made some changes during the migration.*<br>• *I don’t think that you really have some if you have a good plan. We already had a good structure, we already had a new system and we already knew what we are going to do.*<br>• *I think first of all, you should start with the manager. If you give people much room to complain or do other things and you do not support them in how they should work and being an example. I think you have to being an example as a manger how you should work and how can work and if it is not working, you are not start to complaining, you try to fix it and help.* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *Actually one of them didn't apply to a job at my team, because he already saw he didn't have work anymore after the migration of AS400.*<br>• *We do have a plan, we start but to do more training, do small implementations.*<br>• *No they're not satisfied. But I think in IT you do not really have satisfy end user because if the system does not work for 5 minutes, they are already frustrating because they need to do something in those 5 minutes.* |
| **[1.1.1]** | ***Modifiability*** | • *Upgrading and getting to the higher level for supporting the organization in your business, your legacy system is getting more difficult.* |
| **[1.13.1]** | ***Doesn’t fit with future strategy*** | |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *Like we have a good system, why should you change the system when everyone is happy with it.* |
| **[1.11.2]** | ***Database issue*** | • *Normally legacy system doesn't support very good databases or does not have relation database model.*<br>• *There was the data in the system, it was old, it was not update.* |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | • *We were in the time squeeze. We got only a few months to do the conversion and we didn't have any time to did very good research.* |
| **[2.7.2]** | ***Predicting Return of Investment (ROI) of modernization*** | • *I think maybe the acceptance of the business people that they won’t accept the newer system or hard to change.* |

# Participant 18

| | | |
|---|---|---|
| **[1.12.1]** | ***Knowledge become scarce*** | • *Nobody really knows about it, but see there's reason for not to touch it.*<br>• *They do not know what the hidden features are behind or what the business rules.*<br>• *The problem with the language like C is that there's no...less knowledge in the market right now.*<br>• *It is not documented very well.*<br>• *For the old systems, we reverse engineer the old system and document them well so we know what the requirements are and can help to build new system.*<br>• *What we trying to do now is, [XXX] use the document standard the JSTD. It's a...right now it is a IEEE standard, but this is the old version coming from the defense industry but IEEE is a newer standard* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.6.2]** | ***Adaptability*** | • *Yes because it doesn't work in windows. It works on windows 8 but not really... doesn’t' really support all technology.* |
| **[1.7.1]** | ***Unsupported supplier*** | • *Hardware that are not support anymore...environment.* |
| **[1.9.1]** | ***Old system*** | • *Legacy system are system that were built long time ago.*<br>• *The performance is not really good.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *One is getting very expensive to maintenance the system.*<br>• *You can just say goodbye to few people.*<br>• *From [XXX] is money. it is all money and to reduce the cost of this thing and maintaining the IT.* |
| **[1.2.1]** | ***Robust*** | • *It is reliable...people know how to use it. All the problems have been xed over the years from it. So technical problems are usually not there".* |
| **[1.5.2]** | ***Functional appropriateness*** | • *But it is working so everyone keeps it that way because it is doing its things.* |

| | | |
|---|---|---|
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | - *We have to spend so much time to do the extra things that are not in legacy system.*<br>- *There's we cannot take few people and just keep them there.* |
| **[1.11.4]** | ***Complicated system*** | - *The one I know is pretty complex. The one I use for the IT contain terminal.* |
| **[1.6.1]** | ***Interoperability*** | - *But they have an old legacy system which is very hard to connect to that bus.* |
| **[1.3.1]** | ***User interface aesthetics*** | - *Interface is pretty clear that's just a messages like XML but in the old version.*<br>- *Clean and clear interfaces.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | - *They have cargo, they have to take cargo from there to there. That is their view. Sometimes you have to drop something off and pick something up. But the government wants to use it for traffic management. So very another way of looking at the data.* |
| **[3.1.1]** | ***Agile to business requirements*** | - *If something happen and then there is…they send new regulations...the one of the things are changing over the time the way they have to handle the danger goods.*<br>- *And that's calculation is changing every 2 years.*<br>- *Dutch government wants to do is to promote the traffic by the river. And then off by the road, promote traffic jam etc.*<br>- *For the government, change in regulation.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | - *Because the program is really hard-coded, it is not congurable that makes modernization difficult.* |
| **[1.3.2]** | ***Learnability*** | - *So then usually takes few months for someone anew to learn the system or old system.* |
| **[2.5.1]** | ***Difficult to extract business rules or knowledge*** | - *Yes, you can only do that if you know exactly what the thing does. That is [business logic extraction] is the hard part.*<br>- *We try to get the functional requirements very clear, what should the new system do.* |
| **[3.4.1]** | ***Lack of supplier / vendor*** | - *Not only because it is old technology.* |
| **[3.1.2]** | ***Create new business opportunity*** | - *To make more use of the information the customers are given. Because in this process in translation, some information also lost. We do not do it, because we do not really know what to do with it, so just throw it away.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | - *So we really assure communication part with the end user.*<br>- *People who think they know better. Because they have read a book about SOA and wooo...it should be SOA.*<br>- *So there is a lot of confidence and emotion.* |
| **[2.7.1]** | ***Funding legacy modernization projects*** | - *The ROI is too long.*<br>- *And a lot of companies also say they have their return of investment. They say it can be 1 year and then it should be done. well…I do not think return of investment of 1 year can...*<br>- *Big financial given by the governments to make some modernization.* |

# Participant 19

| | | |
|---|---|---|
| **[1.13.1]** | ***Doesn't fit with future strategy*** | - *My definition of a legacy system is systems and technologies that do not belong to your strategic technology goals.* |
| **[1.6.2]** | ***Adaptability*** | - *Very hard to keep it up to date, to make...to move that forward to newer versions.*<br>- *I only call legacy system if you are stuck in the technology that you dont wanna carry forward.* |
| **[1.1.1]** | ***Modifiability*** | - *Now that is the legacy. You will not find anyone who can maintain that.*<br>- *It is always new rules, regulations, common labour agreement, and thing that changed between taxes, so that's always somthing changes. So there, legacy is really painful.* |
| **[1.12.1]** | ***Knowledge become scarce*** | - *It's a product that we still maintain as we go forward but for us it is very hard to find new UNIFACE developer anymore.*<br>- *But how many you can find that who can understand really crypthographic algorithm where youcould do all kind of bit shifting, hasing and those kind of things. That's very limited group.*<br>- *Undocumented feature or knowledge, things in the product that no one knows about or spesific feature that works forever and that have been maintained recently and the one that invented it has long left the company.*<br>- *[Q] SO you think important for the company to educate people in COBOL nowadays? [A] Not yet, but it will be in 10 years. Because then you get distraction when so much COBOL code stil around and you need people people to maintain it. And to do that forward.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.5.2]** | ***Functional appropriateness*** | - *It works as expected. Even if has bugs, people know the bugs, people have all worked around.*<br>- *Because their expectation are completely inline with what they are getting.* |
| **[1.2.2]** | ***Stable system*** | - *Then good things about a product, a program that you dont maintain anymore is that stable, and it gets faster.*<br>- *So the perception is, it is becoming more stable. It's not actually more stable, because still bugs in it, but everyone knows about the funny things, how to fix that, how to work around it and it's predictable system to everyone.* |
| **[1.4.2]** | ***General Performance is good*** | - *So the performance as expected.* |
| **[1.4.1]** | ***Time behavior*** | - *Because hardware, mono-hardware alwasy get faster.*<br>- *Old code went as amazing speed on new hardware.* |
| **[1.3.2]** | ***Learnability*** | - *Yeah..they know that they have to not touch certain key or if you are in that screen dont touch that button because then your screen will die. Or if it goes wrong just do this step and then you back on tract.* |
| **[2.6.2** | ***Resistance from the current users or maintainers in the organization*** | - *And most people very happy.* |
| **[2.1.1]** | ***Data migration*** | - *You always have to suffer from migration, incomplete data, lost of data, bug still in the system, not know how it works.*<br>- *Big challenge is always data migration.*<br>- *Mainframe you have those back decimal, how you store them on your databases, so that's all kind of spesific in the data migration that is hard.* |
| **[2.6.1]** | ***Cultural resistance in organization not to adapt new system*** | - *They like they safety zone and they only use the system because they have to get the job done and get how at 5 o'clock leave the office. go watch football..something like that.*<br>- *I think last one is a the human factors. Developers that dont wanna change.*<br>- *You have to somehow come up with the business case that says what is my current cost, what is the cost of migration, what the new total cost of ownership, and that you have to predict the Return of Investment. A business case can have soft components like improve maintenance or improve performance because they represent business value.* |
| **[3.1.1]** | ***Agile to business requirements*** | - *If they come up wiht new kind of rules, of regulation and you have to come up with additional text then it has the deferent way of calculation, you have to go in and build new type of rule.*<br>- *Say everyone has to pay this amount of text except for single mothers with children under 5, single retired people over the age of 50 or 70, people like this, peole like that.* |
| **[1.7.1]** | ***Unsupported supplier:*** | - *If the technology supplier is not support it anymore, then you also have a risk.*<br>- *So, even [XXX] stop suportting it, then you are stuck because then you get a new hardware..you dont have a driver anymore for the chipset or, so you have to figure out how can install the platform and install the application in the platform.* |
| **[1.14.1]** | ***Expensive in maintenance*** | - *So prices goes up, because they have, basically they have so much capasity and for maintaining it, it cost them thier money.* |
| **[3.4.1]** | ***Lack of supplier / vendor*** | - *Where the product was at the end of the product life cycle.*<br>- *Continuity. Continuity is the biggest issue that I have becuase if..and it can be..because you can find the developers anymore. you are stuck. if the technolgoy is not support anymore by a provider.* |
| **[2.5.1]** | ***Difficult to extract business rules or knowledge*** | - *It is running code, so you can always use the application and test it, to see what it does. So from the functional point of view since it is a prototype product, it is not in house system that no one know about, it is just a common product, it is very clear on what it does for many costumers. So that helps.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | - *But usually systems are so closely integrated and complex.* |
| **[1.6.1]** | ***Interoperability*** | - *Technology incompatibility. So you have rack in mainframe and you have rack in windows, and still rack script that one on mainframe will not always run on windows, because implementation, slightly different.* |

# Participant 20 and Participant 21

| | | |
|---|---|---|
| **[1.13.2]** | ***Business Critical*** | - *And we love our customer legacy system. Because as a company, we are successful because of all differences between legacy systems. Because if all legacy system talk and communicate with the same protocol, then we would not exist.*<br>- *So the difficulty of supporting legacy is one of the cores of our business.* |
| **[1.6.1]** | ***Interoperability*** | - *Because normally these systems are not talk each other.* |
| **[1.1.1]** | ***Modifiability*** | - *Can we do maintenance on the system? And if it became difficult, then it is a legacy.*<br>- *There are people who know about it and interact with it, but not how to modify it. So we are very limited in ability to modify it and to* |

| | | |
|---|---|---|
| | | *add new functionality and we don't do that anymore.* |
| **[1.10.1]** | ***Risk of running legacy system*** | • *It is always about risk right? You have to balance risks. If it big risk to maintain in current state...well risk could be economical risk or technical risk...if the risk become so big, then we should consider it as a legacy.*<br>• *We also have legacy but we do not get rid of it, because it has no big risk yet. But we still consider it legacy.* |
| **[1.13.1]** | ***Doesn't fit with future strategy*** | • *Some of the engine we consider it is legacy, but we still use them, we just do not enhance them anymore or add new feature to that.* |
| **[1.11.1]** | ***Multiple system*** | • *We use other engine and they are newer generation engine before that. But we still maintain the old one.*<br>• *For compatibility reason because in ecosystem around us, some parties is still using the old format.* |
| **[1.11.4]** | ***Complicated system*** | • *There is so much rule and information in that engine.* |
| **[2.5.1]** | ***Difficult to extract business rules or knowledge*** | • *To deal with all kind of exception. To work around and cooperate with other system.*<br>• *To extract all the rules and details in there [legacy systems] is really dfficult.* |
| **[1.3.2]** | ***Learnability*** | • *If you come here as a new developer, it would be very difficult to do anything inside the piece of the system.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | • *And extra handicapped is we are 24/7 highly reliable, up and running, in ecosystem. So it is very difficult in the situation like that to just take a piece out and refactor it and put the piece back in without any down time.* |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | • *The legacy system still there and we are facing it out, but it is going to take a long time.* |
| **[1.2.1]** | ***Robust*** | • *So the good think is it is still running, it is working, it is petty reliable, pretty good.*<br>• *We keeping them there because it is performing well.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | • *The priority can change quickly.*<br>• *How to find a good migration path* |
| **[3.1.2]** | ***Create new business opportunity*** | • *New opportunity, if you think the market changes, new message format or new way of connecting.* |
| **[3.1.1]** | ***Agile to business requirements*** | • *There is a government body called NICTIS, who decide the standard for messaging. And that is one driver if they upgrade the standard, then we have to migrate too.*<br>• *Other point is that my costumer wants flexibility, and a short time-to-market, then you have to get rid of your legacy. Because legacy is rigid, and it is not flexible* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *One of the factors for maintainability is also documentation. So it that lacking, then it is much more difficult for new developers to learn how things work. a huge factor.*<br>• *If documentation is lacking then it is a bigger risk to migrate. Because you don't know what is going on in the old system and the risk to migrate is bigger.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.7.1]** | ***Unsupported supplier*** | • *Whenever the platform the software run on is outdated, so we cannot then support what so ever, I mean our server level are very high.* |
| **[2.7.2]** | ***Predicting Return of Investment (ROI) of modernization*** | • *Who looks further than current version and who look at new techniques and new frameworks an new things like SQRS or big data or who looks further than current system and spot to the technique that are useful to migrate to*<br>• *They [Top management] are always looking for a short term Return of Investment. Once you put the money in, they want to earn it back."* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *Can everybody handle it? Can we explain good enough? Can people learn it? And to use new technology, we have to have skill to use it. We have to invest in it for a longer time to get the experience.* |
| **[2.3.2]** | ***Difficult to test*** | • *Regression test that cover enough all the continuality.* |

# Participant 22

| | | |
|---|---|---|
| **[1.13.1]** | ***Doesn't fit with future strategy*** | • *Legacy is what I would get rid off. Legacy is kind of what hindering me.* |
| **[1.8.1]** | ***Not flexible*** | • *No it is not flexible enough to do the things you wanna do.*<br>• *They cannot reponse to change request.* |
| **[1.1.1]** | ***Modifiability*** | • *Because you cannot attract people that they have to maintain it or it does not cooperate new kind of technology.*<br>• *I think they have to maintain to many knowladge area, COBOL, Java.* |
| **[2.7.1]** | ***Funding legacy modernization projects*** | • *It is too expensive to replace them.*<br>• *Now I think 5 years later we still are not able to say that we replace the old system. I think we have estimation about 30-40million euros still to spend to get rid of the old crap.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | • *But that doesnt work. Because evry time the attention goes away from modernizing to functuinality.*<br>• *Getting it done.. That's really our challenge.*<br>• *Keeping pace between change demand that come from our environment, our fund, everybody...and innovation. innovation target you have.* |
| **[3.1.1]** | ***Agile to business requirements*** | • *The pension world is very much movements now, so it is very rapidly changing.* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *We cannot find the people to keep it going.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[3.1.3]** | ***Faster time-to-market product*** | • *They have too high time to market.*<br>• *We need a faster time to market, and we are not able to do that in COBOL environment."* |
| **[2.6.1]** | ***Cultural resistance in organization not to adapt new system*** | • *Yes... you train all kind of people. More than hundred people.*<br>• *Yes it is, beacause then you are decovering that not ebody is going to be able to use the new technolgy.*<br>• *[Q] So it is difficult to get people to learn people to get use to the new environment [A] Yeap... yeah it is difficult.* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *There's always people that cannot move to the new technologyies. But we dont have that problem because it takes so long that everybody in the old technolgy is really on the retirement day period.* |
| **[3.4.1]** | ***Lack of supplier / vendor / system continuity*** | • *Keep access to your resources.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *It is a supported from the top number one I think.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *They are too expensive.* |
| **[2.5.1]** | ***Difficult to extract business rules or knowledge*** | • *[Q] To understand you own system it is really difficult at the begining. [A] Yess...* |

# Participant 23

| | | |
|---|---|---|
| **[1.9.1]** | ***Old system*** | • *Software of almost 20 years, hardware of almost 20 years that is still working.*<br>• *In my opinion, legayc is old hardware, old software thta's still need support.* |
| **[1.7.1]** | ***Unsupported supplier*** | • *If system cannot longer be supported by software or hardware.*<br>• *Because machine that are not supported any longer is a great risk for business.* |
| **[1.12.1** | ***Knowledge become scarce*** | • *So that is very dificult task, to maintain the knowledge about the software..*<br>• *We have couple of programmers around 45 - 50 and 1 even older 85, 58 where able to support those PASCAL software*<br>• *[Q] Is that well documented? The old guy with PACAL... [A] Not everything.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.6.2]** | ***Adaptability*** | • *When need to deliver support to old systems then you often confronted the fact that the parts that are broken hardware or software, can not be exchange by the same part.*<br>• *And another thing is a…it is also difficult task that the old system, the old hardware sytems need still to be supported in new software system.* |
| **[1.2.2]** | ***Stable system*** | • *Often the old technology is more stable than the new technology.* |
| **[1.9.2]** | ***Availability*** | • *That can stay over a couple of years that way, so you can say that the old technology is proven technology.* |
| **[1.6.1]** | ***Interoperability*** | • *Because old system doesnt have information exchange system.* |
| **[3.4.1]** | ***Lack of supplier / vendor*** | • *Other reason is when you have good running product which is well designed and almost at the end of its life cycle.* |
| **[2.1.1]** | ***Data migration*** | • *It is more difficult when a lot of third parties are involved. SO client has machine from us, and warehouse is from supplier A and warehose management system from supllier B, etc... and what you see is that they seldomly at the same level. So one work with data exchange based on small text files, and they are vary each other. So that's always complicated, so always...data exchagne is allways need extra costumization.* |

# Participant 24

| | | |
|---|---|---|
| **[1.14.1]** | ***Expensive in maintenance*** | • *Usually the cost of [get rid] of your legacy systme is exteremly prohibited.*<br>• *Most company claims that they spent 80% of their budget on running and 20% of innovation.* |
| **[1.11.1]** | ***Multiple systems*** | • *Something like 30 or 40 years of programming.* |
| **[1.11.4]** | ***Complicated system*** | • *[Q] A lot of business rule, a lot of business logic inside of legacy system which make it difficult or even impossible to [A] yeah..this is so..make it difficult or nearly impossible to get rid of it.* |
| **[1.13.2]** | ***Business Critical*** | • *Rules are have been key for the survival for the company. because they [Legacy systems] have been there for 30 years so they really are the foundation for the survival of the organization*<br>• *legacy system contain the key business rules for a business.* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *I think it is very difficult to get people from university and i aske them do you wnat to be a COBOL programmer?* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.4.2]** | ***General Performance is good*** | • *I think ten thousand people are doing airlines booking with this little processing power. So performance is never an issue in legacy system, at least I've never seen it."* |
| **[3.2-Special]** | ***It's business, not IT decision*** | • *I dont think it make sense from technolgy perspective becuase almost all of it isworking*<br>• *There's anoher priority for tomorrow. and you can repeat that same question for years and years and nothing will happen and that still wont break.so ...youve been saying that it will break within the next 4 months from the last 10 years. hahaha...* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the organization*** | • *It is from those people that have built their career on that system. SO they 're not in the day to day operation.* |
| **[2.1.1]** | ***Data migration*** | • *Somebody use a field that was designed to hold some information was miss-used to some other information.* |
| **[2.2.1]** | ***Lack of resource (e.g. documentation, expertise, etc)*** | • *Yes...you rally need deep expertise. because they have been so many tricks has used in the past.* |
| **[1.10.1]** | ***Risk of running legacy system*** | • *They key problem is they go out of fashion.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | • *[Q] so it means that the support from board, the support higher position in the company also determine the successful and the failure of the modernizaiton. [A] Yeah.yeah...absolutely* |
| **[2.4.1]** | ***Time constraints to finish legacy modernization*** | • *Make it as short as possible.* |
| **[2.7.1]** | ***Funding legacy modernization projects*** | • *So to replace a system would probably take at least 10 times than normal maintenace cost.*<br>• *So it is money but also being very clear that has to be happen.* |
| **[1.5.2]** | ***Functional appropriateness*** | • *So an anything that is working, it is legacy.*<br>• *So if it working and proven then you could actaully consider it legacy.* |
| **[1.9.2]** | ***Availability*** | |
| **[3.1.2]** | ***Create new business opportunity*** | • *If some of the fundamental rules in the business change. and that could either be because they are entering a new business or they merge and they access to another sytem, that could reasons* |
| **[3.1.2]** | ***Create new business opportunity*** | |

# Participant 25

| | | |
|---|---|---|
| **[1.8.1]** | ***Not flexible*** | • *So then they need to rebuild the whole system to get more flexibility.*<br>• *We want more flexible and be able to change faster, be more adaptable.*<br>• *But it did its job, but less flexibility.*<br>• *We had a lot of systems before, and they were built in CICS, COBOL, DB2 and were not flexible. So they needed to be modernized to get more exibility.* |
| **[1.5.2]** | ***Functional appropriateness*** | • *But it did its job, but less flexibility.*<br>• *But normally they work.*<br>• *When you do high volume, mainframe is very good in it.* |
| **[1.5.1]** | ***Functional completeness*** | • *There was no portal for citizens. So we need a lot of functionality and we build the new system and throw away the old system.* |
| **[1.9.1]** | ***Old system*** | • *For the system we use, they are old and old technology.* |
| **[1.6.2]** | ***Adaptability*** | • *We want more flexible and be able to change faster, be more adaptable* |
| **[3.1.2]** | ***Create new business opportunity*** | • *Higher demand also from the public to the system that [XXX] has. The public want immediately response.* |
| **[1.4.1]** | ***Time behavior*** | • *Higher demand also from the public to the system that [XXX] has. The public want immediately response.*<br>• *It works but it is not quite responsive system.*<br>• *But if we want I/O intensive system, then we use mainframe since mainframe is really good for that kind of task. I/O in mainframe is really…really fast.* |
| **[1.11.5]** | ***Poor quality of design and ecosystem*** | • *It is step-by-step process, all written in COBOL and they are connected to each other.* |
| **[1.13.2]** | ***Business Critical*** | • *We want to change but it is hard since it is the core of our...all the people in the Netherlands are in it. We cannot afford if database corrupt or to make any changes that can affect the continuity.* |
| **[1.7.1]** | ***Unsupported supplier*** | • *Before they were taken over, we were really worried about continuity. If the company go out of business, we really have a problem* |
| **[1.10.1]** | ***Risk of running legacy system*** | • *If we arrive at the point that we cannot guarantee the continuity, then we really need to change that system.* |
| **[3.1.1]** | ***Agile to business requirements*** | • *We are government organization and there is also politic involve that in circumstance that we really need to do. And if we do not, then our higher chief, the minister has a problem.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *We do know that we spend a lot of effort and money on maintenance.* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *I think it is going to be a problem for few futures, but i don't know what the plans are. I think age is a problem in [XXX].* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[3.1.2]** | ***Create new business opportunity*** | • *Also creating business opportunity.* |
| **[3.1.3]** | ***Faster time-to-market product*** | • *It has to do with how fast we can deliver...time to market. If you want to shorten time to market then modern system should be easier and possible to realize it.* |

# Participant 26

| | | |
|---|---|---|
| **[1.9.1]** | ***Old system*** | • *It has some age. In [XXX] we have system over 30 years now and still using until now.* |
| **[1.8.1]** | ***Not flexible*** | • *However, they are inflexible.* |
| **[1.12.1]** | ***Knowledge become scarce*** | • *We have no documentation, or it is not updated.*<br>• *When they are not there we cannot help new employee to work with it.*<br>• *You have to have people who understand the new technology.* |
| **[3.4.2]** | ***Lack of documentation / experts*** | |
| **[1.1.1]** | ***Modifiability*** | • *So we are very afraid to make major changes to it.* |
| **[1.5.2]** | ***Functional appropriateness*** | • *Technology still working.* |
| **[1.3.2]** | ***Learnability*** | • *Not easy used by the employees.* |
| **[1.6.2]** | ***Adaptability*** | • *It's a big problem, because a lot of software were created for the platform XP, does not work for platform window 7.* |
| **[1.7.1]** | ***Unsupported supplier*** | • *If we continue then we don't get the security update.* |
| **[3.3.1]** | ***Prone to failure*** | • *However, once when are we modernizing it we are using different languages now which are more mainstream.* |
| **[1.14.1]** | ***Expensive in maintenance*** | • *[Q] Do you think that the maintenance of legacy system could be cheaper if you move legacy system to the new environment? [A] Yes. Now it is a lot higher. We have a budget every year for IT, more than 85% for just maintenance and 15% to do something new.* |
| **[1.4.2]** | ***General Performance is good*** | • *Yes... performance still OK. There's not a real problem.* |
| **[3.1.1]** | ***Agile to business requirements*** | • *But it is not just migrating the system, but it also major change in legislation.*<br>• *Yes...Law. So it is influence also by regulation* |
| **[2.6.2]** | ***Resistance from the current users or maintainers in the*** | • *[Q] So even people from old system not cooperate? [A] Yeah...* |

| | | |
|---|---|---|
| | ***organization*** | |
| **[2.2.1]** | ***Lack of resource (e.g. documentation, expertise, etc)*** | - *We hire people from [XXX], consultant, architect. It's too big for our regular IT staff to do it, so we're hiring a lot of experts from companies like [XXX], [XXX] to help design new system and help develop the new system, test it.* |
| **[2.6.1]** | ***Cultural resistance in organization not to adapt new system*** | - *Culture yes...people are used to do thing in a way, when you are changing that, well that's you have to think about it, how can we change the behavior about people.* |
| **[2.6.3]** | ***Communicate the reasons or consequences of modernization*** | - *Yes...when they support it, they have to give time and money to go to education.* |
| **[2.5.2]** | ***Difficult to effectively prioritize the functionality for modernization*** | - *We are very optimistic. That's one of the lesson learnt to take your time.* |

## Index of Codes to Participants

**[1.1.1] Modifiability**
P1, P2, P6, P7 and P8, P9, P10, P11, P12, P13, P16, P17, P19, P20 and P21, P22, P26, P4 and P5.

**[1.2.1] Robust**
P18, P20 and P21, P1, P6, P12, P15.

**[1.2.2] Stable system**
P1, P2, P7 and P8, P16, P17, P19, P14, P23, P4 and P5.

**[1.3.1] User interface aesthetics**
P6, P18.

**[1.3.2] Learnability**
P1, P2, P6, P7 and P8, P19, P20 and P21, P18, P26, P4 and P5.

**[1.4.1] Time behavior**
P2, P11, P19, P25.

**[1.4.2] General Performance is good**
P1, P2, P7 and P8, P19, P22, P14, P24, P26.

**[1.5.1] Functional completeness**
P1, P2, P6, P7 and P8, P9, P11, P17, P25, P12, P15.

**[1.5.2] Functional appropriateness**
P1, P2, P19, P18, P25, P15, P24, P26.

**[1.6.1] Interoperability**
P1, P9, P11, P17, P19, P20 and P21, P14, P23, P18, P15, P10, P16, P4 and P5.

**[1.6.2] Adaptability**
P1, P2, P7 and P8, P9, P16, P19, P23, P18, P25, P26.

**[1.7.1] Unsupported supplier**
P2, P7 and P8, P12, P17, P19, P20 and P21, P14, P23, P18, P25, P11, P26.

**[1.8.1] Not flexible**
P1, P2, P9, P12, P13, P16, P22, P14, P25, P15, P6, P26.

**[1.9.1] Old system**
P1, P2, P7 and P8, P9, P11, P13, P17, P14, P23, P18, P25, P6, P26, P4 and P5.

**[1.9.2] Availability**
P6, P11, P1, P17, P14, P23, P24, P4 and P5.

**[1.10.1] Risk of running legacy system**
P1, P12, P16, P20 and P21, P14, P25, P13.

**[1.11.1] Large multiple systems**
P2, P10, P16, P20 and P21, P6, P1, P7 and P8, P9, P17, P24.

**[1.11.2] Database issue**
P10, P12, P17, P6.

**[1.11.3] Back Office**
P2, P7 and P8.

**[1.11.4] Complicated systems**
P7 and P8, P9, P12, P20 and P21, P18, P15, P24.

**[1.11.5] Poor design and ecosystem**
P1, P2, P7 and P8, P9, P10, P18, P15, P6, P11, P19, P14, P25.

**[1.12.1] Knowledge become scarce**
P1, P2, P7 and P8, P9, P10, P11, P12, P13, P16, P19, P20 and P21, P22, P14, P23, P18, P25, P15, P24, P6, P26, P4 and P5.

**[1.13.1] Does not fit with future strategy**
P1, P10, P11, P19, P20 and P21, P22, P14, P9.

**[1.13.2] Business Critical**
P1, P9, P10, P11, P12, P14, P25, P24, P6, P2, P7 and P8, P11, P20 and P21.

**[1.14.1] Expensive in maintenance**
P1, P2, P7 and P8, P9, P10, P11, P12, P13, P16, P17, P22, P14, P18, P25, P24, P6, P4 and P5, P26.

**[3.1.1] Become flexible to support changing business requirements**
P1, P2, P9, P12, P13, P16, P22, P14, P25, P15, P6, P26.

**[3.1.2] Create new business opportunity**
P2, P10, P16, P18, P25, P13, P17, P24.

**[3.1.3] Faster time-to-market product**
P2, P7 and P8, P16, P22, P14, P25.

**[3.2.1] Reduce cost of maintenance and operation**
P1, P2, P7 and P8, P9, P10, P11, P12, P13, P16, P17, P22, P14, P18, P25, P24, P6, P4 and P5, P26, P15.

**[3.3.1] Prone to failure**
P12, P16, P20 and P21, P14, P25, P7 and P8, P1, P13, P15, P26.

**[3.4.1] Lack of supplier / vendor**
P1, P7 and P8, P9, P16, P17, P19, P22, P23, P18, P4 and P5.

**[3.4.2] Lack of documentation / experts**
P1, P2, P7 and P8, P9, P10, P11, P12, P13, P16, P19, P20 and P21, P22, P14, P23, P18, P25, P15, P24, P6, P26, P4 and P5.

**[2.1.1] Data migration**
P2, P10, P11, P12, P16, P17, P19, P14, P23, P15, P24, P6.

**[2.2.1] Lack of resources**
P2, P9, P11, P13, P15, P24, P6, P26, P16, P17.

**[2.3.1] Poor system architecture or infrastructure**
P1, P2, P7 and P8, P9, P10, P18, P15, P6, P11, P19, P14, P25.

**[2.3.2] Difficult to Test**
P12, P13, P15, P20 and P21.

**[2.4.1] Time constraints to finish legacy modernization**
P1, P2, P7 and P8, P9, P12, P13, P16, P17, P20 and P21, P15, P24, P6.

**[2.5.1] Difficult to extract business rules or knowledge**
P1, P12, P19, P20 and P21, P22, P18, P14.

**[2.5.2] Difficult to effectively prioritize the functionality for modernization**
P2, P7 and P8, P9, P11, P12, P13, P17, P20 and P21, P22, P14, P26, P10.

**[2.6.1] Cultural resistance in organization not to adapt new system**
P11, P16, P22, P26, P12, P16, P17, P19, P14, P6.

**[2.6.2] Resistance from the current users or maintainer in the organization**
P1, P9, P17, P22, P15, P24, P6, P26, P4 and P5, P7 and P8, P12, P13, P19, P18.

**[2.6.3] Communicate the reasons or consequences of modernization**
P1, P2, P11, P12, P17, P20 and P21, P18, P6, P9, P10, P13, P22, P14, P15, P24, P26, P16.

**[2.7.1] Funding legacy modernization projects**
P1, P2, P7 and P8, P9, P12, P13, P16, P22, P18, P15, P24, P6.

**[2.7.2] Predicting Return on Investment (ROI) of modernization**
P9, P20 and P21, P10, P11, P17, P14.

**Appendix D**

In this section we present the online survey that is publicly available at esurv.org[4].

## 1. Personal Information

All of your responses will be treated as confidential. The results of the survey will be used for research purposes.

While participating in this survey, please always relate the answers to your experience with legacy systems and/or legacy modernization projects you have taken part in.
Please add your personal information.

* Country of work

* Experience (No. of years in IT)

Do you have experiences with systems that are considered legacy within the organization?

○ Yes ○ No

If Yes, please indicate your experience with legacy systems in years.

Please specify the domain of your organization.

□ Academic Institution □ Telecommunication Institution □ Software Security

□ Consulting Company □ Software Development Company □ Service Provider

□ Health-Care Institution □ IT Research Institution □ Financial Institution

□ Government organization

Other :

What is your role in the organization?

□ Researcher

□ Developer

□ System Analyst

□ Business Analyst

□ IT manager

□ Chief Information Officer

Other (specify)

[4] http://esurv.org/online-survey.php?surveyID=OCMKGL_857c0d3b

## 2. Characteristics of legacy systems

**Reliable system** : degree to which a system performs specified functions under specified conditions for a specified period of time; includes *robustness*, *stable*, *high availabilit*y sub-characteristics [**ISO/IEC 9126**].
**High performance** : the capability of a system to process a given amount of tasks in a determined time interval; includes *response time* sub-characteristic.
**Proven technology** : Systems that have been demonstrated or verified without doubt to comply business requirements for long period of time.
**Business critical :** systems whose disruption or malfunctioning will cause a failure in business operations. The business critical systems contain vital features/functionalities and their failure can result on serious financial & legal problems, damages and other penalties.

What do you think are defining characteristics of legacy systems?

□ Reliable system

□ High performance

□ Proven technology

□ Business critical

Others :

Please rank on a scale of 1-5 (1-being least critical; 5-being most critical) the following problem(s) around legacy systems that you have experienced.

| | **1** | **2** | **3** | **4** | **5** |
|---|---|---|---|---|---|
| **Lack of documentation** | ○ | ○ | ○ | ○ | ○ |
| **Lack of experienced manpower** | ○ | ○ | ○ | ○ | ○ |
| **Limited suppliers/vendors to support and maintain** | ○ | ○ | ○ | ○ | ○ |
| **Unable to adequately support, maintain, or enhance inhouse** | ○ | ○ | ○ | ○ | ○ |
| **Incompatible with current and/or future technological environments** | ○ | ○ | ○ | ○ | ○ |
| **Too rigid to comply with new business requirements** | ○ | ○ | ○ | ○ | ○ |
| **Monolithic architecture** | ○ | ○ | ○ | ○ | ○ |
| **High risk of failure** | ○ | ○ | ○ | ○ | ○ |
| **Poor user interface** | ○ | ○ | ○ | ○ | ○ |
| **Too costly to maintain** | ○ | ○ | ○ | ○ | ○ |

Reset

Do you consider the programming language as one of the factors to decide if a system is legacy?

○ Yes ○ No

Can you give some examples of programming languages that you consider a legacy or are being used as a legacy in your organization?

## 3. Legacy Modernization Projects

**Definition:**
Legacy Modernization is a process of migrating / evolving a software system to a new target system.

Have you ever been involved in legacy modernization?

○ Yes ○ No

Please rank which of the following driver(s) are the motivating factors to modernize legacy systems.

| | **Very weak** | **Weak** | **Strong** | **Very Strong** |
|---|---|---|---|---|
| **Become flexible to support changing business requirements** | ○ | ○ | ○ | ○ |
| **Create new business opportunities via mergers & acquisitions** | ○ | ○ | ○ | ○ |
| **Reduce the cost of maintenance & operations** | ○ | ○ | ○ | ○ |
| **Limited suppliers/vendors to support legacy systems** | ○ | ○ | ○ | ○ |
| **Lack of experts/documentation around legacy systems** | ○ | ○ | ○ | ○ |
| **Prone to failures** | ○ | ○ | ○ | ○ |
| **Faster time-to-market of product** | ○ | ○ | ○ | ○ |

Reset

Others :

How challenging are the following obstacles in a legacy system migration projects?

| | **Not Challenging** | **Less Challenging** | **Quite Challenging** | **Challenging** | **Very Challenging** | **I don't know** |
|---|---|---|---|---|---|---|
| **Data Migration** | ○ | ○ | ○ | ○ | ○ | ○ |
| **Lack of resources (e.g. documentation, experts)** | ○ | ○ | ○ | ○ | ○ | ○ |
| **Poor system architecture or infrastructure (e.g. monolith, hardcoded, spaghetti architecture)** | ○ | ○ | ○ | ○ | ○ | ○ |
| **Difficult to extract business rules/knowledge** | ○ | ○ | ○ | ○ | ○ | ○ |
| **Difficult to test** | ○ | ○ | ○ | ○ | ○ | ○ |
| **Resistance from the current users/maintainers in the** | ○ | ○ | ○ | ○ | ○ | ○ |

| | | | | | | |
|---|---|---|---|---|---|---|
| **organization** | | | | | | |
| **Cultural resistance in organization not to adapt new system** | ○ | ○ | ○ | ○ | ○ | ○ |
| **Difficult to effectively prioritize the functionality for modernization** | ○ | ○ | ○ | ○ | ○ | ○ |
| **Difficult to communicate the reasons/consequences of modernization** | ○ | ○ | ○ | ○ | ○ | ○ |
| **Funding legacy modernization projects** | ○ | ○ | ○ | ○ | ○ | ○ |
| **Time constraints to finish legacy modernization** | ○ | ○ | ○ | ○ | ○ | ○ |
| **Predicting Return of Investment (ROI) of modernization** | ○ | ○ | ○ | ○ | ○ | ○ |

Reset

Other :

Please indicate which of these factors are responsible for postponing legacy modernization. (Please check all that apply)

- □ Huge investment
- □ Less changes in core business process
- □ Difficult to find preferable / customize software
- □ No profit made during modernization project
- □ New system might behave differently
- □ Loss of clients
- □ Resistance from the current users/maintainers in the organization
- □ Downtime of the system during modernization
- □ Do not fix a system, until it is broken

Others :

Do you use any academic resources while performing legacy modernization projects? (for example: research articles, tools, etc)

○ Yes

○ No

## 4. Legacy Modernization Project

What resources do you use from academia?

□ Research articles

□ Research software tools

□ Research collaboration with academics

□ Internship students

Other (specify)

## 5. Legacy Modernization Project

If your answer to Question 13 (Do you use any academic resources while performing legacy modernization projects?) is No, please specify some reasons.

How can academics play a larger role in industrial legacy modernization projects?

Please provide your email below if you would like to get the result of the survey.

(We will not use your email for advertisement neither will distribute it to third parties.)

This research is a part of the ServiciFi research project , funded by NWO/JACQUARD research program and is carried out by B.V. Batlajery, drs. R. Khadka, and Dr. S. Jansen on behalf of Utrecht University.

Thank you very much for your participation in this survey. If you have any questions or concerns please do not hesitate to email us at legacysystemstudy@gmail.com.

This research is being carried out by B. V. Batlajery; drs. R. Khadka; Dr. S. Jansen on behalf of Utrecht University.

## Appendix E

We presented the overall result of the survey from the esurv tool[5]. Note that these results represent the findings of all respondents (i.e., this result also includes responses that were discarded). Hence, the results presented by esurv differ from the results of Section 7.

[5] www.esurv.org

# Results for: Revisiting legacy systems and legacy modernization from industrial perspective

## Page: 1/7

## Personal Information

All of your responses will be treated as confidential. The results of the survey will be used for research purposes.

While participating in this survey, please always relate the answers to your experience with legacy systems and/or legacy modernization projects you have taken part in.

1) Please add your personal information.

| Country of work | Experience (No. of years in IT) |
|---|---|
| Netherlands | 5 |
| Portugal | 25 |
| United States | 40 |
| usa | 30 |
| India | 13 |
| Canada | 32 |
| Costa Rica | 20 |
| United States | 35 |
| Netherlands | 23 |
| france | 25 |
| usa | 30 |
| netherlands | 27 |
| Holland | 25 |
| Netherlands | 29 |
| Netherlands | 19 |

2) Do you have experiences with systems that are considered legacy within the organization?

| | Response (%) | Responses |
|---|---|---|
| Yes | 89.90 | 187 |
| No | 10.10 | 21 |
| | Answered Question | 209 |
| | Skipped Question | 2 |

| If Yes, please indicate your experience with legacy systems in years. |
|---|
| 10 |
| 30 |
| 15 |
| 1 |
| 20 |
| 25 |
| 10 |
| 24 |
| 15 |
| 4 |

3) Please specify the domain of your organization.

| | Response (%) | Responses |
|---|---|---|
| Academic Institution | 9.35 | 26 |

|  |  |  |  |
|---|---|---|---|
| Telecommunication Institution |  | 3.60 | 10 |
| Software Security |  | 2.88 | 8 |
| Consulting Company |  | 21.94 | 61 |
| Software Development Company |  | 28.06 | 78 |
| Service Provider |  | 11.87 | 33 |
| Health-Care Institution |  | 1.08 | 3 |
| IT Research Institution |  | 4.32 | 12 |
| Financial Institution |  | 10.07 | 28 |
| Government organization |  | 6.83 | 19 |
|  |  | Answered Question | 208 |
|  |  | Skipped Question | 3 |

| Other : |
|---|
| Utility (Energy distribution) |
| Retail |
| Utilities |
| I checked multiple options because I worked for many different companies during my 20+ years. The bulk of my experience with legacy systems was in the telecom and financial sectors. |
| Digital Media |
| Manufacturing |
| electronic equipment manufacturer |
| Semiconductor manufacturer |
| Railroad company |
| Rail Infrastructure Manager |

4) What is your role in the organization?

|  |  | Response (%) | Responses |
|---|---|---|---|
| Researcher |  | 18.43 | 40 |
| Developer |  | 26.27 | 57 |
| System Analyst |  | 17.51 | 38 |
| Business Analyst |  | 12.90 | 28 |
| IT manager |  | 16.13 | 35 |
| Chief Information Officer |  | 8.76 | 19 |
|  |  | Answered Question | 208 |
|  |  | Skipped Question | 3 |

| Other (specify) |
|---|
| Sales &.Marketing |
| Technical lead legacy data migration |
| Domain Architect |
| Partner |
| IT Architect |
| IT consultant |
| business architect |
| Project manager |
| Marketing |
| Director |

## Page: 2/7

## Characteristics of legacy systems

**Reliable system** : degree to which a system performs specified functions under specified conditions for a specified period of time; includes *robustness*, *stable*, *high availability* sub-characteristics [**ISO/IEC 9126**].
**High performance** : the capability of a system to process a given amount of tasks in a determined time interval; includes *response time* sub-characteristic.

**Proven technology** : Systems that have been demonstrated or verified without doubt to comply business requirements for long period of time.
**Business critical :** systems whose disruption or malfunctioning will cause a failure in business operations. The business critical systems contain vital features/functionalities and their failure can result on serious financial & legal problems, damages and other penalties.

5) What do you think are defining characteristics of legacy systems?

| | Response (%) | Responses |
|---|---|---|
| Reliable system | 25.60 | 107 |
| High performance | 11.72 | 49 |
| Proven technology | 26.32 | 110 |
| Business critical | 36.36 | 152 |
| | Answered Question | 189 |
| | Skipped Question | 22 |

| Others : |
|---|
| New systems and technologies are invariably superior to legacy |
| No longer under active development except for minor feature upgrades and bug fixes |
| Not flexible (hard and expensive to change) |
| Mature applications Strong integration with business processes |
| poorly documented, limited skilled staff available |
| People don't want to change them; if there's a bug, fix it without changing the code (or change it as little as possible). |
| Frequently underinvested in; based on vendor hardware or software that is out of support; hard to find trained resources |
| Mostly a perfect fit for the organization. |
| End of life hardware, hard to replace, hard to find spare parts |
| Resistance to evolution and maintenance |

6) Please rank on a scale of 1-5 (1-being least critical; 5-being most critical) the following problem(s) around legacy systems that you have experienced.

| | 1 | 2 | 3 | 4 | 5 | Responses | Total |
|---|---|---|---|---|---|---|---|
| Lack of documentation | **9.29%** | **11.48%** | **24.04%** | **26.23%** | **28.96%** | **183** | **10%** |
| Lack of experienced manpower | **2.72%** | **16.85%** | **28.26%** | **32.07%** | **20.11%** | **184** | **10%** |
| Limited suppliers/vendors to support and maintain | **10.56%** | **17.78%** | **26.67%** | **27.78%** | **17.22%** | **180** | **10%** |
| Unable to adequately support, maintain, or enhance inhouse | **7.82%** | **20.11%** | **27.93%** | **26.26%** | **17.88%** | **179** | **10%** |
| Incompatible with current and/or future technological environments | **10.99%** | **12.64%** | **25.82%** | **27.47%** | **23.08%** | **182** | **10%** |
| Too rigid to comply with new business requirements | **10.50%** | **14.92%** | **20.44%** | **32.04%** | **22.10%** | **181** | **10%** |
| Monolithic architecture | **11.05%** | **22.10%** | **24.31%** | **28.18%** | **14.36%** | **181** | **10%** |
| High risk of failure | **30.56%** | **32.22%** | **17.22%** | **12.78%** | **7.22%** | **180** | **10%** |
| Poor user interface | **11.36%** | **28.41%** | **27.27%** | **23.30%** | **9.66%** | **176** | **10%** |
| Too costly to maintain | **9.34%** | **13.19%** | **26.37%** | **25.82%** | **25.27%** | **182** | **10%** |

7) Do you consider the programming language as one of the factors to decide if a system is legacy?

| | Response (%) | Responses |
|---|---|---|
| Yes | 54.05 | 100 |
| No | 45.95 | 85 |
| | Answered Question | 185 |
| | Skipped Question | 26 |

8) Can you give some examples of programming languages that you consider a legacy or are being used as a legacy in your organization?

| Text Answers (15) |
|---|
| CoBOL, FORTRAN, Assembler, Pascal, Powerbuilder, Visual Basic 6.0 |
| Assembler, PL/1, COBOL, AS/400 |
| Java |
| Visual Basic 6, Informix 4GL, Unisys EAE/LINC |
| Visual Basic (not VB.NET), C, COBOL, BASIC, PowerBuilder, Delphi (Pascal), Informix |
| Cobol, Progress |
| cobol |
| Pascal. Or source code under Open VMS, Decnet protocol, DMQ |
| Cobol, C++, PL1, LINC, Java(Old versions), VB, CoolGen, |
| Pascal dedicated for the hardware it runs on |
| cobol |
| Legacy has nothing to do with programming language. If a system is not build in a structured, maintainable way this system of today will be the legacy of tomorrow. |
| Natural Adabas |
| RPG, COBOL, C ANSI, FORTRAN, ASSEMBLER |
| VB - Cobol CICS |

# Page: 3/7

## Legacy Modernization Projects

**Definition :**
Legacy Modernization is a process of migrating / evolving a software system to a new target system.

9) Have you ever been involved in legacy modernization?

| | Response (%) | Responses |
|---|---|---|
| Yes | 89.47 | 153 |
| No | 10.53 | 18 |
| | Answered Question | 171 |
| | Skipped Question | 40 |

10) Please rank which of the following driver(s) are the motivating factors to modernize legacy systems.

| | Very weak | Weak | Strong | Very Strong | Responses | Total |
|---|---|---|---|---|---|---|
| Become flexible to support changing business requirements | **4.82%** | **11.45%** | **39.16%** | **44.58%** | **166** | **14%** |
| Create new business opportunities via mergers & acquisitions | **16.56%** | **38.65%** | **28.83%** | **15.95%** | **163** | **14%** |
| Reduce the cost of maintenance & operations | **5.39%** | **16.17%** | **41.92%** | **36.53%** | **167** | **14%** |
| Limited suppliers/vendors to support legacy systems | **7.32%** | **32.93%** | **44.51%** | **15.24%** | **164** | **14%** |
| Lack of experts/documentation around legacy systems | **6.63%** | **25.90%** | **48.80%** | **18.67%** | **166** | **14%** |
| Prone to failures | **32.34%** | **42.51%** | **18.56%** | **6.59%** | **167** | **14%** |
| Faster time-to-market of product | **9.52%** | **20.83%** | **39.29%** | **30.36%** | **168** | **14%** |

| Others : |
|---|
| Enhanced capabilities (which is not the same as "become flexible") |
| For the most part, it is the PERCEPTION that Windows systems are cheaper that drives most legacy migrations. In actuality, they are NOT cheaper to support in the long run. |
| - failing to adequately integrate with the ever changing systems/software landscap - failing to comply with new crosscutting standards such as security mechanisms within a SOA (very strong) |
| Use of modern development tooling (IDE's) Deployment modernization (DevOps) Increase scalability of processes (parallellisation) Agile development/deployment |
| fix bugs. Just because a product is legacy does not mean it's bug-free! |
| Availability of hardware components (e.g. end of life, dead-end technology) |
| Hardware end of life, spare parts hard to find |
| Security vulnerabilities due to out of support platforms; lack of regulatory compliance due to out of support platforms (ie Visual Basic, Windows XP |

| |
|---|
| after April 8, 2014); requirements for hardware or operating systems that are no longer available (ex: DOS). |
| An important one is compliance. If the underlying technology is not supported by the original vendor, the main driver becomes compliance. |
| Politics |

11) How challenging are the following obstacles in a legacy system migration projects?

| | Not Challenging | Less Challenging | Quite Challenging | Challenging | Very Challenging | I don't know | Responses | Total |
|---|---|---|---|---|---|---|---|---|
| Data Migration | **7.23%** | **19.88%** | **22.89%** | **25.90%** | **23.49%** | **0.60%** | **166** | **8%** |
| Lack of resources (e.g. documentation, experts) | **1.23%** | **19.63%** | **17.18%** | **30.67%** | **30.06%** | **1.23%** | **163** | **8%** |
| Poor system architecture or infrastructure (e.g. monolith, hardcoded, spaghetti architecture) | **4.85%** | **15.15%** | **20.61%** | **27.88%** | **30.91%** | **0.61%** | **165** | **8%** |
| Difficult to extract business rules/knowledge | **5.39%** | **13.77%** | **17.37%** | **31.74%** | **30.54%** | **1.20%** | **167** | **8%** |
| Difficult to test | **6.59%** | **22.16%** | **27.54%** | **25.15%** | **17.96%** | **0.60%** | **167** | **8%** |
| Resistance from the current users/maintainers in the organization | **4.76%** | **27.38%** | **22.02%** | **19.64%** | **26.19%** | 0% | **168** | **8%** |
| Cultural resistance in organization not to adapt new system | **10.71%** | **18.45%** | **26.19%** | **22.62%** | **20.83%** | **1.19%** | **168** | **8%** |
| Difficult to effectively prioritize the functionality for modernization | **3.59%** | **20.36%** | **30.54%** | **25.75%** | **14.37%** | **5.39%** | **167** | **8%** |
| Difficult to communicate the reasons/consequences of modernization | **10.24%** | **27.11%** | **22.89%** | **23.49%** | **11.45%** | **4.82%** | **166** | **8%** |
| Funding legacy modernization projects | **2.98%** | **14.88%** | **22.02%** | **28.57%** | **27.38%** | **4.17%** | **168** | **8%** |
| Time constraints to finish legacy modernization | **2.99%** | **13.17%** | **17.96%** | **34.13%** | **30.54%** | **1.20%** | **167** | **8%** |
| Predicting Return of Investment (ROI) of modernization | **4.76%** | **10.12%** | **19.64%** | **27.98%** | **31.55%** | **5.95%** | **168** | **8%** |

| **Other :** |
|---|
| Retraining existing IT-personell |
| Suboptimal code quality due to offshore code migrations (e.g. copy&paste vs. reuse of proprietary frameworks) |
| Selecting the correct migration approach according to the needs of the stakeholders (it involves deciding whether the system is suitable for modernization or replacement): Challenging |
| Multiple wrap-ups are built up over time and it's hard to understand what would happen or what other systems would be affected (andso what stakeholders), if modernization projects is implemneted. |
| From the legacy migration projects I've been involved in, the ROI is negative. Yes, negative. |
| jsut a motivation: data incompatibility is the primary problem. Software functions (methods, api interface definitions) are tightly coupled with the datamodels. As time progresses the legacy system will be expanded upon without much consideration for the integrity of the primary datamodels. ("things have allways been done so"). As such the functions will recieve additional, oftentimes anomylous, side-effected behavior. |
| A lot depends on the approach for your modernization. The approach I have used mitigates a lot of the challenges/risks that are usually part of the choices made... |
| I have trouble with your column "quite challenging". To me, "quite challenging" would be more challenging than "challenging", yet this column is located in a less-challenging area. Perhaps you meant to say "somewhat challenging"? |
| Handle expectations of end-users. The new system must at least contain all the excisting functionality. |
| It is important to evaluate the impact of deploying the modernized system both from a technical and user perspective. |

12) Please indicate which of these factors are responsible for postponing legacy modernization. (Please check all that apply)

| | **Response (%)** | **Responses** |
|---|---|---|
| Huge investment | 24.73 | 138 |
| Less changes in core business process | 8.06 | 45 |
| Difficult to find preferable / customize software | 6.09 | 34 |
| No profit made during modernization project | 10.22 | 57 |
| New system might behave differently | 11.65 | 65 |

| | | |
|---|---|---|
| Loss of clients | 2.87 | 16 |
| Resistance from the current users/maintainers in the organization | 15.05 | 84 |
| Downtime of the system during modernization | 5.91 | 33 |
| Do not fix a system, until it is broken | 15.41 | 86 |
| | Answered Question | 169 |
| | Skipped Question | 42 |

| Others : |
|---|
| lack of resources (experts) - they should cover both legacy and modernized systems |
| Not easy to convince organisation that the costs are earned back on the longer term view by less testing, less field reports, less bugs, faster extension of features. Going for short term wins |
| Clients should never get impacted |
| Up to now the most new systems offer fancy devices and features but do not provide proven stability, security, and reliability. |
| Not clear cost-benefit ratio. |
| Time it takes to make the transition |
| Main reason is a combination of budget and unforseen bahavior COMBINED. As the system is expanded upon all manners of uncontrollable behavior is added into the system, also there will be many non-standard organisation-specific functionality hacks within the system. There is just no way that some off-the-shelf system will come even remotely close to the datastructures and software functions of the original. Refactoring efforts are underestimated with a false sense of "we know what we need". These projects are VERY prone to fail as they will not be setup and run with the required vigor and vision. Refactoring projects require a strong controlled vision/architecture without the interference of the business. RFCs can be added later, but not during the *technical* decomposition and reconstruction of the technical legacy application. |
| Again, the approach is very important. With the approach I have used, a lot of these items present no issues for the migration. |
| The initial system took a long time and cost lots of money due to poor management of the project and lack of developer skills. Ever since then there has been a fear of undertaking another rewrite of the application. Ironically newer versions (4 to date) have taken as long as a rewrite just to update to newer user interfaces/environments because of the lack of the complexity in the application. One investment in time/money to rewrite the application would have alleviated the 3 updates that took at least 2 years each to bring the application into current technology/web etc. |
| Those who have created the legacy product have moved on to new positions (or possibly even different companies). |

## Page: 4/7

*13) Do you use any academic resources while performing legacy modernization projects? (for example: research articles, tools, etc)

| | Response (%) | Responses |
|---|---|---|
| Yes | 46.82 | 81 |
| No | 53.18 | 92 |
| | Answered Question | 173 |
| | Skipped Question | 38 |

## Page: 5/7

## Legacy Modernization Project

14) What resources do you use from academia?

| | Response (%) | Responses |
|---|---|---|
| Research articles | 37.75 | 57 |
| Research software tools | 27.81 | 42 |
| Research collaboration with academics | 19.87 | 30 |
| Internship students | 14.57 | 22 |
| | Answered Question | 81 |
| | Skipped Question | 130 |

| Other (specify) |
|---|
| Anything I can find to support the argument to modernise. |
| We develop tools ourselves(we are tool vendor) and these are developed in Cooperation with University of Leipzig |
| I wrote a thesis on data migration before effectively working on a data migration platform. I've read a lot of articles on the subject, however it is no longer part of my day to day. Our department uses interns to research various topics that are of value to our modernization efforts. |
| Experiences with other organisations |

| |
|---|
| many many articles about computation and complexity. Mainly information related to functional programming (Hurray for Jeroen Fokker and Eric Meijer) |
| Gartner, etc. to justify approaches and vendors with whom to work |

## Page: 6/7

## Legacy Modernization Project

15) If your answer to Question 13 (Do you use any academic resources while performing legacy modernization projects?) is No, please specify some reasons.

| Text Answers (15) |
|---|
| Lack of adequate offering from University/Research to modernization challenges. To much focus on technological/theoretical techniques. Modernization is seen as a technical problem, not a socio-business one. |
| Our company develops Modernization technologies and we use them to help our customers to rescue their current investments. |
| We use internally developed tools; some of these originally came from academia 20 years ago. |
| No articles known. |
| don't need them |
| Don't realy know what is meant by this question. |
| I'm not sure if the tools/products we use is in its origin academic? I think some must be but I can't say which. |
| I always work with the applications which are build to replace the legacy systems. |
| Never had in my mind |
| Never thought of. |
| The modernization process, for our company M Corp, is a structured and method driven approach. Academic organizations don't add specific value to the modernization challenge other than addressing why modernization projects fail or generate less value than anticipated. Those studies/analysis are what fuels the consideration of how to modernize. |
| not available at the time ! |
| No need |
| Never thought of it... |
| First of all, how do you define "legacy modernization"? To me, that means keeping the legacy product, but making minor changes to improve it. It's not a huge project to modernize it; just fixing bugs in existing systems. |

## Page: 7/7

16) How can academics play a larger role in industrial legacy modernization projects?

| **Text Answers (15)** |
|---|
| Listen to markets, as you are doing with this survey |
| Join my fledgling Legacy Guild at www.legacyguild.org . A newly created site dedicated to the support, maintenance, currency, enhancement, preservation, and transformation of Legacy Systems, with emphasis on modernization with Service Oriented Architecture (SOA). |
| They should work more with organization like ours who perform modernization projects every day. There is not a good academy-industry relationship. |
| Academia could help by clarifying the real costs and risks of various approaches to software modernization, including: 1. Calculating the financial risks of running legacy or obsolete systems 2. Understanding the real risks and costs of different mitigation strategies, including: a) Rewrite from scratch b) Migrate to modern code base with automated tools c) Run on modern platform using virtualization or runtime libraries d) Replace with commercial off-the-shelf software (COTS) and customize e) Piecemeal component modernization and/or wrapping. |
| Make articles with direct applicable solutions, which can be used asap. |
| teach cobol |
| Educate young people in the world of the legacy systems and modernisation. |
| Smart solutions to data migration, reuse 'legacy' test base for 'as is' rebuilt of the legacy system in different environment. Conversion tooling for program languages? |
| If organisations now what the have to do. such as look what you want the system to do, and not what is the system doing. You need with legacy systems a different approch then with new applications. What are the steps to get ride of the legacy systems in steps. |
| In the first instance, communicating their experiences more easily. |
| Address details on why projects suffer in the modernization efforts |
| creating and supporting of frameworks and standards \ governance bodies for adoption ! |
| Not sure |
| a) overall thought leadership - what are viable approaches and what are not b) practice research - in depth studies of modernization efforts and the outcomes (we know that 50% of new dev projects fail; what is it for modernization) c) modernization framework (like TOGAF or CMM) that allows CIO's to convey where a company is on the "legacy" curve; d) Translate "legacy risk" into "operational risk" - case studies and databases of operational losses resulting from legacy software that can be used to justify the investment in modernization |
| Does academics hae experience in this area? How many legacy systems have they migrated or use? |