XVII International Scientific Conference on Industrial Systems (IS'17)
Novi Sad, Serbia, October 4. – 6. 2017.
University of Novi Sad, Faculty of Technical Sciences,
Department for Industrial Engineering and Management
Available online at http://www.iim.ftn.uns.ac.rs/is17

On the Problem of Software Project Management Tools Coexistence and Migration

Dajana Narandžić
(Teaching Assistant, University of Novi Sad, Faculty of Technical Sciences, Trg Dositeja Obradovića 6, 21101 Novi Sad, Serbia, narandzic@uns.ac.rs)

Vladimir Mandić
(Assistant with PhD, University of Novi Sad, Faculty of Technical Sciences, Trg Dositeja Obradovića 6, 21101 Novi Sad, Serbia, vladman@uns.ac.rs)

Marija Rakić-Skoković
(Lecturer, University of Novi Sad, Faculty of Technical Sciences, Trg Dositeja Obradovića 6, 21101 Novi Sad, Serbia, marijars@uns.ac.rs)

Sonja Ristić
(Associate Professor, University of Novi Sad, Faculty of Technical Sciences, Trg Dositeja Obradovića 6, 21101 Novi Sad, Serbia, sdristic@uns.ac.rs)

Abstract

Software development is a complex process involving many diverse activities. In order to eliminate delays, stay within the budget, and prevent project runaways software project managers must control and guide the development team throughout the project's lifecycle. They apply a number of project management tools, of both economic and managerial nature, to help manage a number of important issues in the process of software development. Different project management tools (PMTs) can be used simultaneously within the same software project that can cause the problems with the data synchronization and the data exchange between these PMTs. Besides, during the changes which occur in the organization and its environment, an organization is sometimes forced or willing to migrate to other PMT(s). New tools mostly do not support previously used data formats. That way, a vast amount of data and knowledge about the projects that were managed with the support of legacy tools is lost. In this paper an approach to solve the problems of data synchronization, exchange and loss caused by PMTs' coexistence and migration is presented. It is suggested to design a data warehouse that transforms diverse data sources into an integrated data set for permanent, long-term storage. Integrated data provide a unified, reconciled view of project management data gathered by means of different project management tools suitable to analytical decision makers.

Key words: *data formats, data synchronization, data warehouse, integrated data, migration, organization, project management tools, software development*

1. INTRODUCTION

Software development projects are getting larger and more complex and demand more sophisticated methodologies, tools and multi-functional development teams with well-defined and understood roles and responsibilities. The trend toward business globalization and the advancement of information technologies (IT) have enabled involving project team members from different geographical sites, organizations and cultures. Besides, there are several software development lifecycle models and there is no ‘one size fits all’ solution. In such a context, managers of software development projects have to eliminate delays, stay within the budget, and prevent project runaways. At the same time they have to achieve a balance between the reliability of the end software product and its compliance with functional and non-functional requirements and its ease of maintenance. For successful project management, an efficient supporting infrastructure must be implemented. Organizations purchase or develop project management tools (PMT) to support and automate their project management processes. Positive impact of PMTs' usage on project performance is reported in survey on project management practices presented in [12].

Potential benefits of an appropriate use of PMTs are as follows: increased efficiency, reduced training, improved project predictability, increased stakeholder confidence and probability of project success, and improved communication [2]. Additionally, in [2] is suggested that project managers should have adequate knowledge and experience in the use of a PMT. In addition to all the benefits that PMT brings some problems would be noted, too. Here we emphasize those that arise due to the coexistence of different PMTs within an organization or due to the migration to the other PMT(s).

Various organizations can participate in the realization of a software development project. The organizations involved in a realization of the project can use different PMTs. Problems occur with the data synchronization and the data exchange between these PMTs that are used simultaneously within the same software project.

During the changes which occur in the organization and its environment, an organization is sometimes forced or willing to migrate to other PMT(s) or to upgrade an existing PMT. New or upgraded tools very often do not support data formats that were supported by previously used tools. That way, a vast amount of data and knowledge about the projects that were managed with the support of previously used tools is lost.

As a solution in the paper it is proposed to design and implement a data warehouse (DW) that transforms diverse data sources into an integrated data set for permanent, long-term storage. In the case of the coexistence of different PMTs it would be incrementally loaded from different PMTs' data sources. In the case of PMTs upgrade/migration DW would be implemented simultaneously with the migration process and afterwards it will allow legacy data to be used in conjunction with the data incrementally loaded from new PMTs' data sources.

Apart from Introduction and Conclusion the paper has three sections. In Section 2 the background and related work are presented. The case study is described in Section 3. Proposed solution is given in Section 4.

2. BACKGROUND AND RELATED WORK

A project can be defined as a temporary endeavor undertaken to create a unique product, service or result with a defined beginning and end undertaken to meet unique goals and objectives [13]. The temporary nature of projects makes the difference between the general management of an organization and the project management. Project is unique in the way that it comprises specific set of operations designed to accomplish particular goal. A project team often includes people who do not usually work together and that are sometimes from different organizations and across multiple geographical sites and cultures. Project management is the application of knowledge, skills, tools, and techniques to project activities to meet the project requirements [10]. In the paper we focus on software development projects' management.

Most of the organizations involved in software development projects use PMTs to support and automate their project management processes. PMT comes in many degrees of sophistication and prices range. Besides the commercial PMTs there is a great number of open-source PMTs such as Redmine, Trello, Tiagaio, Restya, or Taskboard. PMTs range in their capabilities, and it is important to assess what the business needs are and how PM processes could be improved. While most quality PMTs offer similar capabilities, no two are exactly the same. That makes PMTs' selection process an essential component of organization's success. Several organizations cooperating within the same software development project may use different PMTs. Also an organization may decide to upgrade or change actual PMT. In both cases there is a problem of retaining data from different sources, their synchronization and their usage for further decision making. Project managers search for automated tools to bridge the semantics and syntax gap between data stores of source and target PMTs. There are plug-ins and migratory tools that enable export from source to target PMT, like Redmine-to-Trello plug-in and Redmine to Taiga migrator. Unfortunately, these solutions are “one-way trip” from source to target data store. In reality, PMTs' integration problems are rarely so simple that they could be solved by single one-way data transfer from source to target data store. Instead, a project of any complexity transforms data substantially and involves multiple data sources and targets. Data may exist in old and new systems simultaneously, requiring that the two systems be synchronized [14]. The solution that we propose in this paper is based on data migration, data integration and data warehouses.

The term *migration* generally refers to the movement of technology from older systems to newer, feature-rich and cost-effective systems that will result in improved interoperability, reliability and manageability [17]. Typically, migration is a very complex, long-lasting process [18]. There are two broad categories of migrations: infrastructure and application migrations. The first category refers to the migration process of more or even all layers of the computing platform while the second one applies to applications rather than infrastructures [17]. The replacement of PMT and the coexistence of different PMTs' within an organization may be classified in the second category. In this paper we assume that there are both old and new (or coexistent) PMTs and that there is no need to analyze old PMT system and to design and develop new PMT system based on that analysis. Therefore, here we do not deal with various strategies available for a full application migration effort. An overview of migration methodologies may be found in [18] and [17]. One of the common phases of different migration methodologies is Build & Implement phase. In the paper we focus on data migration step of that phase.

Data migrations generally result from the introduction of a new system. This may involve an application migration or consolidation in which one or more legacy systems are replaced or the deployment of an additional system that will coexist with the present applications [11]. Data migration is, conceptually, a simple task. Data migration is a one-time tool-supported process aiming to transfer stored data from a source/old system to a target/new one without affecting operations. Whether one is transferring that data between storage types, formats, or changing it over to a completely different system, data migration is most often undertaken as part of a larger project. Because the new system or application that is adopted in that larger project is often seen as the main investment, data migration planning is very often viewed as a process of less importance. In reality the data that organizations already possess maximizes the use of any newly acquired systems. On the other side, if data migration strategy is planned and implemented poorly than already possessed data may complicates the use of a new system. Although the data in the source system may be perfectly adequate for its current use, it may be inadequate for the objectives of the target system. Therefore, it is not the conceptual level where complexity of a data migration process lies. The complexity is a consequence of several reasons: i) source data could be stored in more than one system; ii) source data could be incomplete, inaccurate, redundant or duplicated; iii) source data usually must be populated in a different target format and data transformation is required; iv) target data model can still being defined; v) data may exist in old and new systems simultaneously, requiring that the two systems be synchronized for a long time (several months or even years). The impact of data migration on business processes is outlined in [6]. Various processes to follow during the migration of the legacy systems to contemporary applications are documented in [11]. According to [11], two principal types of data migration are: big bang migrations and trickle migrations. Big bang migrations involve system downtime while the data is extracted from the source system, processed, and loaded to the target system, followed by the switching of processing over to the target system [11]. The reality is that few organizations ever do this. Trickle migrations take an incremental approach to migrating data and involve running the old and new systems in parallel and data migration conducted in phases. The solution proposed in our paper is based on the ideas of incremental data migration.

Several data migration methodologies could be found, like those presented in [18], [11], [6], [17] and [4]. Despite some differences between them it could be summarized that data migration process can be conducted through six phases: data migration project planning, data profiling, data cleansing, data transformation, data migration and migration validation. Phases of data profiling, cleansing and transformation are conducted through multiple cycles. It is very important to notice that these phases are in the great extent similar to the phases of the process of getting data into the data warehouse and business intelligence (DW/BI) environment. The extract, transformation and load (ETL) system is positioned between the operational source systems and DW/BI presentation area and it is aimed at data getting into DW ([7], [8]). That was one of the reasons why we decide to design and implement DW in order to integrate PMTs' data and in that way to support different PMTs' coexistence within an organization and PMTs' migration/upgrade.

Data warehousing technologies have been successfully deployed in many areas: manufacturing, retail, financial services, transportation, telecommunications, utilities, and healthcare. The importance of data warehousing in the commercial segment appears to be due to a need of organizations to gather all of their information into a single place, a data warehouse, for in-depth analysis, and the desire to decouple DW from on-line transaction processing systems [19]. However, building an enterprise data warehouse is a long and complex process, requiring extensive business modeling, and may take many years to succeed. Some organizations are settling for data marts instead, which are departmental subsets focused on selected subjects [3]. DW and BI are closely related with Enterprise Resource Planning (ERP) systems. In the context of proposals presented in this paper the relationship between ERP system upgrade, replacement or integration and DW design and implementation is important.

Enterprise Resource Planning (ERP) applications are software suites that help organizations integrate their information flow and business processes [1]. Organizations regard their ERP systems as strategic assets as they contain important and mostly confidential enterprise wide data. In [14] are reported several user stories tackling the problem of ERP system replacement or integration illustrating that large projects may require multiple data movement techniques executed in multiple phases. In the case of a manufacturer that has decided to consolidate the 17 instances of ERP applications (collected through mergers, acquisitions, and the IT budgets of its subsidiaries), data migrations and application upgrades were preparatory steps for the consolidation of ERP applications and their data. Achieving application consolidation required considerable work in migrating, consolidating, upgrading, and integrating application data. For all presented user stories it is concluded that

ETL is the preferred technology for data migration. According to [5] organizations change their ERP systems in search of efficiencies to be gained from a new data structure, a new set of software tools, and the better integrated processes these systems are designed to support. The importance of ERP's upgrade is emphasized in [15] and [16] claiming that periodic ERP upgrades are necessary to incorporate new features and to stay within the vendor’s window of preferred releases. The views on the relationship between ERP upgrade process and DW/BI implementation presented in [5], [15] and [16] converge and can be summarized as follows: i) data conversion is one of the key activities involved in upgrading an ERP system; ii) ERP upgrading is an appropriate time to assess the quality of the data that is to be converted; and iii) businesses that decide to implement a data warehouse during the ERP upgrade will discover that doing so provides an opportunity to address their data quality issues. That was the final reason that leads us to apply similar approach to address problem of PMTs’ upgrade and replacement, as well as the problem of different PMTs coexistence within the same organization.

3. CASE STUDY

The problem that is being discussed here concerns an organization *X* that deals with the production and maintenance of software products.

The business of this organization takes place in the form of projects. Project management is responsible for organizing all tasks and scheduling tasks for employees. Also another task of project management is reporting on the worker's performance within the projects. Currently, organization use Redmine as PMT and is considering migration on Trello PMT. Besides, the organization *X* participates in some projects as a subcontractor. Project management for these projects is in charge of performing organization that may use PMT different from the PMT used by organization *X*. Organization *X* by itself manages the relevant project data for each subproject that it is involved in as a subcontractor. There is no automatic data exchange between organization *X* and performing organizations of projects in which organization *X* participates as a subcontractor.

Therefore, two or more organizations can work together on a project, using different tools used to manage the project data. That causes the problem of data integration across different data sources that are gathered by means of different PMTs. Another problem is that data that is important for making business decisions and reporting are also found in other sources such as Excel files or .txt files.

When making decisions about future activities of great importance are historical data. Because of the limited capacity they cannot be stored within Redmine data base and for this reason they have to be stored somewhere else.

There are numerous reports – annual, monthly, and daily. Queries aimed at their creation join several tables. The number of joined tables can be pretty large. Consequently, the response time of the system is long and does not meet the requirements and needs of the organization.

Described problem is similar to the problem of ERP upgrade and replacement and here we propose to solve it by creation of a data warehouse during the PMTs’ replacement. In that way all data from different sources will be integrated into a unique location, and all historical data will also be found here, too. Time savings and the efficiency of getting information about the current and future situation, as well as the quality of this information would reach a higher level.

4. DATA WAREHOUSE DESIGN AND IMPLEMENTATION

Data warehousing allows an organization to create a consolidated view of its enterprise data, optimized for reporting and analysis. Historical, summarized and consolidated data could be even more important than detailed, individual records [3]. According to coverage can be distinguished: Enterprise DW (EDW) covering the entire business and data mart (DM) covering only one business segment (usually only one subject) [8].

Data warehouses can be built in two ways, the first being a top-down design, which collects all data from the company at a granular level and then allocates the data to specific DM, or organized sections. The other is a bottom-up approach, where the DMs are created first and then later combined to form a complete DW [8]. In the paper we use bottom-up approach. We decide to create DW pilot project first. In that project one DM is designed and implemented. The first decision was to select business process that will be used to define a DM design. Business process to be chosen would be the one that is not very simple or too complicated, that end-users are familiar with and that is suitable for relatively rapid development of corresponding DM. In this case we select the registering process of committed tasks of project members. The creation of pilot project brings practical experience and provides additional motivation to include new users and cover new user requirements. In Figure 1, the star schema of the first-selected DM within pilot project is presented.

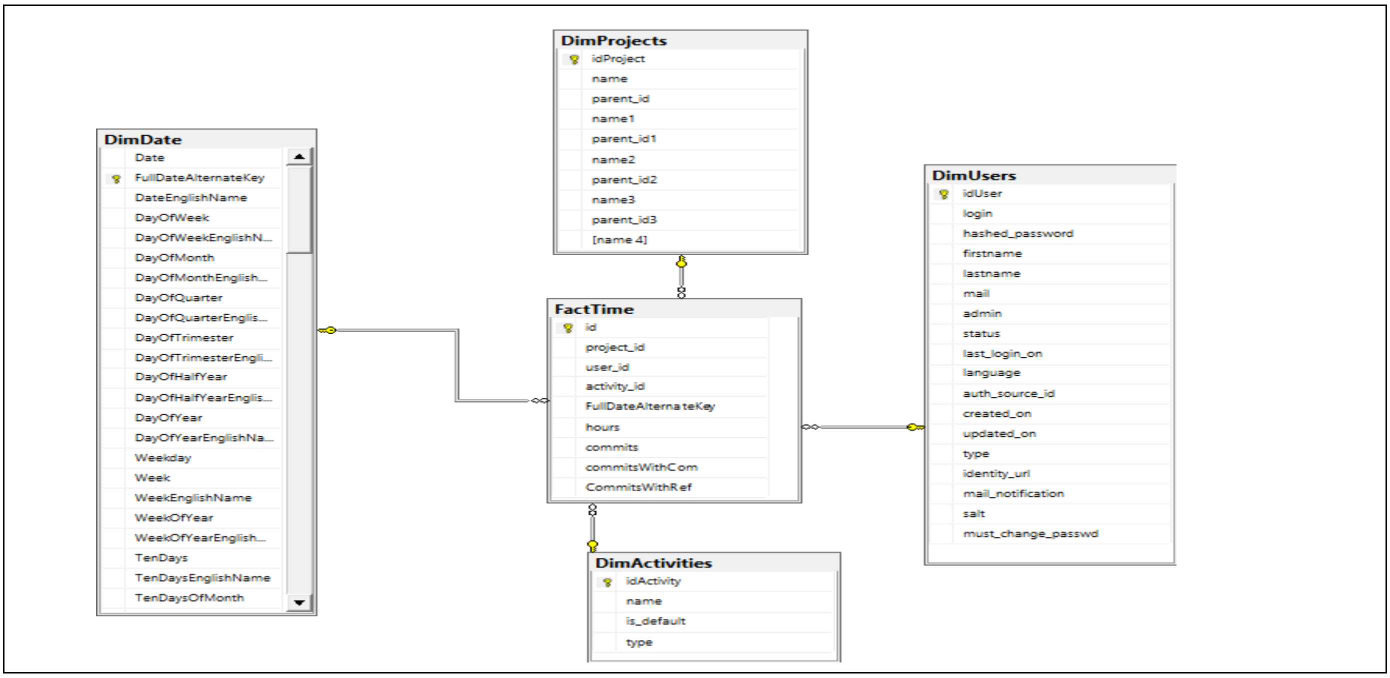

Figure 1. Data mart star schema

DM star schema contains five tables: one fact table and four dimension tables. Within the fact table, *FactTime*, there are the measures that are important for recording the effect of the team members on the projects. The most important is the number of hours spent on project activities. Other three attributes indicate whether commit happens, whether commit is commented and whether commit references particular task. Besides, fact table contains additional four attributes: *project_id*, *user_id*, *activity_id*, and *FullDateAlternateKey*. They are foreign keys that are migrated from tables which represent the dimensions. In addition to these foreign keys, a surrogate primary key is created, too. Four dimensional tables are *DimProjects*, *DimUsers*, *DimActivities and DimDate* that contain detail data about projects, users, activities and time intervals respectively. Time is of great importance for systems that support business operations for organizations. It is a key factor that defines the frames in which certain activities are performed.

After the DM is designed it would be created and ETL process would be conducted. DM star scheme is created by means of Microsoft SQL Server Management Studio (SSMS). A tool used for transformation and integration of data from an existing database into a newly created database, is Microsoft SQL Server Integration Service (SSIS). For cube creation and data analysis is used tool SQL Server Analytic Services (SSAS). Once DM is implemented several reports can be created. Reports and their visualization are created by means of SQL Server Reporting Services (SSRS). All mentioned tools are included in Microsoft SQL Server Data Tools - Business Intelligence for Visual Studio 2016.

In Figure 2 are presented two reports on spent hours per project per years, to illustrate use of implemented DM. The rows of the table are projects. Projects are structured so that one project can be a sub-project of some other project. In the report only project ids are presented. The reason for displaying only them, is to protect the confidential information of the organization *X*. Columns of these reports include years that can be further fragmented for months, weeks, days, and quarters, and in this way an accurate insight into number of hours that is dedicated to a project in particular time interval can be get. By drill option, it is possible to monitor the hours spent on projects by seeing which specific projects are within another project and to the lowest level. Report presented in the upper part of Figure 2 contains summarized data about the number of working hours on project 120. Report presented in the lower part of Figure 2 is drilled down from the upper report and contains the number of hours per each sub-project of project 120.

5. CONCLUSION

In the paper we deal with the problems of data synchronization, exchange and loss caused by PMTs' coexistence and migration. In order to find the solution the similar problems and their solutions were analyzed. The main idea arose from the experiences connected with ERP system upgrading and replacement. Based on literature review it could be concluded that ERP upgrading is an appropriate time to assess the quality of the data that is to be converted and that implementation of a data warehouse during the ERP upgrade will provide an opportunity to address a lot of data quality issues. In the way, knowledge about the source data would be reused in two ETL processes aimed at DW creation and legacy ERP data migration. Based on these ideas we propose the creation of DW during the PMTs' upgrade/replacement process. In the pilot project a data mart is designed and implemented. Examples of reports are created, too.

Created DM could be a starting point for creating a project data DW. Project management is rapidly changing due to the development of current IT. Project managers today have to use analytical techniques to monitor and control the uncertainty as well as to estimate project schedule and cost more accurately with analytics-driven prediction. Some of the applications of project analytics are: assessing feasibility of various alternatives, managing data overload, and analyzing project portfolios for project selection and prioritization. DW solution proposed in this paper could be used not only to solve the problems caused by PMTs' coexistence and migration, but also as the data source for project management analytics. In our future work we will examine application of proposed DW in that context.

		⊞ 2011	⊞ 2012	⊞ 2013	⊞
⊞	63		217.853333186358	408.30999969691	
⊞	120	94.4200001209974	653.720000348985	1424.53000074625	
⊞	121	8312.34100316465	13061.779338805	10935.6400023662	

				⊞ 2011	⊞ 2012	⊞ 2013	⊞ 2014	I
⊞			63		217.853333186358	408.30999969691	1559.13000048697	
⊟ 120	⊟ 82	⊞	29	30	93	21.5		
		⊞	142			84.2600002139807	46.2400000542402	
	⊞		120	64.4200001209974	560.720000348985	1318.77000053227	4562.64667374268	
⊞			121	8312.34100316465	13061.779338805	10935.6400023662	9400.47133047506	

Figure 2. Reports on spent hours per project and per year

6. REFERENCES

[1] Abdinnour-Helm, S., Lengnick-Hall, L.M. and Lengnick-Hall, A. S. (2003), "*Pre-implementation attitudes and organizational readiness for implementing an Enterprise Resource Planning system*", European Journal of Operational Research, Volume 146, Issue 2, pp. 258-273.

[2] Cash, C. and Fox R. (1992), "*Elements of successful project management*", Journal of Systems Management; 43(9):10-12.

[3] Chaudhuri, S. and Dayal, U. (1997), " *An overview of data warehousing and olap technology*", SIGMOD 26(1), pp. 65–74.

[4] Datalynx Pty. Ltd. (2014), " *Datalynx PCTM Methodology for Migrating Legacy Data*", White paper. available at: http://www.datalynx.com.au/Datalynx%20PCTM%20Methodology.pdf (accessed: July, 2017)

[5] Grant, T. (2015), "*The Big Secret to Painless Data Migration*, White paper. Jet Enterprise, available at: https://www.hso.com/fileadmin/user_upload/The_Big_Secret_to_Painless_Data_Migration__1_.pdf (accessed: July, 2017)

[6] IBM Global Technology Services. (2007), "*Best practices for data migration: Methodologies for planning, designing, migrating and validating data migration*", available at: http://www-935.ibm.com/services/us/gts/pdf/bestpracticesfordatamigration-wpgtw01275-usen-01-121307.pdf. (accessed: July, 2017)

[7] Inmon, W.H. (2005), *Building the Data Warehouse*, John Wiley

[8] Kimball, R. (2013), *The Data Warehouse Toolkit*, John Wiley

[9] Linthicum, D.S. (2000), *Enterprise Application Integration*, Addison-Wesley Professional

[10] Meredith, J. R. and Mantel, S. J. (2015), *Project management. A managerial approach*, Wiley Text Books (9th Edition)

[11] Oracle (2011), "*Successful Data Migration*", White paper, available at: http://www.oracle.com/technetwork/middleware/oedq/successful-data-migration-wp-1555708.pdf (accessed: July, 2017)

[12] PricewaterhouseCoopers (2007), "*Insights and Trends: Current Program and Project Management Practices* ", available at: http://www.pwcprojects.co/Documentos/BRO Boosting Business Performance DC-07-1104-A v8[1].pdf (accessed: July, 2017)

[13] *Project Management Institute A Guide to the Project Management Body of Knowledge* (PMBOK® Guide) — 2013 Fifth Edition

[14] Russom, F. (2006),. *Best Practices in Data Migration*, TDWI Monograph Series. available at: http://download.101com.com/pub/tdwi/files/tdwi_monograph_bpindatamigration_april2006.pdf (accessed: July, 2017)

[15] Syed, J. (2011), "*Implementing Oracle BI Applications during ERP Upgrade*", White paper. Emtec, Inc, available at: https://www.emtecinc.com/assets/uploads/whitepapers/Emtec_OBIA_for_ERP_wp.pdf (accessed: July, 2017)

[16] TimeXtender . (2014), "*Advantages of Implementing a Data Warehouse During an ERP Upgrade (v. 3)* ", White paper: available at:https://www.victa.nl/timextender/wp-content/uploads/timextender_whitepaper_advantages_of_implementing_a_data_warehouse_during_an_erp_upgrade_v3_hq.pdf (accessed: July, 2017)

[17] Transvive, Inc. (2011), "*Migration strategies and methodologies*", White paper. available at: https://www.google.rs/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwifvaat7_fVAhVC1xQKHUnhDp8QFggkMAA&url=https%3A%2F%2Fwww.platformmodernization.org%2Ftransvive%2FLists%2FResearchPapers%2FAttachments%2F1%2FTransvive-MainframeMigrationStrategy-WP.pdf&usg=AFQjCNGuWIStNzB2_apaxJdfNLXtas650A (accessed: July, 2017)

[18] Tripathy, P. and Naik, K. (2015), *Software Evolution and Maintenance – a Practitioner's Approach*, 1th Edition, Wiley.

[19] Widom, J. (1995), "*Research Problems in Data Warehousing*", Proc. 4th Intl. CIKM Conf.