

Microsoft Change Management Applying Comparison of Different Versions

I. Orosz, T. Orosz

Óbuda University, Székesfehérvár, Hungary

e-mail:orosz.istvan@arek.uni-obuda.hu, orosz.tamas@arek.uni-obuda.hu

Abstract: Microsoft Dynamics AX continuously extends functional and technical approaches. Former Axapta and Navision applications are being replaced with the modern and smart AX and NAV solutions, like the other Dynamics modules and components. A brand new MS Dynamics implementation follows standard methodologies of Sure Step. However, companies, where one can find former Microsoft ERP systems, like Axapta or Navision, face to several change management issues and problems. One can figure out that business requirements for Software change management depend on specific business functions, industry, technologies and so on. This paper introduces an agile way for business change management in Microsoft Dynamics environment. A comparison of different business requirements is made in different MS Dynamics business change issues.

Keywords: *Microsoft Dynamics AX, change management, Navision*

1. Introduction

Microsoft widely used in either in several core business areas, like logistics, financials and human resources or in the industry, such as automotive, pharmaceuticals, etc. Business deploying Dynamics to cover required functionalities needs continuous improvement and changes in functional and development framework of MS Dynamics System.

Dynamics AX [1-2], provides full solutions for small, middle and larger sizes of organizations. Multiple examples can be found either for replacing suddenly the entire IT System or for changing it slowly and gradually. However, we cannot either repair business processes which should be replaced sooner or later, nor replace them immediately due to sustainability reasons.

Dynamics AX, one of the main ERP software in the World was announced in 1998 as IBM Dynamics AX, it was developed by IBM and Damgaard. The developers later merged with Navision Software A/S, the combined company was annexed by Microsoft in 2002 [5]. After Microsoft handled over the product, it becomes a real enterprise resource planning system. The AX modifications is done via using MorphX, X++, AOS, AOT and other technologies. The AX 2012 version uses a new technique for custom modifications; Visual Studio 2010 has to be installed for developing clients.

The Dynamics AX software consists of four main parts [3-4]:

- Database Server, a database that stores the Dynamics AX data. This server is a Microsoft SQL server usually.
- File Server, a folder containing the Dynamics AX application files.
- Application Object Server(s) (AOS), a service that controls all aspects of Microsoft Dynamics AX's operation. There can be more AOS in case of scalability.
- Client(s), the actual user interface [13] into Microsoft Dynamics AX. Some of the business logic is programmed in clients, and some are implemented in the Application Object Server.
- The core paradigm of AX change management is innovation and sustainability, so the main focus of this paper is how these two aspects are present.
- Microsoft Dynamics AX change management

The implementation of Microsoft Dynamics AX 2012 according to Microsoft paradigm of change management is somehow different from the traditional aspects. This method describes the transition of an organization from a previous state to a planned state. This process is a managed one, which leads the whole organization along the change. In project management, this refers to a project management process, where the changes form a project and are officially introduced and approved [6]. The change management usually comes with organizational developments as well. The purpose of change management is to minimize the negative impacts on the organization and to avoid concerns. There are some different types of change management: changing the behavior of the personnel; technological changes; operational and structural changes; strategic and mission changes. The changes, which affect Dynamics AX, come from operational, structural and technological sides. The organizational change management should start with detailed description of the current situation [14], and after that focusing on the need of change and the ability of change. The objectives, content, and the process all should be part of the organizational change management plan. It uses the following techniques: performance metrics such as financial results, operational efficiency, leadership commitment, communication effectiveness. The perceived need for change to plan the appropriate strategies, in order to avoid change failures or solve troubled change projects.

Organizational change management is closer to a successful project ending, if these points are contained in the project:

1. The management of proceeds, the realization to define measurable participant aims, creating a business case for their achievement (which should be continuously updated), and monitor assumptions, risks, dependencies, costs, return on investment, disadvantages and cultural issues affecting the progress of the associated work.
2. Effective Communications that informs various stakeholders of the reasons for the change (why?), the benefits of successful implementation (what is in it for us, and you) as well as the details of the change (when? where? who is involved? how much will it cost? etc.).

3. Devise an effective education, training and/or skills upgrading scheme for the organization.
4. Counter resistance from the employees of companies and align them to overall strategic direction of the organization.
5. Provide personal coaching (if needed) to alleviate any change related fears.
6. Monitoring of the implementation and fine-tuning as required.[8]

It is also important to make a clear governance and organizational model of the company, which is suited and aligned to the expectations of the company. The consultation model has to take a closer look at the organisational culture of the company, the principles and the values along which the company is driven.

The future structure of the organization has to be designed with considering the following fields: segmentation, autonomy, integration. The flow of the changes which affects the technological side of Dynamics AX changes is shown on Fig.1.

In order to overcome the needs of BPR [10-12] which remarkably require a long process, Dynamics AX has developed an implementation methodology to speed-up the whole process by instantly implementing principals which has been underlined by Dynamics AX business process recipes.

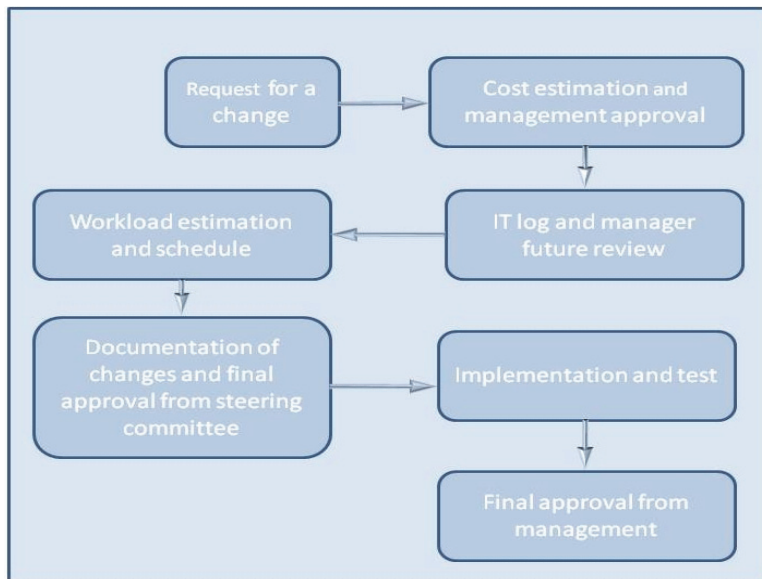


Figure 1. Flow of technological changes

Sustainability is one of the core factors of developing Dynamics AX 2012. One of the key aspects was to speed up the business and technological process of upgrading from the previous versions to AX 2012. Sure Step is the official software process for Dynamics AX. This is the tool of defining process phases, milestone roles, artifacts, cross phase processes, additional project management processes. Supports a wide range of software like Dynamics AX, NAV, GP, SL and CRM. The project types, which are

supported, are the following: full and rapid implementation, optimization and upgrade. The core steps of these implementations are: diagnostics, analysis, design, development, deployment, operation. Optimization and upgrade are different from these implementations in the way that an existing software environment has to be modified.

The process guide is an html solution combined with some ActiveX controller, which allows tailoring the process needs, as shown in Fig. 2.

Sure Step describes six main and phases and 2 additional phases for optimization and upgrade. It covers the complete lifecycle from project initiation, development, deployment, optimization and upgrade to the next version, where every phase ends at a milestone. A milestone is a sum of artifacts created or refined in the phase. If a milestone is executed with success, then the next step can be started. At the first look the phases look like a waterfall model, but naturally Sure Step isn't all about waterfall.

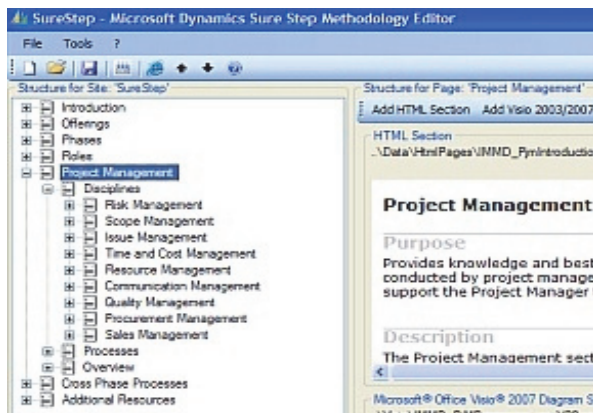


Figure 2. Microsoft Sure Step Editor

These phases are the following:

- Diagnostic: the diagnostic phase contains analysis of the customer process at a very high level. Focus of the diagnostic phase is the project initialization; setup a project plan, agree on an approach and scope definition.
- Analysis: in the analysis phase most of the business processes are identified and documented at a high level. If necessary an external specialist helps in executing this step. The goal of the analysis phase is to understand the customers business and processes performed. Modeling and documenting the customers business is important. Microsoft Provides a new tool for the modeling: Microsoft Sure Step Business Modeler
- Design: main purpose of the design phase is to find a way how the customers' processes and needs can be implemented with Dynamics. At this point there may be more than one solution for a topic. Need to identify the best strategy for the implementation phase. Although prototyping is not covered in sure step, this is the point where to build prototypes and do load testing on the prototype implementation.

- **Development:** most of the programming work is done in the development. Development covers the creation of new features and adaption of existing features as well as the data migration. All features and the data migration have to be tested. Beside feature and data migration testing it is necessary to do security testing. The phase ends when most of the specified features are developed and tested and the data migration is done.
- **Deployment:** goal of the deployment phase is to setup the operational dynamics environment at the customer. Beside installation, configuration and feature deployment, the phase focuses on testing at system level. So the specialists have to run the user acceptance tests, process tests, security tests and load tests as defined in the test plan. When the system runs and all tests pass the deployment phase is finished.
- **Operation:** this is the place for work in live environment.

Optimization and upgrade: these two often go together. The optimization steps usually lead to a new version upgrade.

2. Dynamics AX innovation and sustainability

There are some situations, where the original waterfall concept of Sure Step is not suitable. Waterfall is the classic implementation model staging from Analysis, to Design, to Deployment, and finally Operation it works for MS Dynamics Sure Step project types (Enterprise, Standard, Rapid) where those project types have different customization levels except Rapid, which has no or minimal customization, as it presents and out-of-the-box approach.

Typically the waterfall is a good method, shown at Fig. 3., when the duration of the project is shorter than the frequency of organizational change. In other words, if the organizational change is prevalent, and the project is longer than this period, then the project might fail, because the requirements might change when the project is running.

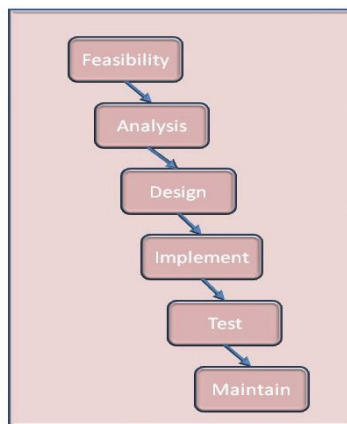


Figure 3. Waterfall method

Other issue of waterfall method is that project step has to be done in strict order, and there is no possibility to move back to the previous stage. This means some lack of innovation of the project, but with more focus on sustainability. Because all the requirements are analysed at the beginning, if any change occur in the needs, the whole process will be affected. At the end, all the development is tested and deployed at once, in a big step. The biggest risk of the waterfall method is that if there is a misunderstanding in the requirements at the beginning, there is no way to step back while the project runs, and this affects the project ending as well. These events can make the project more expensive, and the fix of the possible mistakes can take a long period of time. Typically one can say that waterfall method provides a structure, which suits best to traditional type of projects with little innovation. Projects, which need new improvements in technology, maybe not the best subject to a waterfall method.

3. Agile way for business change management in Dynamics AX

There is a different new way of implementing Dynamics AX, which is called Agile Implementation. Agile method is totally different from waterfall method. Agile implementation model is an iterative, incremental process for developing Microsoft Dynamics Solutions. This Project Type gives customers greater control over the final solution because they can quickly change the direction of solution development and implementation from one sprint cycle to the next. It focuses delivering the whole functionality in smaller chunks, within a series of smaller developing series. By delivering and accepting smaller parts of development, the risk of the customer will not get the solution needed is smaller. [9] If the developed solution does not fit into the needs of the customer, it can be easily and quickly fixed in the next chunk, which means that customers get involved into the project deeper. This also means, that customers get a deeper understanding of the project, resulting a higher quality of the product, and often resulting lower costs. The drawback of agile implementation is that it is useful only with strict material and time agreement.

Typically used at a single site requiring specific features and moderate to complex customizations [3]. The development phase in the agile execution operates in Scrum development model, life cycle is divided to 30 days sprint cycle and daily sprint cycle contains analysis, design, coding implementation and end up with solution test, and finalizing product specification.

The short description of the process is the following, Fig. 5.:

- Solution Backlog: the solution feature list is listed here.
- Release Backlog: identifying and prioritizing the set of features which are to be developed during the 30 day sprint, and determines all of the feature time estimation (story point).
- Sprint Backlog: it is the breakdown of release backlog compared to feature priority and estimation it could be 3-days to 30 days (daily sprint)
- User story: this is the place for descript feature business function, users (roles), and test script (less documentation).

- Defect Backlog: identifying and reporting system bugs.
- Stand-up daily meeting: what has been done today (status).

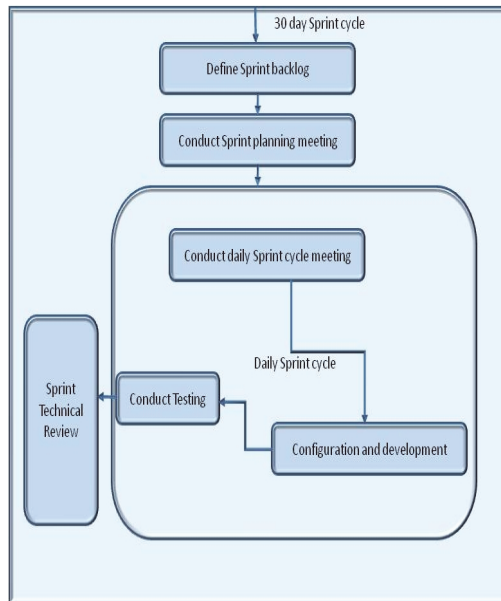


Figure 4. Sprint cycle

The agile implementation starts with a detailed business process analysis, and the declaration of high level requirements with fit-gap analysis. The output of the fit-gap analysis will become a main document of the project, it is called Solution Backlog. The Solution Backlog is a living and changing document, tracking the current business and project priorities along the project. Until this stage, the agile project looks like a usual project type.

At the stage of project execution the agile implementation is totally different. The traditional project is divided into two stages, Design and Development, the agile project is doing 30 day sprint cycles, shown in Fig. 5.

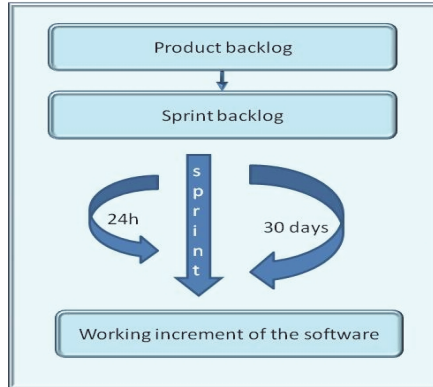


Figure 5. 24h and 30days sprint cycles

This 30 days sprint cycle contains a small part of the Solution backlog, which is called Sprint Backlog. Each requirement is split into small parts, no longer than 16 hours of development time, and connected to developers. During this every cycle, the project team does daily sprint meetings.

The goal of these meetings are the following: each team member shares what has been achieved since last meeting, what will be done until the next meeting and what issues affects the project.

The functionalities, which have been developed, are synchronized and built into daily builds. As soon as a requirement is reached, the sprint testing will be executed. If any change is needed after testing, this will be the task of next daily sprint cycle.

At the end of the sprint cycle, the Sprint Technical Preview is done, where the customer reviews, and signs the developed functionality. After the next sprint cycle is started, and iterated as many times as necessary, until the Solution Backlog is empty. At this point, the developer team does a detailed Solution Testing, to verify all the requirements are satisfied, and the whole solution is working perfectly according to the expectations. After the Solution Testing, the project resembles again to the standard project types, as the deployment status arrives and the whole solution goes into live.

4. Summary

An overview of Dynamics AX introduction methodology changes has been given in the previous sections. It is clear that speed of changing development and functional approaches depends on business. Therefore it is hard to estimate the success of business changes. A categorization of MS Dynamics change management is being built for core applications. Business issues and corresponding tasks required are registered and tracked for such cases. This tool will help decision-makers to make realistic estimations for necessary business changes.

It is clear now that the Agile Implementation opens a good framework for developments in organizations which requirements are volatile and fuzzy. The solution development is incremental, but the deployment is still done in one big step. There some

reasons, why Agile Implementation does this, but the strongest is the ERP data consistency.

Further questions will have to be answered, for example: why would the customers choose agile implementation over the well known waterfall methods? Are there enough benefits in using agile implementations over the waterfall methods, which has the advantage of the (almost) fix price and time? These issues have to be carefully answered before choosing implementation method. Despite of these questions, Agile Implementation method is completely different approach, which needs totally new way of thinking from the development team members. This implementation method has surely more chance to deliver a solution which suits the needs of the customer better.

References

- [1] Gupta, A. K.: *Quality Assurance for Dynamics AX-Based ERP Solutions: Verifying Dynamics AX customization to the Microsoft IBI Standards*, Packt Publishing 2008 ISBN-10: 1847192912
- [2] Pocius, M.: *Microsoft Dynamics Ax 2009 Development Cookbook*, 2009, ISBN: 1847199429
- [3] Hamilton, S.: *Managing Lean Manufacturing using Microsoft Dynamics AX 2009*, 2009 ISBN-10: 097925521X
- [4] Koop, R., Muris, E.: *Successfully Implementing Microsoft Dynamics(TM): By Using the Regatta® Approach for Microsoft Dynamics (TM)*, Springer 2007 ISBN-10: 3540715924
- [5] Andreasen, S.: *MORPHX IT, An introduction to Axapta X++ and the MorphX Development Suite*, 2007 ASIN: B001CJQU6G
- [6] Filicetti, J. *PMO and Project Management Dictionary*, PM Hut. (August 20, 2007): Retrieved 16 November 2009 <http://www.pmhut.com/pmo-and-project-management-d>
- [7] Greef, A., Pontoppidan, M. F., Olsen, S. D.: *Inside Microsoft Dynamics AX 4.0. Microsoft*, Microsoft Press E-Book, 2006 ISBN-10: 0-7356-2257-4
- [8] Phillips, J. R.: *Enhancing the Effectiveness of Organizational Change Management*, Human Resource Management, vol. 22(iss. 1/2), pp. 183-199, 1983 Retrieved 12/21/11 from <http://onlinelibrary.wiley.com> DOI: 10.1002/hrm.3930220125
- [9] Shankar, C., Bellefroid, V.: *Microsoft Dynamics Sure Step 2010*, 2011 ISBN-10: 1849681104
- [10] Muka L., Lencse, G.: *Developing a meta-methodology for efficient simulation of infocommunication systems and related processes*, Infocommunications Journal, vol. LXIII, no. 7. pp. 9-14, 2008
- [11] Muka, L., Lencse, G.: *Cooperating Modelling Methods for Performance Evaluation of Interconnected Infocommunication and Business Process Systems*, Proceedings of the 2008 European Simulation and Modelling Conference, 2008

- [12] Muka, L., Benkő, B. K.: *Meta-level performance management of simulation: The problemcontext retrieval approach*, *Periodica Polytechnica*, vol. 55, no. 1-2, pp. 53-64, 2011
DOI: 10.3311/pp.ee.2011-1-2.06
- [13] Mátrai, R., Kosztyán, Zs. T.: *A New Method for the Characterization of the Perspicuity of User Interfaces*, *Acta Polytechnica Hungarica*, vol. 9, no. 1, pp. 139-156, 2012
- [14] Ósz, R.: *Educational organization for new generation*, SAMI 2012, IEEE 10th International Symposium on Applied Machine Intelligence and Informatics, Herl'any (Szlovákia) Conference Proceedings, pp. 373-375, 2012
DOI: 10.1109/SAMI.2012.6208992