EXPERIENCES IN APPLYING AGILE SOFTWARE DEVELOPMENT PRACTICES IN NEW PRODUCT DEVELOPMENT

Aldo Dagnino[1], Karen Smiley[1], Hema Srikanth[2], Annie I. Antón[2], Laurie Williams[2]

[1] ABB US Corporate Research, {aldo.dagnino, karen.smiley}@us.abb.com
1021 Main Campus Drive, Raleigh, NC, 27606, USA
[2] North Carolina State University, {hlsrikan, aianton, lawilli3}@ncsu.edu
900 Main Campus Drive, Raleigh, NC, 27695, USA

ABSTRACT
Experiences with software technology development projects at ABB Inc. indicated a need for additional flexibility and speed during explorations of applying new technologies to future products. A case study was conducted at ABB to compare and contrast the use of an evolutionary-agile approach with a more traditional incremental approach in two different technology development projects. The study indicated benefits associated with the evolutionary approach with agile practices, such as streamlined documentation, increased customer involvement, enhanced customer satisfaction, increased capability to include emergent requirements, and increased risk management ability. This paper suggests that using agile practices during the Research and Development (R&D) phase of new product development contributes to improving productivity, to increasing value-added activities, to showing progress early in the development project, and to enhancing customer satisfaction. Another observation derived from this study is that by offering a carefully selected subset of agile practices, ABB R&D groups are more likely to successfully incorporate them into their existing processes.

KEY WORDS
New software-intensive product development, agile practices

1. Introduction

ABB Inc.[1] is a multi-national corporation that develops products, largely software-intensive, for the power and automation technology market segments. Before a decision is made to begin a new project, a feasibility study is carried out to ensure that a resulting product is likely to have good economic potential [1]. After a working prototype is developed at the Corporate Research Center (**CRC**) lab, this prototype is turned into a product, or "productized," by the Business Unit (**BU**). To ensure maximum efficiency, suitable lifecycles and processes need to be applied during the various development phases.

The **A**BB **S**oftware **P**rocess **I**nitiative (ASPI) group, an international team of in-house product development process engineers, acts as the internal ABB Corporate Engineering Process Group (CEPG) and helps BUs choose the most efficient product development processes to optimize time, quality, and functionality [2, 3, 4, 5, 6, 7]. The ASPI group has deployed research and development techniques to improve speed and flexibility in software-intensive product development.

Early in the year 2001, a two-stage Incremental Development Model (**IDM**) was created, tailored, and deployed by the ASPI group. After several technology development projects were carried out using this model, a need for further flexibility and speed sparked the creation of a more flexible and more agile Evolutionary Development Model (**EDM**) [5]. This paper reports on a case study that compares the incremental and the evolutionary models applied to two technology development projects at ABB. The projects were of similar size and comparable complexity. The estimated development time was under one calendar year for each project. The same team of five people developed both prototype-working systems (which served as the basis for the final products) in parallel.

Our research goal for this study was to identify the process model more suitable for technology development projects and ABB. Using the Goal-Question-Metric approach [8], we defined the research question and metrics that needed to be collected to answer the research question.

Question: Which process model is more effective for technology development projects: incremental or evolutionary?

To answer this question, we determined that five metrics should be collected:

a. Time spent on documentation and project planning
b. Customer involvement and satisfaction
c. Requirements volatility
d. Delivery of business value
e. Risk reduction

Details on how these measures were defined and gathered are provided below.

2. Application of Agile Practices in ABB Research & Development

In this section, we discuss ABB's product development cycle. We also provide information on the software processes that were developed and used for technology development projects.

2.1. The ABB Product Development Cycle

The product development cycle at ABB includes three primary phases, as shown in **Figure 1**. During the ***Feasibility Study Phase***, a product idea is evaluated based on its potential business value and technological feasibility.

The ***Technology Development (TD) Phase*** is associated with the activities performed to evaluate in greater depth the technological feasibility and business value of the proposed product. These initial development activities are primarily executed by the ABB CRCs, with involvement from the ABB BUs. TD projects at ABB typically last 6-12 months. The primary deliverable from the TD phase is a *working prototype* system that has the primary functionality of the intended final product.

The third phase is the ***Product Development (PD) Phase***, during which the prototype system is further enhanced and developed into a *product* by the BU that is responsible for sales to external customers and for supporting and maintaining the final product.

ABB uses a "Gate Model" (**GM**) [1] to evaluate the business value of potential new products, to help ensure consistent execution and sound decision-making throughout the development lifecycle. Formal business decision processes like the ABB GM generally consist of different development stages, separated by business-decision evaluation points known as ***gates***. **Figure 1** shows the gates (numbered 0 through 7) of the ABB Gate Model, and how the TD and PD phases are synchronized. Using the ABB Gate Model, achievement of predetermined pre-gate milestones is evaluated and business decisions are made on whether the project should be continued, amended, or stopped.

2.2. ABB Incremental Product Development

In early 2001, an evaluation of the software development models used at ABB revealed that the traditional "waterfall" or Big Design Up Front (**BDUF**) [2] model was most commonly used within ABB development groups. However, the BDUF models did not always suit the needs of rapid-development TD projects within ABB. The exploratory nature of the projects and associated volatility of the requirements exacerbates the high documentation burden of BDUF models, frustrating the development teams and increasing the costs of adapting to shifts in technological direction based upon early discoveries from the feasibility assessments.

For those projects in which the requirements are not highly volatile and the scope of the project can be defined early in the project, using an IDM is appropriate. Traditionally, an IDM had been used in TD activities at ABB. This incremental process incorporated small releases (in two increments of 2-4 months each) and a sound software development discipline. Given the desired time-to-market, the period from Gates 2-5 was expected to span a short period of time, so using more than two increments was viewed as impractical. However, an assessment of the amount of effort invested in documenting the requirements, managing the project, and preparing for (sometimes repeated) gate meetings for projects in the TD phase revealed that a "leaner" development model might be beneficial. Additional analyses showed that there was also significant room for improvement in speed, flexibility to accommodate emerging requirements, customer focus, customer engagement, and developer-friendliness. As a result of this evaluation, two initiatives were undertaken:

- Continued process streamlining and reduction of the artifacts associated with the incremental model; and
- Creation of an agile process alternative.

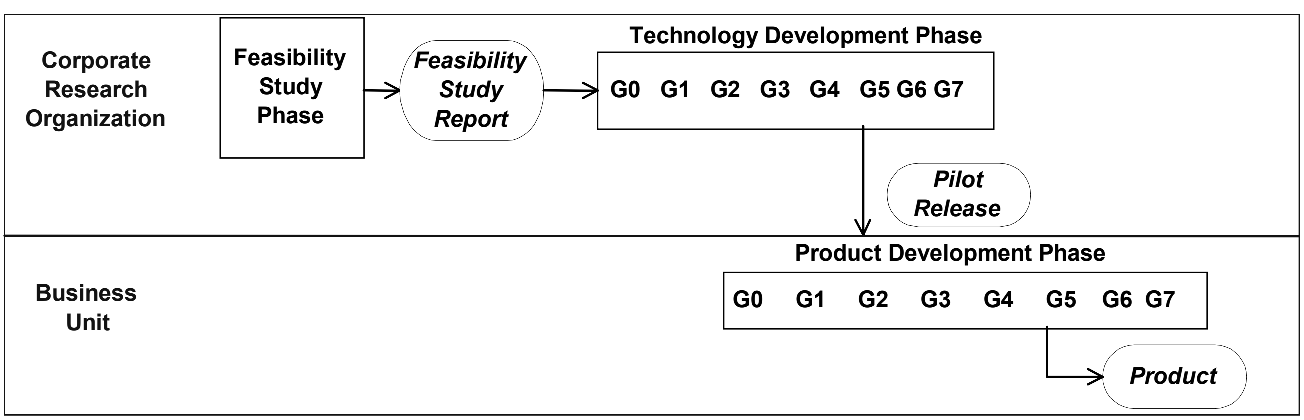

Figure 1. ABB Product Development Phases and Gates

The second initiative will be discussed in the following two subsections.

2.3. Agile Product Development: ADEPT

Agile Development in Evolutionary Prototyping Technique (ADEPT) [5, 10] was developed to increase agility and maturity on TD projects. Although several existing agile processes and lifecycle models [2, 3, 6] were considered, we decided to use a traditional evolutionary lifecycle model as a base model and add a subset of agile practices to that model to create ADEPT [5, 9]. It has been our experience that ABB development groups who are accustomed to more traditional software development approaches are able to accept more easily a gradual approach to incorporating agile practices, as provided by ADEPT, versus attempting to transition immediately to a fully agile methodology. **Figure 2** provides a graphical representation of ADEPT. The ovals portray process activities, curved rectangles represent the artifacts, thicker arrows indicate control flows through the process, and the narrower arrows indicate data flowing as a result of the activities.

The ADEPT model has three primary stages:

- During the **Project Evaluation** stage, team members meet with the customer to negotiate project scope, gather system features, assign feature responsibility to team members or groups within the project team, and negotiate features to be implemented in the upcoming iteration.
- The **Feature Development** stage encapsulates the activities performed in an ADEPT iteration. During this stage, developers may employ "agile" practices to plan, design, develop, test, and integrate features. At the beginning of the stage, the group meets with the customer to identify and negotiate the requirements to be implemented in the iteration. After the negotiation process, each team member or group concurrently plans, implements, and tests the requirements for the assigned features. During the initial customer evaluation, the customer suggests additions or revisions. These suggestions are addressed by applying additional feature development stages or iterations as needed.
- During the **Project Completion** stage, the development team validates the system, delivers the system (pilot or prototype) to the customer, and conducts a project evaluation. The working prototype is given to the BU responsible for productizing it.

By having an evolutionary lifecycle model, an important agility principle was included in ADEPT: the ability to incrementally enhance the functionality of the software and thus allow for flexible adaptation to changing requirements.

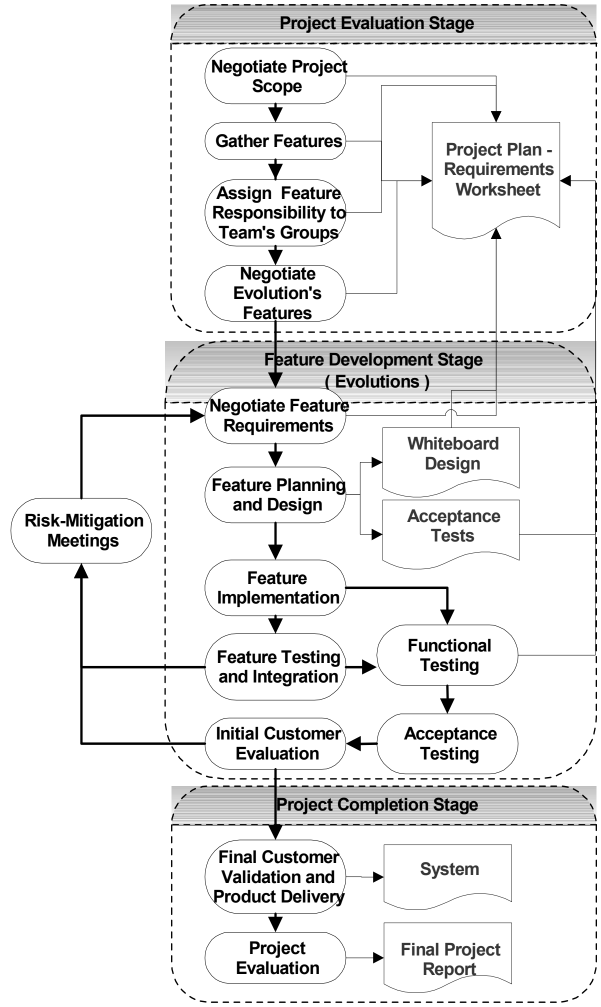

Figure 2. ADEPT Lifecycle Model

2.4. Comparisons and Contrasts Between IDM and ADEPT

A comparison between the IDM and the ADEPT model is presented in **Table 1**, highlighting diverse aspects of both models as applied on TD projects at ABB.

2.5 Case Study

A case study to compare the use of ADEPT and IDM was conducted using two projects at ABB. Project A used the more traditional IDM, and Project B used ADEPT.

The goal of Project A was to develop a working prototype system for monitoring key performance indicators (KPI) of critical manufacturing processes (e.g. winding, assembly, test) and report the status from the shop floor to the boardroom for continuous process improvements and benchmarking. The project was developed using a new ABB integration platform [9] to monitor all manufacturing processes for a power transformer factory. The working prototype system was

Criteria	IDM (Incremental Development Model)	ADEPT (Evolutionary Development Model [EDM])
Lifecycle model	Incremental	Evolutionary
Number of Cycles	2 increments	Typically 4 to 5 iterations
Cycle length	Approximately 3 months	Approximately 1 month
Planning focus	Entire project (G2-G5)	Next cycle (emergent plans)
Documentation 'weight'	Traditional ('heavy') with some adjustments for research nature and short timetables	Lightweight (spreadsheets, whiteboards, pictures)
Customer involvement	Requirements elicitation (G0-G1) and sign-off at Gate 2; acceptance testing before Gate 5	Active involvement throughout the project in identifying, negotiating, and evaluating results for each cycle
Inspections	Traditional - peer reviews of document deliverables, code reviews	Traditional - peer reviews of document deliverables, code reviews
Risk management	'Active' (goal), risk sheets updated at least once per gate	Weekly formal risk mitigation meetings, daily informal meetings, and risk sheets
Customer Satisfaction	Limited customer involvement after Gate 2; customer satisfaction often does not meet expectations	Continuous involvement of customers and shorter development cycles yields increased customer satisfaction
Development Team Satisfaction	Team complains about spending a lot of time documenting project	Team happy with customer and with progress shown in project

Table 1. Comparison of IDM and ADEPT

implemented in one factory in Switzerland, and the solution will be scalable throughout the Business Area (25 facilities) to enable real-time enterprise management capabilities. Additionally, the solution will be capable of supporting forecasting and scheduling in the future.

Project B was developed using the same integration platform used in Project A. The goal of Project B was to develop a working prototype system to assist power transformer sales hubs to utilize production and engineering resources globally. The new working prototype system links an ABB customer, one of ten ABB sales hubs, four of 25 ABB manufacturing sites, and a supplier, with real time information flows among them. The new system allows the ABB sales hub to better utilize global production and engineering resources.

Each factory has one or more systems to manage forecasting, quoting, product engineering, order management, scheduling and manufacturing execution, finished goods, inventory, materials, and supply chain management. Project B's working prototype system provided the vision for the final system: a single viewpoint into the critical information needed for engineering and manufacturing resource management around the world. This single point of view will then allow economies of scale to flourish. Critical resources and supplies required to manufacture power transformers will now become shared resources.

Projects A and B were selected for this case study because of their similarities, which included:

1. Both teams had the same developers working together, although different team members served as project leaders. Both project teams were geographically dispersed: the software developers were all working at the same location in Raleigh, North Carolina, but the team members primarily responsible for the requirements development activities (the project managers) were located in Athens, Georgia, and traveled to sites that served as pilots for the working-prototype systems.
2. The objective of both projects was to deliver a working prototype application system to improve ABB's internal manufacturing processes.
3. The customers for both projects were located in the same ABB BU, which is dedicated to the manufacture of power transformers. The internal ABB BU customers for both projects (plant operators and sales personnel in distribution transformer plants located in the US, Switzerland, and Canada) were reasonably accessible to the requirements development teams.
4. The estimated effort and expected elapsed time for both projects were very similar (1.5 person-years effort and 8 calendar months duration).
5. Similar new technology was employed in both projects, as described above.
6. The projects commenced at the same time and progressed in parallel through their development lifecycles.
7. ASPI would be actively coaching both projects.

The case study was conducted as follows. Two people in each development team (the technical writer and the project leader) regularly collected project data and supplied it to ASPI for analysis. Data were collected twice a month during the case study, and included activity time records, monitoring of how often software requirements were modified, analysis of customer satisfaction, project progress measures, and artifacts used in the projects.

As the projects were completed and there were indications that the use of ADEPT brought significant benefits, additional TD projects were started with ADEPT.

A summary of the qualitative observations carried out during the case study is presented below in **Table 2**, organized according to several topics. Each topic describes specific problems encountered during the 18 months of TD projects in ABB CRCs prior to the inception of ADEPT, how ADEPT addresses these problems, and the observations identified by comparing Project A (which used the more traditional IDM) and Project B (which used ADEPT). These topic areas are also discussed in further detail below.

2.5.1 Streamlined Documentation and Project Planning

The team for Project A spent 150 hours developing the project plan and 330 hours documenting the project requirements, for a total of **480 hours** of documentation. The total effort for Project A was around 2400 hours. Since the second pilot customer did not find the proposed solution acceptable, this development and testing effort was for supporting only a single pilot customer. The team for Project B took 110 hours to develop the project plan and 165 to document the requirements, for a total of **275 hours** of documentation effort using ADEPT. The total effort for Project B was around 2700 hours to develop and deploy for both pilot customers. A qualitative estimation of the complexity of both projects suggested that they were comparable. Therefore, the reduction in the documentation time is attributed to the "lighter" approach defined in ADEPT, which prescribes the use of non-traditional documentation templates such as spreadsheets, white-boards, and pictures.

2.5.2 Customer Satisfaction

When IDM was applied to developing Project A and previous ABB technology projects, customers were typically involved only at the beginning of the project (to obtain requirements) and at the end of the two increments (to show resulting functionality). This approach increased the possibility of miscommunication, delays, deficiency in providing timely feedback during the development phase, and overall lack of flexibility during the development phase. The ADEPT methodology includes an explicit mechanism to facilitate customer involvement by increasing the frequency of meetings with the customer. The use of ADEPT allowed the team for Project B to have continuous feedback from the customer and adjust the activities as the project progressed. The observed increase in customer satisfaction on Project B, compared to Project A, is directly attributable to this increased involvement and feedback.

2.5.3 Accommodation of Volatility in Requirements

Volatility of the requirements was measured by counting the number of times a requirement was changed, and by counting the number of new requirements identified during each cycle. When Project A was evaluated at Gate 2, the requirements were perceived to be stable (based upon the degree of uncertainty expressed by the customers and relevant stakeholders in defining the desired functionality of the proposed system), and the stakeholders considered two increments based on approved requirements to be acceptable. However, as the

Factors	Validation Approach	Incremental Model (IDM) Results	ADEPT Results
Streamlined Documentation	Recorded time spent by team members in documentation	20% of project time	10% of project time
Customer Satisfaction	Interviews with customers	50% of customers completely satisfied (1 out of 2)	100% of customers completely satisfied (2 out of 2)
Requirements Volatility	Interviews with Project Manager	Limited capability to accept changes in requirements	More adaptable to accept changes in requirements: 50-60% of requirements changed
Business Value	Validation with the customer(s)	Value shown in two increments	Value shown in four iterations
Risk Reduction	Continuous monitoring	Risks evaluated at two points in time (Gate 3 and Gate 4)	Risks evaluated every week
Number of Cycles	Recorded number of cycles from project	2 iterations	4 iterations
Cycle length	Recorded length of cycles in projects	3 months	1 month average

Table 2. Validation Approach for IDM and ADEPT

project progressed, requirements changes surfaced repeatedly, and the team was not able to accommodate all of them. We observed significantly increased effort by the team for requirements documentation, as well as the dissatisfaction of one of the two pilot customers for Project A. Both of these observations are likely to be attributable to the decreased flexibility and increased overhead of the IDM. ADEPT inherently allows a development team to be adaptable and re-evaluate the requirements for each iteration through the risk mitigation meetings. By doing detailed planning only on the features and requirements to be implemented in a specific cycle, ADEPT enabled the team for Project B to incorporate changes in requirements at a later stage with less impact to the project.

2.5.4 Business Value

Implementation of requirements using IDM on Project A showed progress in two increments of about four months, and therefore showed customer value twice throughout the project following Gate 2. Using the more frequent deliveries that ADEPT incorporates (about a month apart), the developers on Project B were able to show initial value more quickly after Gate 2, and showed added value more often during implementation, without wasting time on throwaway mock-ups, because ADEPT focuses on working software from the start.

2.5.5 Reduced Risks

Previous projects that followed IDM often did not focus on risk management until shortly before each gate assessment, and then strived to update their risk sheets all at once. The weekly team meetings advocated by ADEPT included risk identification and evaluation, and risk mitigation planning. In the case study, the team that used ADEPT was observed to demonstrate a higher level of awareness of risks and a more proactive approach to managing them. As a result, the ADEPT team (Project B) was more effective in minimizing the impact of risks on their project than the IDM team (Project A).

3. Conclusion

Project B using ADEPT was successful in satisfying two pilot customers in 2700 effort hours, with 10% of the project effort dedicated to documentation and planning. Project A using IDM was only able to satisfy one of two pilot customers in 2400 effort hours, with twice the effort in documentation and planning, in a comparable project.

This study suggests that ADEPT may facilitate the management of technology development projects with their inherent uncertainty, due to their need for greater adaptability. This is because ADEPT increases the flexibility with which the development team can respond to changes in requirements, enhances customer involvement and communication, increases team productive time, and allows showing progress frequently. This study also suggests that by gradually introducing agile practices, a development team may be more likely to incorporate them into their internal development process.

4. Acknowledgements

The authors wish to acknowledge the insights and contributions of James Pierre and Guojun Zhu (ABB Project Managers) and the whole ABB development team.

References:

[1] C. Wallin, F. Ekdahl, and S. B. M. Larsson, Integrating business and software development models, *IEEE Software,* November-December 2002, 28-33.

[2] K. Auer and R. Miller, *XP applied* (Reading, Massachusetts: Addison Wesley Professional, 2001).

[3] A. Cockburn, *Agile software development* (Reading, Massachusetts: Addison Wesley Longman, 2001).

[4] M. B. Chrissis, M. Konrad, and S. Shrum, *CMMI®: guidelines for process integration and product improvement* (SEI Series in Software Engineering, Pearson Education Inc., 2003).

[5] A. Dagnino and K. Smiley, Agile development in evolutionary prototyping technique (Internal ABB document, 2002).

[6] J. Highsmith, *Agile software development ecosystems* (Boston, MA: Addison Wesley, 2002).

[7] M. C. Paulk, B. Curtis, and M. B. Chrissis, *Capability maturity model for software version 1.1* (SEI CMU/SEI-93-TR, February, 1993).

[8] V. Basili, G. Caldiera, and D. H. Rombach, "The Goal Question Metric Paradigm," in *Encyclopedia of Software Engineering*, vol. 2: John Wiley and Sons, Inc., 1994, pp. 528-532.

[9] L. Bratthall, R. van der Geest, H. Hoffman, E. Jellum, Z. Korendo, R. Martinez, M. Orkisz, C. Zeidler, and J. S. Andersson, Integrating hundreds of products through one architecture - the Industrial IT architecture. *Proceedings, International Conference on Software Engineering,* 2002.

[10] H. Srikanth and A. I. Antón, *Towards achieving a balance between planning and agility in software development* (NCSU Technical Report TR-2003-11, April 5, 2003)