What Makes Information System Developers Produce Defective Information Systems For Their Clients?

Tero Vartiainen
Department of Information Processing Science, University of Oulu, tero.vartiainen@oulu.fi

Mikko Siponen
Department of Information Processing Science, University of Oulu, mikko.t.siponen@jyu.fi

Follow this and additional works at: http://aisel.aisnet.org/pacis2012

Recommended Citation
http://aisel.aisnet.org/pacis2012/107

This material is brought to you by the Pacific Asia Conference on Information Systems (PACIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in PACIS 2012 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
WHAT MAKES INFORMATION SYSTEM DEVELOPERS PRODUCE DEFECTIVE INFORMATION SYSTEMS FOR THEIR CLIENTS?

Tero Vartiainen, Department of Information Processing Science, University of Oulu, tero.vartiainen@oulu.fi

Mikko Siponen, Department of Information Processing Science, University of Oulu, mikko.siponen@oulu.fi

Abstract

Quality problems in information systems and software (IS/SW) are common technological problems. A number of different development methods and mature models have been proposed to address these challenges. Recent research on agile and lean SW development methods suggests that companies often speed up IS/SW development by omitting necessary quality procedures. While agile methods are commonly used in SW development, however, we have found no research that specifically explores why deficient IS/SW emerges. We therefore interviewed IS/SW developers in two rounds (n = 20 and n = 9) in order to inductively build a theoretical model that explains this issue. We identified two processes that lead to deficient IS/SW and the factors that affect the emergence of this problem. Results suggest that economic factors pressure developers to omit important quality controls in different stages of IS/SW development. Our resulting theoretical model belongs to the process theory paradigm of dialectical process theory. Based on our findings, implications for practice and research are proposed. Our results shed new light on how developers’ omission of quality controls can be overcome so that the quality of IS/SW can be improved.

Keywords: Gray area, information systems and software development, dialectics, process theory
1 INTRODUCTION

Information systems and software (IS/SW) have become a part of everyday life in developed countries. Although the number of IS/SW is on the rise and IS/SW is increasingly embedded in different devices, the number of SW errors has not decreased, as widely reported in the literature (Gibson & Senn 1989; Kafura & Reddy 1987). While ordinary users perceive these as program or OS crashes as well as delays in services, deficiencies in SW may lead to serious problems in the case of critical safety SW (Leveson & Turner 1993). Vartiainen et al. (2011) believe that an increase in the current trend of applying agile IS/SW development methods may pave the way for even more SW errors (Martin 2003; Baskerville & Pries-Heje 2004; Baskerville et al. 2003). They maintain that this may happen because SW companies intentionally omit quality steps in IS/SW development in order to hit the market earlier than competitors, meet tight deadlines, and to cut out “unnecessary” documentation (Abrahamsson et al. 2003; Ahonen & Junttila 2003; Baskerville & Pries-Heje 2004; Baskerville et al. 2003; Vartiainen et al. 2011).

While previous studies on IS/SW have reported that developers omit IS/SW tests and quality checks to achieve faster product release (Baskerville & Pries-Heje 2004; Baskerville et al. 2003), no explicit research has focused on why this phenomenon happens (Vartiainen et al. 2011). To address this research gap, we interviewed IS/SW developers in order to understand what motivates them to produce deficient IS/SW.

The rest of our paper is organised as follows. The second section presents four paradigms of process theories, and the third section discusses the research method used. The fourth section presents the results of our interviews, and the fifth section analyses the results of our study. Finally, the sixth section presents the conclusions.

2 FOUR PARADIGMS OF PROCESS THEORIES

Van de Ven (1992) and van de Ven and Poole (1995) have developed a taxonomy of four ideal types of process theories: (1) life cycle, (2) teleology, (3) dialectic, and (4) evolutionary process. The purpose is to be able to evaluate the process theories against this framework (van de Ven & Poole 1995). In Table 1, we present the four paradigms of the process theories (adapted from van de Ven 1992 and van de Ven & Poole 1995). These meta-views of the process theories underlie our inductive process model.

<table>
<thead>
<tr>
<th>Four Categories of the Process Theories</th>
<th>The Goal of Development Progression is Known and Predetermined</th>
<th>Predetermined Stages</th>
<th>Step-by-Step Progression</th>
<th>Types of Progression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Life Cycle</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Unitary</td>
</tr>
<tr>
<td>Teleology</td>
<td>Yes</td>
<td>No; they can only list possible paths</td>
<td>No</td>
<td>Multiple cumulative progression</td>
</tr>
<tr>
<td>Dialectic</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Recurrent convergence of multiple divergent progression</td>
</tr>
<tr>
<td>Evolutionary</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Recurrent, cumulative and conjunctive sequence of variation, selection, and retention events</td>
</tr>
</tbody>
</table>

Table 1. The four paradigms of process theories (Van de Ven 1992; van de Ven & Poole 1995).
2.1 Life-Cycle Process Theories

Life cycle process theories hold that an entity has a predefined logic, program, or law that regulates its development toward a certain predefined goal. For example, humans are born as babies, mature into children, and eventually become adults. The external environment may influence the development of an entity, but such entity is always influenced by this predefined logic that regulates its development (van de Ven & Poole 1995). In addition, life cycle process theories have a unitary sequence, meaning that these theories affirm that not only predefined goals, but also predefined trajectories, exist (van de Ven 1992). These development paths have to take place in a certain fixed order (e.g., stage A always comes before stage B), and each stage is a necessary precursor for the next stage (e.g., stage A needs to be completed before one can proceed to stage B) (van de Ven & Poole 1995). The knowledge view of life cycle theories is cumulative; for example, in stage \( n \), an entity retains all the characteristics it has acquired from all previous stages (van de Ven & Poole 1995).

2.2 Teleological Process Theories

Teleological process theories assume that an entity adapts its behaviour either by itself or in interaction with others and correspondingly constructs its own desired end state (van de Ven 1992). These theories assume freedom because the developing entity has the ability to pursue whatever goals it prefers (van de Ven & Poole 1995). While life cycle process theories assume that a predefined sequence of events or stages exists, teleological process theories do not, because for such theories, only the possible paths (trajectories) can be noted (van de Ven & Poole 1995). Finally, external factors may introduce instability that pushes the entity toward a new trajectory (van de Ven & Poole 1995).

2.3 Dialectic Process Theories

Dialectic process theories have origins in the work of Marxism and Hegel, in which an entity (herein, the organisation or IS developer) lives in a pluralistic world. In this pluralistic life, the entity is faced with colliding events and rivaling forces with contradictory values that compete with one another for control or domination (van de Ven 1992). An example of dialectic process theories is the dialectical theory of human development by Riegel (1975). Development from one stage to another occurs when the entity encounters a value contradiction between rivaling forces (thesis and antithesis); confronting this tension helps create synthesis, which, in turn, assists the entity to move to the next stage. Often, the synthesis is a new solution that differs from the thesis and the antithesis (van de Ven & Poole 1995). However, in reality, things do not always happen according to dialectical theory. Sometimes, the opposite group has enough power to fully overthrow the dominant group so that no synthesis exists, and sometimes, the antithesis is actually a win-win situation (van de Ven & Poole 1995).

2.4 Evolutionary Process Theories

Evolutionary process theories suggest that development occurs through a continuous process of variation, selection, and retention (van de Ven 1992). As in biological evolution, the process of selection happens through competition among forms, and the environment selects those forms that optimise or are best suited to the resource base of an environmental niche (van de Ven 1992). Evolutionary process theories assume that users progress through certain stages, as they follow, for example, the social Darwinian selection process.
3 RESEARCH DESIGN

We applied a qualitative and interpretive approach (Patton 1990; Trauth 2001) in studying questionable and dubious work practices of IS/SW developers. First, we interviewed 20 IT professionals using a phenomenographical method (e.g., Marton 1986). From these interviews, we identified the problem of deficient IS/SW and the reasons that cause it. At this point, we determined that the dialectical process theories (Van de Ven 1992) helped us understand the phenomenon of deficient IS/SW. The reason for this was that from the interviewee’s descriptions, we found that there is a variety of rivalling forces affecting the decision to produce defective information systems for clients. We did not identify any predefined sequence of events that is typical in life cycle theories. We also did not identify a continuous process of variation, selection, and retention that is typical of evolutionary theories. Teleological theories were another possibility as the act of producing defective information systems could be perceived as an end state. However, we found that simplification of the conflict to fit the format of thesis–antithesis and synthesis would describe the problem in a meaningful way.

Dialectical process theories hold that change happens through certain factors. To obtain further information on these factors, we conducted a second round of interviews. The interview questions were targeted at determining the forces and counterforces that affect the development of deficient IS/SW. Six of the interviewees from the first round were asked specific questions with respect to the themes they raised during the previous questioning. In addition, three new IT professionals were interviewed. In summary, the following specifying questions were asked of the nine interviewees (the interviewer also asked other probing questions during the interview):

“Describe how the situation evolved into the practice you described” and “What forces and counterforces existed in the process you described?”

The age and gender (F = female, M = male) of our respondents were as follows: 42/F, 57/M, 63/M, 51/M, 56/M, 55/M, 34/M, 32/M, and 42/M. They were coded as Subject 1 through 9 in Section 5 of this report.

During the analysis of the interviews, we identified the stages of information systems and software development (ISSD), the major factors that contribute to deficient IS/SW, the gray areas in the practice described, and the barriers involved. The next section describes our results.

4 AN EXPLAINING THEORY OF WHY DEVELOPERS PRODUCE DEFECTIVE INFORMATION SYSTEMS FOR CLIENTS

The theoretical model we created, which explains why developers intentionally produce defective IS/SW, consists of two main parts: (1) the thesis and the antithesis that affect what happens during the stages of ISSD and (2) that eventually lead to the production of defective IS/SW for clients. Table 2 lays out the theoretical elements that were used in our model.

Figure 1 summarises three trajectories that were affected by two forces—the economic pressure of the client to obtain benefit (or to maximise profits) (thesis) and the similar pressure of the client (antithesis). Figure 2 presents one trajectory that is affected by the economic pressure of the client to obtain benefit (thesis) and the developer’s inclination to produce high-quality work (antithesis). The trajectories differ from one another in that the barrier that is not overcome (e.g., inadequate requirements definition) emerges in different stages. Failure to overcome a barrier means that the client eventually receives defective IS/SW.
The production of the aim to produce high-quality products (e.g., coders) channelled within as good results as possible with the resources available as possible to get the work done (i.e., when producing a SW product) as prescribed in the contract clearly exists. Similarly, the client needs to get as good results as possible with the resources channelled into the project. In addition, a contradiction within the provider firm may exist when those who implement the results (e.g., coders or analysts) seek to uphold high quality, while management aims to promote economic benefits, which means that they want the production costs to be minimal. In this case, the economic consideration may dominate the aim to produce high-quality products. Figure 1 presents the trajectories that describe the production of defective results for the client.

Table 2: Theoretical elements of the model.

<table>
<thead>
<tr>
<th>Element</th>
<th>Theoretical Background and Definition</th>
<th>Description in Our Research</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thesis</td>
<td>A force, which is a key characteristic of dialectic process theories (van de Ven &amp; Poole 1995)</td>
<td>It is a force that gives motivation to practices. The need of firms to obtain profit from business partnerships is an example of a force.</td>
</tr>
<tr>
<td>Antithesis</td>
<td>A counterforce for the thesis, which is a key characteristic of dialectic process theories (van de Ven &amp; Poole 1995)</td>
<td>It is a counterforce for the thesis. The integration of the economic pressures of two firms in order to derive benefit is an example of a case with both force and counterforce.</td>
</tr>
<tr>
<td>Stage</td>
<td>A concept of stage (Weinstein et al. 1998)</td>
<td>The following ISSD stages have been found critical in producing deficient IS/SW: requirements definition, implementation, part testing, and integration testing. Stages are cumulative, as the IS/SW develops stage by stage.</td>
</tr>
<tr>
<td>Barrier</td>
<td>A barrier (Weinstein et al. 1998) that occurs in a particular stage</td>
<td>In our study, a barrier refers to the possibility of failing in conducting a good practice, such as in comprehensive testing. If a barrier exists, a gray area of practice emerges. If the barrier is overcome, the gray area of practice is avoided.</td>
</tr>
<tr>
<td>Transition</td>
<td>Movement from one stage to the following stage (van de Ven 1992)</td>
<td>In our study, we have transitions from one stage to another.</td>
</tr>
<tr>
<td>Trajectory</td>
<td>Description of stages and transition from one stage to another</td>
<td>We have four trajectories that all lead to defective IS/SW.</td>
</tr>
<tr>
<td>Synthesis</td>
<td>A new solution stemming from the struggle with the thesis and the antithesis (van de Ven &amp; Poole 1995)</td>
<td>A gray area of practice emerges as a result of a force and a counterforce affecting the stage and the failure to overcome the barrier.</td>
</tr>
</tbody>
</table>
Figure 1: Three trajectories that lead to the production of defective IS/SW for the client (an interplay between the provider and the client).

<table>
<thead>
<tr>
<th>Thesis</th>
<th>Trajectory 1</th>
<th>Trajectory 2</th>
<th>Trajectory 3</th>
<th>Antithesis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider: Economic pressure to follow what is stipulated in the contract and to avoid using too many resources for the client</td>
<td>Requirements definition is not adequately completed as it costs too much</td>
<td>Not all bugs are fixed because of schedule limitations and poor testing practices</td>
<td>Incomprehensively tested products are sent to the market</td>
<td>Client: Economic pressure to obtain beneficial results (or high-quality products)</td>
</tr>
</tbody>
</table>

- Requirements definition
  - Inadequate requirements definition
  - Implementation
  - Part testing
    - Not all bugs are identified and reported
  - Integration testing
    - Not all bugs are identified and reported

Synthesis Producing defective information systems for the client
Figure 2: A trajectory that leads to the production of defective IS/SW for the client (an interplay between the provider and the developer).

**Thesis**
Provider:
Economic pressure to follow what is stipulated in the contract and to avoid using too many resources for the client

**Antithesis**
Developer:
Pressure to do one’s job and the desire to produce high-quality results

**Trajectory 4**
The developer is pressured to produce low-quality results in order to save on resources.

**Requirements definition**

**Implementation:**
The developer confronts the pressure to do the job quickly and to avoid wasting resources by producing low-quality results.

**Part testing**

**Integration testing**

**Synthesis**
Producing defective information systems for the client
4.1 Trajectory 1: Requirements Definition is Not Adequately Completed as It Costs Too Much

A contradiction exists between how the provider and the client fulfils a contract, what the future IS/SW should address, and how both parties address the practical complexity of IS/SW use. Subject 2 described the entire process from obtaining bids for offers to producing offers and implementing what is stipulated in the contract. Evidently, only those features (requirements) that are prescribed in the contract are implemented and nothing else. In this way, the complex reality of IS/SW use may be ignored, which means that defective use or error-prone use is not taken into account in the development of IS/SW. Prior to contract signing, all contract stipulations should be carefully reviewed to avoid business losses.

According to Subject 2, ‘The client expects that the IS developed is fully efficient. This means that the system should respond to faulty use in a logical way. However, this feature is seldom specified in offers ... This means that the system does exactly what is specified in use cases ... In the requirements definition, we aim for the removal of all sentences that might lead to a tenfold workload, for example … You constantly have this dialog with specialists that they should not write so-called minefields so that we do not promise clients [too much].’

The definition of comprehensive requirements was perceived by Subject 6 as important in the achievement of successful IS/SW development. However, coming up with a usable and effective requirements definition entails the use of resources. If the requirements definition is inadequate, problems will ensue in the long run.

According to Subject 6, ‘When we went through these ERP projects, we noticed that the requirement definitions were inadequate. This will certainly lead to problems … Costs are entailed for a sufficient requirements definition. You cannot avoid costs. If we think about a counterforce, one possible factor is money, and another is the fact that the job will not be good.’

4.2 Trajectory 2: Not All Bugs are Fixed because of Schedule Limitations and Poor Testing Practices

When implementing the code, the coder typically conducts unit testing, which is then followed by integration testing. The testing practices of a firm affect the identification, reporting, and fixing of bugs, along with adherence to the set schedule. In the following transcript, Subject 1 raised the issue of how well test plans are made.

According to Subject 1, ‘It again originates from schedule limitations. You have to deliver [the results], and then the testing is abandoned for one reason or another. In the processes, testing is not so well planned and integrated into them. Perhaps, extensive testing is not part of the firm’s practices at all … You may not properly identify the bugs because of the way your testing plans are made … You may know about the existence of a bug [in the SW], but you consider it insignificant, so you do not report it to your client. Therefore, you are acting questionably.’

4.3 Trajectory 3: Incomprehensively Tested Products are Sent to the Market

In IS/SW business, strong pressure to immediately send products into the market exists. How well products are tested depends on the availability of resources. In the following transcript, Subject 5 raised the issue of incomprehensively tested products sent to the market: ‘Incomplete products that are not polished to perfection are sent to the market. This is a reality in [the] software business.’

4.4 Trajectory 4: The Developer is Pressured to Produce Low-Quality Results in Order to Save on Resources

Developers are confronted with scheduling and other forms of business pressure so that they are not able to produce high-quality work. High-quality work is costly and is perceived by management as a waste. According to Subject 2, ‘Business demands that you cannot do [high-quality work]. This has
been a constant battle between specialists’ intent to do what is right and business requirements that have to be met at a certain price.’

Programmers also face schedule limitations because they have inadequate time to fix the existing bugs. Subject 1 described a scenario in which a programmer confronts schedule pressure in fixing bugs: ‘How can [the programmer] deal with the pressure when bugs are found and he/she knows that the schedule is now dependent on him/her?’

### 4.5 Summary

Table 2 summarises the stages, the barriers, and the underlying forces that affect the gray areas of practice. In each ISSD stage, a barrier has to be overcome in order to avoid the gray area of practice. The economic pressure faced by the provider affects the entire process and also influences the economic pressure of the client and the intent of the developer to produce high-quality results.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Barrier</th>
<th>Thesis</th>
<th>Antithesis 1 and 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements definition</td>
<td>Inadequate requirements definition</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Implementation</td>
<td>The developer confronts the pressure to do the job quickly and to avoid wasting resources by producing low-quality results</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Testing a part of the code</td>
<td>Not all bugs are identified and reported</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Integration testing</td>
<td>Not all bugs are identified and reported</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

*Table 3: Summary of the phases, the gray areas of practice, and the underlying forces (thesis and antithesis) that affect these gray areas; X = the particular thesis or antithesis affects the emergence of the gray area of practice.*

### 5 DISCUSSION

In this section, we highlight the major findings of our study. First, we found two instantiations for our thesis—the antithesis and synthesis for the gray area of practice in the development of defective IS/SW for clients. The thesis–antithesis perspective that leads to the synthesis accords with the dialectic process theories (van de Ven & Poole 1995). From the provider’s viewpoint, there is an economic pressure to follow what is stipulated in the contract and to avoid using too many resources for the client (thesis). In the same manner, from the client’s viewpoint, there is economic pressure to obtain beneficial results/products (antithesis). Three trajectories that represent this interplay were found. In these trajectories, if barriers—such as inadequate requirements, not identifying and reporting bugs, and leaving the IS/SW untested—are not overcome, a gray area of practice (synthesis) emerges, leading to producing defective IS/SW for the client. In other studies on IS/SW, it has been determined that the developers do omit quality steps in IS/SW development. The major reason for this is the pressure to hit the market earlier than competitors (Abrahamsson et al. 2003; Ahonen & Junttila 2003;
Baskerville & Pries-Heje 2004; Baskerville et al. 2003). In our current study, schedule pressure also emerged as a reason for the inadequate conduct of tests.

Second, a trajectory that represents the internal interplay between the provider and the developer was found. From the provider’s side, there is economic pressure to follow exactly what is stipulated in the contract and to avoid using too many resources for the client (thesis). Meanwhile, individual professionals may want to produce high-quality results (antithesis). If this barrier—the pressure confronted by the developer to not waste resources by producing low-quality results—is not overcome, a gray area of practice (synthesis) emerges, which results in producing defective IS for the client.

To summarise the contribution of our study, we developed a new theoretical model that explains why developers produce defective IS/SW for clients. Our model shows the critical stages of ISSD for the development of ‘bad practices’ that eventually lead to deficient IS/SW. Our model shows the underlying reasons and the forces and the counterforces that affect the emergence of this gray area of practice.

5.1 Limitations of the Study

Our research was subject to typical limitations in qualitative interview studies. The number of interviewees may be seen as relatively small (first round, n = 20; second round, n = 9), but this number is quite normal in qualitative interviews (Myers & Newman 2007). Furthermore, the use of the interview method also has potential limitations (i.e., rationalisation, lack of awareness, and fear of one’s identity being revealed) (Fielding 1993), which can be significant due to the questions regarding questionable and dubious practices. To minimise these effects, we did not ask the interviewees to describe their own practices, but instead, we phrased the interview questions in such a way that the respondents were asked to consider general practices in the field (i.e., practices that they have witnessed). Finally, as in any qualitative study, caution should be exercised in generalising our findings to all ISSD settings. Future research is needed to examine if our findings hold true in different ISSD contexts and cultures.

5.2 Conclusions

Quality problems in IS/SW are common technological problems. A number of different development methods and mature models have been proposed to address these challenges. Recent research on agile and lean SW development methods suggests that companies speed up IS/SW development by omitting necessary quality procedures. While agile methods are commonly used in SW development, however, we found no research that specifically explores why deficient IS/SW emerge. We conducted two rounds of interviews with IS/SW developers (n = 20 and n = 9) in order to inductively develop a theoretical model that explains this issue. From our interviews, we identified two processes that lead to deficient IS/SW and the factors that affect this deficiency. In both processes, economic forces developers to omit important quality controls in different stages of IS/SW development. The theoretical model we developed falls under the process theory paradigm of dialectical process theories. From our findings, we propose that the evolution of a gray area of practice should be examined further. According to dialectical theory, the synthesis will become a new thesis (van de Ven & Poole 1995). Therefore, the synthesis we determined in our study, which is producing defective IS for the client, likely confronts the antithesis, and a new synthesis is formed. This evolution of a gray area of practice should be studied further in order to understand if it evolves in practice and how it shapes the IT business. We also propose that the education of future IT professionals should consider this ISSD gray area. Our results revealed that IT professionals confront barriers in requirements definition and testing, so they need to know how they can overcome such barriers and hence avoid gray areas of practice.
References


