Assessing Scrum-based Software Development Process Measurement from COBIT Perspective

VILJAN MAHNIC, NATASA ZABKAR
Faculty of Computer and Information Science
University of Ljubljana
Trzaska 25, SI-1000 Ljubljana
SLOVENIA

Abstract: - In this paper we present the results of the assessment of Scrum-based software development process measurement using COBIT criteria. At first we present our proposal for the model of the Scrum performance measurement. Then we give a short description of COBIT - a model for IT governance, which is commonly used in the IS auditing environment. After that we present the indicators for selected COBIT processes for system development life cycle and use them as criteria for our assessment. This is followed by the analysis of the assessment results. In the end we propose some adjustments to our model.

Key-Words: - COBIT, system development life cycle, agile software development, Scrum, indicators

1 Introduction
Many surveys have identified that the lack of transparency of IT’s cost, value and risks is one of the most important drivers for IT governance. According to [6], transparency is primarily achieved through performance measurement. This also applies to the transparency of software development process, which is closely related to its performance measurement.

Agile software development methods follow the principle of “maximizing the amount of work that need not be done” and therefore abandon many practices prescribed by software quality models including the need for comprehensive metrics plans. However, many researchers and practitioners have recognized that agile methods need more elaborate metrics [1, 4, 13, 14], and a meta-model for modeling and measuring Scrum development process has been presented in [3].

In this paper we shall focus on one agile software development method – Scrum [12]. Our previous research in this area [9, 10] includes the development of a metrics plan for monitoring and improving the performance of a Scrum-based software development process as well as a demonstration of how CMMI requirements for Measurement and Analysis (MA) Process Area [2] can be used in designing a corresponding measurement repository.

The aim of this paper is to assess the proposed measurement model from IT governance perspective. For this purpose we use criteria that are generally accepted in the information systems auditing community and are commonly used for IT governance implementation and assessment - COBIT (Control Objectives for Information and Related Technology) [6].

At first, we give a short description of our previous research in the Scrum performance measurement area. Then we describe COBIT model and present indicators for selected COBIT processes for system development life cycle. After this, we use them as our criteria for the assessment of the Scrum-based software development process. Finally, we discuss the results of the assessment and propose some adjustments to our model.

2 Previous research
In this paper we shall assume that the reader is already familiar with Scrum-based software development process. Detailed descriptions of Scrum-based software development process can be found in [12].

Within Scrum originally only one software development metric is used: the estimate of the amount of work remaining that needs to be done in order to complete a Product Backlog item or a task in a Sprint Backlog. Using this measure, burndown charts can be developed showing work remaining over time and daily/monthly progress.

In our previous research [9, 10], we have assumed that the best performance is achieved when the goals of all stakeholders are satisfied as proposed by Kueng in [8]. We have identified four stakeholders for Scrum process: IT management, Team members, ScrumMaster and Customers. The first stakeholder is IT management and is mainly concerned with traditional aspects of software development performance considering time, cost, and
quality. The second stakeholder is Team members and its main goal is “Job satisfaction”. The ScrumMaster is the third stakeholder and its main goal is “Efficient impediments resolution”. Finally, the main goal regarding the customers, the fourth stakeholder, is “Customer Satisfaction”.

For the purpose of this paper we shall name our model “AGIT” (AGIle software development). The AGIT performance indicators for each previously stated goal are shown in Table 1. Detailed description and formulas of these 12 indicators can be found in [9].

<table>
<thead>
<tr>
<th>Stakeholder</th>
<th>Goal</th>
<th>AGIT Performance Indicator</th>
</tr>
</thead>
<tbody>
<tr>
<td>IT management</td>
<td>Timely Information on Project</td>
<td>Work Effectiveness (ratio between the work spent and the decrement of work remaining)</td>
</tr>
<tr>
<td></td>
<td>Performance</td>
<td>Schedule Performance Index (SPI) (ratio between the earned value (i.e., the value of all tasks completed) and the planned value (i.e., the initial estimate of value of all tasks to be completed till a certain point within the project))</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cost Performance Index of Labor Costs (CPI) (ratio between the earned value (measured in units of currency) and actual costs)</td>
</tr>
<tr>
<td></td>
<td>Quality Improvement</td>
<td>Error Density (number of errors per KLOC (kilo-lines of code))</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Costs of Rework (product of hours spent on rework and cost of an engineering hour)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fulfillment of Scope (ratio between the number of tasks completed in the Sprint and total number of tasks in the Sprint Backlog or between the number of PBIs completed in the release and total number of PBIs committed)</td>
</tr>
<tr>
<td>Team members</td>
<td>Job Satisfaction</td>
<td>The Average Amount of Overtime at Sprint/Release/Project level (the expected hours, the amount of work spent and administrative days)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The Average Number of Projects the Employees Work in Parallel</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Qualitative Evaluation of Working Conditions (communication and teamwork, physical discomfort, psychological well-being, workload, supervision, opportunities for growth, etc.)</td>
</tr>
<tr>
<td>ScrumMaster</td>
<td>Efficient Impediments Resolution</td>
<td>Average Number of Impediments per Task/Sprint/Team</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Mean Time for Resolving an Impediment (at Task/Sprint/Team level)</td>
</tr>
<tr>
<td>Customers</td>
<td>Customer Satisfaction</td>
<td>Qualitative Evaluation of Customer Satisfaction using Criteria (Error Density, Fulfillment of Scope) (the quality of product, price adequacy, reliability in terms of time and costs, completeness of product delivered at the end of each Sprint or release, flexible handling of changes in requirements, good collaboration with the development team, adequate training and documentation, etc.)</td>
</tr>
</tbody>
</table>

3 COBIT Model

3.1 COBIT Model Description

COBIT (Control Objectives for Information and Related Technology) [6] represents a collection of documents which can be classified as generally accepted best practice for IT governance, control and assurance. According to Information Systems Audit and Control Association (ISACA) standards, the information systems auditors are obliged to use COBIT, the generally accepted internal control framework for IT. This was the main reason why we have decided to use COBIT criteria for assessing Scrum performance measurement. Our hypothesis was that by satisfying information systems auditing criteria we can demonstrate that Scrum indicators proposed by AGIT are (or are not) compliant with good practice.

The IT processes are usually ordered into the responsibility domains of plan, build, run and monitor. Within the COBIT framework, these domains are called:

- **Plan and Organise (PO):** Provides direction to solution delivery (AI) and service delivery (DS);
- **Acquire and Implement (AI):** Provides the solutions and passes them to be turned into services;
- **Deliver and Support (DS):** Receives the solutions and makes them usable for end users;
- **Monitor and Evaluate (ME):** Monitors all processes to ensure that the direction provided is followed.
Across these four domains, COBIT has identified 34 IT processes.

For each of its 34 processes COBIT defines goals and metrics to define and measure their outcome and performance based on the principles of Robert Kaplan and David Norton’s balanced business scorecard [7]. COBIT metrics have been developed with the following characteristics in mind:

- A high insight-to-effort ratio;
- Comparable internally an externally;
- Better to have a few good metrics;
- Easy to measure.

These characteristics are mainly compliant with the agility principles that value individuals and interactions over processes and tools, working software over vulnerability, and David Norton’s balanced business scorecard [7].

3.2 COBIT Performance Indicators for Software Development Process

In order to assess Scrum-based development process, we needed to select those COBIT processes that relate to software development. There were few possible options [5, 15] and we have decided to use the slightly modified selection presented in the ISACA (Information Systems Audit and Control Association) auditing guideline for System Development Life Cycle (SDLC) Reviews (Guideline G23) [5].

According to Guideline G23, the system development life cycle is the process, involving multiple stages (from establishing the feasibility to carrying out post implementation reviews), used to convert a management need into an application system, which is custom-developed or purchased or is a combination of both. Since this definition also applies to the Scrum-based software development process, we have decided to use the G23 selection of COBIT processes for our criteria for the assessment. We have slightly broadened this selection by adding two COBIT processes that we considered to be relevant for the purpose of this paper: PO7: “Manage Human Resources” and DS10: “Manage Problems”.

An IS auditor should evaluate the extent to which selected COBIT control objectives are met and the effectiveness of the mechanisms and procedures employed to achieve such objectives. The guideline G23 states that COBIT guidance for the following 5 processes should be considered relevant when performing the audit of the system development life cycle.

Domain Plan and Organise (PO):
- Process PO8: “Manage Quality” is focused on ongoing performance monitoring against predefined objectives.
- Process PO10: “Manage Projects” is focused on monitoring of project risks and progress.

Domain Acquire and Implement (AI):
- Process AI1: “Identify Automated Solutions” is focused on identifying technically feasible and cost-effective solutions.
- Process AI2: “Acquire and Maintain Application Software” is focused on ensuring that there is a timely and cost-effective development process.

Domain Deliver and Support (DS):
- Process DS5: “Ensure Systems Security” is focused on defining IT security policies, plans and procedures, and monitoring, detecting, reporting and resolving security vulnerabilities and incidents.

We have added the following two processes, that we have considered to be relevant for agile development:
- Process PO7: “Manage Human Resources” is focused on hiring and training personnel, motivating through clear career paths, assigning roles that correspond with skills, establishing a defined review process, creating position descriptions and ensuring awareness of dependency on individuals.
- Process DS10: “Manage Problems” is focused on recording, tracking and resolving operational problems; investigating the root cause of all significant problems; and defining solutions for identified operations problems.

The 20 indicators for these 7 COBIT processes are presented in Table 2.

<table>
<thead>
<tr>
<th>Process</th>
<th>Domain</th>
<th>Indicators</th>
</tr>
</thead>
<tbody>
<tr>
<td>PO7: Manage Human Resources</td>
<td>PO7-1: Level of stakeholders’ satisfaction with IT personnel expertise and skills</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PO7-2: IT personnel turnover</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PO7-3: Percent of IT personnel certified according to job needs</td>
<td></td>
</tr>
<tr>
<td>PO8: Manage Quality</td>
<td>PO8-1: Percent of stakeholders satisfied with IT quality (weighted by importance)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PO8-2: Percent of IT processes that are formally reviewed by QA on a periodic basis and that meet target quality goals and objectives</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PO8-3: Percent of processes receiving QA review</td>
<td></td>
</tr>
<tr>
<td>PO10: Manage Projects</td>
<td>PO10-1: Percent of projects meeting stakeholders expectations (on time, on budget and meeting requirements—weighted by importance)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PO10-2: Percent of projects receiving post-implementation reviews</td>
<td></td>
</tr>
</tbody>
</table>
4 Results of the Assessment

Our aim was to assess the level of compliance of AGIT model to information technology governance good practice requirements as represented in COBIT. Therefore we have compared COBIT and AGIT performance indicators. The results are presented as follows.

4.1 Domain Plan and Organise (PO)

PO7-1: Level of stakeholders’ satisfaction with IT personnel expertise and skills
Status: Compliant
AGIT indicator: “Qualitative Evaluation of Customer Satisfaction using Criteria” is compliant with this COBIT indicator.

PO7-2: IT personnel turnover
Status: Partly compliant
This COBIT indicator is partly covered by the three AGIT indicators that relate to stakeholder “Team members “:
- The Average Amount of Overtime at Sprint/Release/Project level;
- The Average Number of Projects the Employees Work in Parallel;
- Qualitative Evaluation of Working Conditions.

PO7-3: Percent of IT personnel certified according to job needs
Status: Non-compliant
There is no AGIT indicator that would be compliant with this COBIT indicator.

PO8-1: Percent of stakeholders satisfied with IT quality (weighted by importance)
Status: Compliant
This COBIT indicator can be mapped to AGIT indicator “Qualitative Evaluation of Customer Satisfaction using Criteria”.

Apart from this, at the end of the Sprint, every stakeholder can assess product quality at a Sprint review meeting at which the Team presents what was developed during the Sprint to the Product Owner and any other stakeholders who want to attend.

PO8-2: Percent of IT processes that are formally reviewed by QA on a periodic basis and that meet target quality goals and objectives,

PO8-3: Percent of processes receiving QA review
Status: Compliant
These two COBIT indicators are covered by the Sprint retrospective meeting.

PO10-1: Percent of projects meeting stakeholders expectations (on time, on budget and meeting requirements—weighted by importance)
Status: Compliant
This COBIT indicator can be mapped to the following three AGIT indicators:
- Work Effectiveness;
- Schedule Performance Index (SPI);
- Cost Performance Index of Labor Costs (CPI).

PO10-2: Percent of projects receiving post-implementation reviews
Status: Non-compliant
There is no AGIT indicator that would be compliant with this COBIT indicator.

PO10-3: Percent of projects following project management standards and practices
Status: Compliant
This COBIT indicator is covered by the ScrumMaster role, under the assumption that Scrum is used as a project management standard.
4.2 Domain Acquire and Implement (AI)

AI1-1: Number of projects where stated benefits were not achieved due to incorrect feasibility assumptions
Status: Partly compliant
This COBIT indicator is partly covered by the AGIT indicator “Fulfillment of Scope”, under the assumption that the reasons for non-completion of the Sprint tasks are related to incorrect feasibility assumptions.

AI1-2: Percent of feasibility studies signed off on by the business process owner
Status: Compliant
This COBIT indicator is covered by the Sprint planning meeting.

AI1-3, AI2-2: Percent of users satisfied with functionality delivered
Status: Compliant
These two COBIT indicators as well as previously described COBIT indicator PO8-1 (Percent of stakeholders satisfied with IT quality) can be mapped to AGIT indicator “Qualitative Evaluation of Customer Satisfaction using Criteria”.

AI2-1: Number of production problems per application causing visible downtime
Status: Compliant
This COBIT indicator is partly covered by the following two AGIT indicators:
- Error Density;
- Costs of Rework.
AGIT uses the number of errors reported by the user in a fixed period after release as well as classification of tasks in the Sprint Backlog according to the type of work performed (development, testing, rework due to the change in requirements, rework due to error reported by the customer, etc.).

4.3 Domain Deliver and Support (DS)

DS5-1: Number of incidents damaging the organisation’s reputation with the public, DS5-2: Number of systems where security requirements are not met, DS5-3: Number of violations in segregation of duties
Status: Compliant
AGIT indicator: “Qualitative Evaluation of Customer Satisfaction using Criteria” is compliant with this COBIT indicator, under the condition that questionnaire includes questions regarding security requirements that are result of development activities.

DS10-1: Number of recurring problems with an impact on the business
Status: Compliant
AGIT indicator: “Qualitative Evaluation of Customer Satisfaction using Criteria” is compliant with this COBIT indicator.

DS10-2: Percent of problems resolved within the required time period
Status: Compliant
This COBIT indicator is covered by the following two AGIT indicators:
- Average Number of Impediments per Task/Sprint/Team;
- Mean Time for Resolving an Impediment (at Task/Sprint/Team level).

DS10-3: Frequency of reports or updates to an ongoing problem, based on the problem severity
Status: Compliant
This COBIT indicator is covered by the ScrumMaster role.

4.4 Final Results
The results show that COBIT addresses higher level of software development process than AGIT, therefore the comparison of these indicators is not always possible. We can see that AGIT indicators can be compliant with COBIT indicators, partly compliant or non-compliant.

There are 16 (80%) COBIT indicators that are covered by the AGIT indicators (PO7-1, PO8-1, PO8-2, PO8-3, PO10-1, PO10-3, AI1-2, AI1-3, AI2-1, AI2-2, DS5-1, DS5-2, DS5-3, DS10-1, DS10-2, DS10-3).

There are 2 (10%) COBIT indicators that can be partly mapped to AGIT indicators (PO7-2, AI1-1).

There are 2 (10%) COBIT indicators that are not included in the AGIT model (PO7-3, PO10-2).

Our conclusion is that model AGIT almost completely satisfies COBIT criteria. Non-compliances are the result of the different scopes of the models compared. While AGIT model is focused on the operational level of software development, COBIT model addresses development life cycle on a higher level and from different perspectives. Non-compliant COBIT indicators do not depend on the software development method (in our case Scrum), but are related to the human resources strategy (percent of IT personnel certified to job needs) and project management strategy on the organizational level (post-implementation reviews). Therefore these non-compliances cannot be considered as major weaknesses of AGIT model. The same applies to the partly compliant indicators.

We can introduce non-compliant and partly compliant indicators to AGIT model in the following way:
• “Percent of IT personnel certified to job needs” can be calculated by keeping records of the team members’ certificates;
• “Percent of projects receiving post-implementation reviews” can be calculated by keeping records about post-implementation reviews for each project;
• “IT personnel turnover” can be calculated by keeping records not only on the number of all developers, as recommended in AGIT model, but also on the number of the developers that left the company and the number of the new developers (in the certain period of time);
• “Number of projects where stated benefits were not achieved due to incorrect feasibility assumptions” can be calculated by introducing classification of the causes of the non-completion of the tasks for the indicator “fulfillment of scope” (one of the possible causes could be incorrect feasibility assumptions).

5 Conclusion
The aim of this paper was to assess the level of compliance of Scrum-based software development process measurement with COBIT requirements.

We have used COBIT criteria in order to determine compliance of AGIT model with the information systems auditing criteria for system development life cycle, as defined in the information systems auditing guideline G23.

Therefore we have used slightly modified selection of COBIT processes defined in this guideline. We have compared 12 indicators from our AGIT model to 20 indicators recommended by COBIT. Our conclusion is that model AGIT almost completely satisfies COBIT criteria. Non-compliant or partly compliant indicators do not depend on the software development method and can be easily introduced to AGIT model.

We hope that the results of this comparison will contribute to the credibility of AGIT model. Our future research activities will be focused on the mapping of AGIT model to other comparable models.

References: