Identifying Architectural Viewpoints
– A Scenario-Based Approach –

Andreas Fuhr
University of Koblenz-Landau
afuhr@uni-koblenz.de

Abstract

For reverse-engineering software architectures, architectural viewpoints need to be defined in a way where all customer needs are considered. This paper presents a systematic, scenario-based approach for identifying customer-specific viewpoints. The approach was successfully applied in the Cobus II project to identify 11 customer-specific viewpoints for describing the architecture of a COBOL system.

1 Introduction

Recovering the architecture of legacy systems is one key goal in reverse engineering projects. Before applying appropriate techniques to elevate legacy artifacts to a more abstract architectural representation, it is to be defined how the *architectural description (AD)* of the system's architecture is represented.

Like in architectural descriptions of buildings, software architectures are presented in different *views*, presenting certain information for a defined purpose. The structure of views is specified in *viewpoints*.

Several general-purpose sets of viewpoints exist, e.g., the Siemens views [HNS06] or the 4+1 views [Kru95]. Those provide a generic set of viewpoints that have proven to be useful in many architectural descriptions. However, customer-specific views on their legacy systems are often necessary to satisfy customer needs. Identifying which new viewpoints are required and what information to provide may be hard; customers often do not know what architectural perspectives they need to solve their problems.

Facing this challenge in the Cobus II project, a scenario-based approach for identifying useful customer-specific viewpoints was developed (*Scenario-Based Viewpoint Identification, ScnB-VI*). The ScnB-VI approach uses hypothetical scenarios to identify the use-cases of the to-be-created architectural description and to derive concerns, questions and finally viewpoints from them. The ScnB-VI approach was successfully applied in the Cobus II project to identify 11 customer-specific viewpoints.

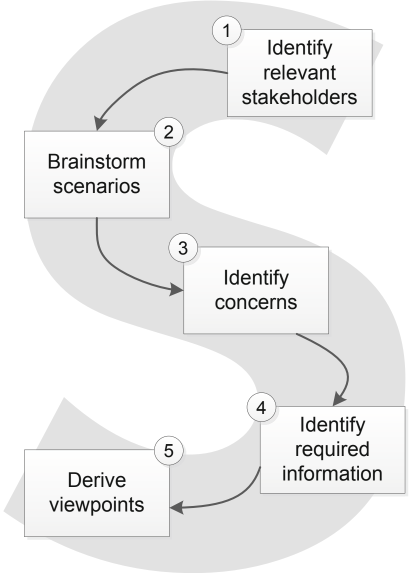

Figure 1: Scneario-Based Viewpoint Identification process

2 ScnB-VI: Scenario-Based Viewpoint Identification

Similar to the Goal Question Metric (GQM) approach [BCR02] for identifying metrics, the ScnB-VI approach starts bottom-up; here, identifying viewpoints starts at the future use of the to-be-constructed architectural description and goes back to the viewpoints required for it.

Figure 1 shows the process of the ScnB-VI approach, containing five phases which are described in the following paragraphs.

Identify relevant stakeholders. First, all *stakeholders* that will use the to-be-developed architectural description (AD) in future, are identified. Amongst others, relevant stakeholders are managers, product designers, software architects, lead developers, maintenance staff, testers or auditors.

Brainstorm scenarios. Together with the stakeholders, *hypothetical scenarios* about how the AD may

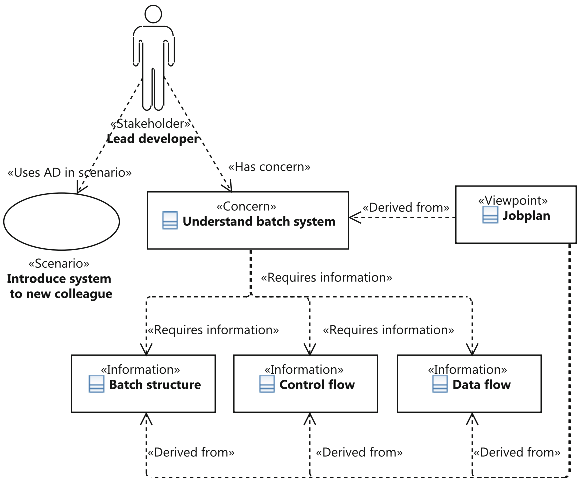

Figure 2: Information pieces gathered when identifying the Jobplan viewpoint

be used in future are brainstormed. Scenarios enclose typical architecture use-cases (e.g., system (re-) design, system understanding, communication about the system or impact analyses) as well as customer-specific use-cases.

Identify concerns. For each scenario, the main *concerns* it encloses are identified. Concerns cover questions a stakeholder wants to be answered by the AD in a scenario. In addition, concerns cover tasks a stakeholder wants to accomplish and for which he needs the AD.

Identify required information. Relating to the concerns, all pieces of information are identified that are necessary for solving the concerns. Information may be discoverable without any sophisticated analysis (e.g., a stakeholder needs the name of a subsystem for solving his concern) or further analyses are required to extract the information from the legacy (e.g., a stakeholder needs the data flow between subsystems to solve his concern).

Derive viewpoints. Finally, customer-specific viewpoints are derived from the concerns as well as what information are required. For each concern, at least one viewpoint should be specified in order to cover all stakeholder needs. However, one viewpoint may cover more than one concern. The required information provides details about the structure of the viewpoints and how to construct concrete views for it (i.e., how to extract information from the legacy).

3 Case study: Cobus II project

The Cobus II project (COBOL Bestandsanalyse und Sanierung II) is a cooperation project between the University of Koblenz-Landau and the Debeka Versicherungsgruppe, aiming at analyzing the as-is state of Debeka's COBOL system. During an "architecture discovery" workpackage, the ScnB-VI approach was applied to identify viewpoints for describing the architecture of the COBOL system.

Together with five stakeholders (1 manager, 1 architect, 2 maintenance staff members, 1 lead developer), four key scenarios were identified. Thinking the scenarios through lead to 25 concerns that were raised by the stakeholders. 15 key pieces of information were identified as necessary for solving the concerns. Based on those findings, 11 customer-specific viewpoints were defined to cover all concerns.

In the following, the identification of one viewpoint is described exemplarily (Figure 2).

Among the key stakeholders using the AD at Debeka were *lead developers*. Lead developers at Debeka design components and supervise their implementation and maintenance. I.e., they cover architectural tasks within their division (*Identify relevant stakeholders*).

Brainstorming with a lead developer, she described a scenario of introducing their subsystem to new colleagues (*Brainstorm scenarios*).

One key concern here is to understand the structure and behavior of the batch system. Within a batch system, *jobplans* define activities to work off a business process. Within a jobplan, *scripts* call COBOL programs and read and write files (*Identify concerns*).

The required information for the "batch system understanding" concern were the structure of the jobplans and their scripts, as well as the control flow and data flow across scripts and across jobplans (*Identify required information*).

Based on these findings, a jobplan viewpoint was specified.

4 Conclusion

This paper introduced the ScnB-VI approach for systematic, scenario-based identification of architectural viewpoints and presented a first, successful application within the Cobus II project to identify 11 customer-specific viewpoints.

References

[BCR02] Victor Robert Basili, Gianluigi Caldiera, and Hans Dieter Rombach. Goal Question Metric (GQM) Approach. In J. J. Marciniak, editor, *Encyclopedia of Software Engineering*. John Wiley, New York, 2002.

[HNS06] Christine Hofmeister, Robert Nord, and Dilip Soni. *Applied software architecture*. The Addison-Wesley object technology series. Addison-Wesley, Reading, 2006.

[Kru95] Philippe B. Kruchten. The 4+1 View Model of architecture. *IEEE Software*, 12(6):42–50, 1995.