

Chapter XX

Formal Analysis of Workflows in Software Development

Harry S. Delugach

University of Alabama in Huntsville, USA

ABSTRACT

Automated tools are often used to support software development workflows. Many of these tools are aimed toward a development workflow that relies implicitly on particular supported roles and activities. Developers may already understand how a tool operates; however, developers do not always understand or adhere to a development process supported (or implied) by the tools, nor adhere to prescribed processes when they are explicit. This chapter is aimed at helping both developers and their managers understand and manage workflows by describing a preliminary formal model of roles and activities in software development. Using this purely descriptive model as a starting point, the authors evaluate some existing tools with respect to their description of roles in their processes, and finally show one application where process modeling was helpful to managers. We also introduce an extended model of problem status as an example of how formal models can enrich understanding of the software development process, based on the analysis of process roles.

People sometimes make errors. The problem here was not the error, it was the failure of NASA's systems engineering, and the checks and balances in our processes to detect the error. That's why we lost the spacecraft.

*Edward Weiler,
NASA's Associate Administrator on the loss
of the \$327 million Mars Climate Orbiter.*

INTRODUCTION

Many automated tools are available to support software development. There are two main reasons for an organization to use these tools:

- Much of software development takes place in distributed environments, or at least where the participants have difficulty meeting regularly face-to-face. Automated (often web-based) tools allow them to collaborate in a generally cost-effective way compared to travel and shipping costs.
- Software development workflows prescribe various *activities* to be tracked and *artifacts* to be created and maintained. Even when developers are able to collaborate in person, the number of these can become large and therefore requires organizing tools and a central repository.

As with all tools, their effectiveness is determined by how well participants understand how to use them. There is ample evidence that mere use of tools is not sufficient to support an effective workflow. Even if developers understand a tool's basic operation, they often do not understand or adhere to any development process supported (or implied) by the tool. This chapter examines some popular web-based software engineering tools from a pragmatic role-oriented perspective. That is, we intend to focus on the roles and purposes within the context of the development process, rather than characteristics of artifacts or products.

Our ultimate goals in developing these models is the following:

1. To better describe and analyze the processes themselves.
2. To formally analyze and evaluate tools with respect to generally accepted process models, and
3. To formally compare and contrast the models with each other.
4. To provide formal definitions based on process models.

The approach in this chapter illustrates all four of these goals. First we motivate the general value of formal models in analyzing process, and then provide some background on workflow modeling with respect to the software development process. The main body of the chapter applies this approach to one particular sub-process (namely bug tracking). Each of the four goals is discussed in turn, using examples to illustrate the approach.

This work continues in the spirit of previous work in modeling development processes (Delugach, 2007) (de Moor & Delugach, 2006) and in using conceptual graphs for modeling communication (Delugach, 2006) and software development (Delugach, 1996) (Delugach, 1992). In this chapter, we use conceptual graphs—a well-known knowledge representation—as a clear and effective way of formally representing the parts of a workflow. In future work, some automated analysis may use conceptual graphs' formal basis in logical reasoning and inference; however, this chapter does not exploit those capabilities for these illustrations.

THE VALUE OF FORMAL MODELING

At this point, it is useful to evaluate the role of formal modeling in software system development. While nearly all developers acknowledge the value of formally modeling the software system itself, there is less agreement on the role of formal modeling of the *process* of software development. Resistance to this idea is usually caused by “horror stories” of:

- Incorrect or incomplete models of a process, which initially give the impression of soundness but then later reveal themselves to be inappropriate
- Models that have been imposed on a development team by either upper management without proper evaluation, or by contractual obligations that are included as “boiler plate” requirements without any evaluation as their appropriateness

16 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/formal-analysis-workflows-software-development/21414

Related Content

The Dynamics of Social Media Marketing Content and Customer Retention

Michelle Willis (2022). *Research Anthology on Social Media Advertising and Building Consumer Relationships* (pp. 217-237).

www.irma-international.org/chapter/the-dynamics-of-social-media-marketing-content-and-customer-retention/305337

A Study on Consumer Intention in a Web Portal Game from the Perspective of Flow Experience

Yi-Fen Chen, Chia-Wen Tsai, Pei-Chun Liao and Wen-Yu Chen (2012). *International Journal of Virtual Communities and Social Networking* (pp. 1-18).

www.irma-international.org/article/study-consumer-intention-web-portal/73003

Constructing and Evaluating Social Software: Lessons from Interaction Design

Christopher Douce (2011). *Social Computing Theory and Practice: Interdisciplinary Approaches* (pp. 197-214).

www.irma-international.org/chapter/constructing-evaluating-social-software/48856

The Impact of Social Marketing on SMEs in India: A Theoretical Perspective

Amandeep Singh and Amrinder Singh (2023). *Strengthening SME Performance Through Social Media Adoption and Usage* (pp. 213-220).

www.irma-international.org/chapter/the-impact-of-social-marketing-on-smes-in-india/316229

Computational Trust in SocialWeb: Concepts, Elements, and Implications

Kiyana Zolfaghar and Abdollah Aghaie (2010). *International Journal of Virtual Communities and Social Networking* (pp. 60-74).

www.irma-international.org/article/computational-trust-socialweb/45779