



Doctoral Thesis

## **Project GIPSY facing the challenge of future integrated software engineering environments**

**Author(s):**

Murer, Tobias

**Publication Date:**

1997

**Permanent Link:**

<https://doi.org/10.3929/ethz-a-001856515> →

**Rights / License:**

[In Copyright - Non-Commercial Use Permitted](#) →

This page was generated automatically upon download from the [ETH Zurich Research Collection](#). For more information please consult the [Terms of use](#).

Miss. ETH ex. B

Diss. ETH No 12350

# **Project GIPSY**

## **Facing the Challenge of Future Integrated Software Engineering Environments**

A dissertation submitted to the  
SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZÜRICH

for the degree of  
DOCTOR OF TECHNICAL SCIENCE

presented by

Tobias Murer  
Dipl. Inf.-Ing ETH  
Dipl. Betr. Wiss. NDS ETH  
ETH Zürich  
born April 10, 1966  
citizen of Beckenried, Nidwalden



CatE

accepted on the recommendation of  
Prof. Dr. Albert Kündig, examiner  
Dr. Bruno Schäffer, co-examiner

1997

## **ABSTRACT**

The goal of a software process is the development and maintenance of a software product. A software engineering environment (SEE) integrates a combination of tools to a cohesive environment to support the software process. Emerging technologies such as software components and the Internet will change the way software is produced and marketed. Component suppliers need to co-operate tightly within virtual organizations and share their software processes to produce properly interoperating components. The support of such a virtual organization that is managed decentralized and involves various cultures and methods challenges the social, technical and organizational aspects of traditional SEE concepts.

The goal of the GIPSY (Generating Integrated Process support SYstems) project was the development of a SEE framework prototype capable to support a virtual organization's software process aimed to produce high quality software. The prototype allows for the generation of a SEE based on process and tool specifications. This thesis outlines the conceptual work of the project. I distinguish two layers of abstraction with different integration requirements: The organization layer focuses on the coarse-grain process steps providing process definition and enactment services reflecting organizational integrity. The artifact layer copes with the fine-grain process steps enabling the specification and generation of tools that are used to work on artifacts and that integrate on a high data integration level.

The aim of the organization layer is to enable cooperation and communication based on organizational integrity. Organizational integrity means the common understanding of the structure and evolution of product, process and organization form shared among all process participants including users and tools. I propose to achieve organizational integrity by using a simple similar model to represent product, process

and organization form. I accept the trade-offs of applying a simple similar model to represent the three main organizational aspects that is less powerful compared to aspect-specific models. Gaining organizational integrity by sacrificing sophisticated models is a promising approach to provide a common understanding of organization within a heterogeneous virtual organization. Reflecting organizational integrity is an essential design issue for future SEEs.

The artifact layer focuses on the specification and generation of highly integrated tool components supporting formal languages. Compiler-compiler technology is extended by the object-oriented paradigm to a tool specification technique providing powerful decomposition, extension and import mechanisms. This allows for a fast and flexible transformation of formal software engineering methods into supporting tools. The goal of this layer is to achieve data integration on a high level.

## ZUSAMMENFASSUNG

Der Zweck eines Softwareprozesses ist die Entwicklung und der Unterhalt von Software. Softwareentwicklungssysteme (SWE-Systeme) integrieren eine Kombination von Werkzeugen, um den Softwareprozess zu unterstützen. Aufstrebende Technologien wie Softwarekomponenten oder das Internet werden die Produktion und Vermarktung von Software verändern. Anbieter von Komponenten kooperieren in virtuellen Organisationen und verbinden ihre Prozesse, um gemeinsam zusammenpassende Komponenten zu produzieren. Die Unterstützung solcher virtuellen Organisationen, welche dezentralisiert verwaltet werden und verschiedene Kulturen und Methoden vereinen, fordern die sozialen, technischen und organisatorischen Aspekte der traditionellen Softwareentwicklung und der entsprechenden SWE-Systemkonzepte heraus.

Das Ziel des GIPSY (Generating Integrated Process support SYstems) Projektes war die Entwicklung eines Prototypen für eine SWE-Systemarchitektur, welche die Softwareprozesse innerhalb einer virtuellen Organisation unterstützen kann. Der Prototyp ermöglicht die Erzeugung von SWE-Systemen basierend auf Spezifikationen von Prozessen und Werkzeugen. Die vorliegende Dissertation gibt einen Überblick über die im Projekt erarbeiteten Konzepte. Ich unterscheide dabei zwei Abstraktionsebenen, welche verschiedene Integrationsanforderungen genügen müssen: Die Organisationsebene behandelt die grossen Prozessschritte, indem sie Dienste für die Definition und Durchführung von Prozessen anbietet und organisatorische Integrität vermittelt. Die Dokumentenebene konzentriert sich auf die kleinen Prozessschritte und bietet Dienste für die Spezifikation und Erzeugung von Werkzeugen an, welche zur Bearbeitung von Dokumenten verwendet werden und eine hohe Datenintegration aufweisen.

Das Ziel der Organisationsebene ist, basierend auf organisatorischer Integrität, Kooperation und Kommunikation zu ermöglichen. Organisato-

rische Integrität beschreibt das gemeinsame Verständnis der Struktur und Entwicklung von Produkt, Prozess und Organisationsform, welches sich sowohl Werkzeuge wie auch Benutzer als Prozessbeteiligte teilen. Ich schlage ein einfaches, ähnliches Modell zur Beschreibung von Produkt, Prozess und Organisationsform vor, um organisatorische Integrität zu erreichen. Ich akzeptiere dabei die Nachteile eines einfachen, ähnlichen Modells für die drei organisatorischen Hauptaspekte, welches weniger mächtig ist als entsprechende aspektspezifische Modelle. Ich denke, dass das Erreichen von organisatorischer Integrität durch das Opfern von mächtigen Modellen eine interessante Möglichkeit darstellt, um ein gemeinsames Verständnis von Organisation innerhalb einer heterogenen, virtuellen Organisation zu vermitteln. Organisatorische Integrität wird dadurch zu einem äusserst wichtigen Aspekt für den Entwurf von zukünftigen SWE-Systemen.

Die Dokumentenebene konzentriert sich auf das Spezifizieren und Erzeugung von hoch integrierten Werkzeugkomponenten, welche die Arbeit mit formalen Sprachen unterstützen. Compiler-Compiler-Technologie wurde mit dem Paradigma der Objekt-orientierung zu einer Technik erweitert, welche mächtige Dekomposition-, Erweiterungs- und Importmechanismen zur Beschreibung von Werkzeugen anbietet. Dies erlaubt den schnellen und flexiblen Transfer von formalen Methoden der Softwaretechnik in unterstützende Werkzeuge. Das Ziel dieser Ebene ist das Erreichen von Datenintegration auf einem hohen Niveau.