



Doctoral Thesis

GIPSY: Ein Ansatz zum Entwurf integrierter Softwareentwicklungssysteme

Author(s):

Marti, Reto

Publication Date:

1994

Permanent Link:

<https://doi.org/10.3929/ethz-a-000943619> →

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](#).

3. Mai 1994

DISS. ETH Nr. 10463

GIPSY: Ein Ansatz zum Entwurf integrierter Softwareentwicklungssysteme

ABHANDLUNG
zur Erlangung des Titels

DOKTOR DER TECHNISCHEN WISSENSCHAFTEN
der
EIDGENÖSSISCHEN TECHNISCHEN HOCHSCHULE ZÜRICH

vorgelegt von
Reto Marti
Dipl. El.-Ing ETH
ETH Zürich

geboren am 13.6.1961
von
Etziken, Solothurn

Angenommen auf Antrag von:
Prof. Dr. A. Kündig, Referent
Prof. Dr. H. Mössenböck, Korreferent



Abstract

The GIPSY project (**Generator of Integrated Programming Systems**) at the Laboratory of Computer Engineering and Networks concerns the design of a novel architecture for distributed and integrated software development environments (SWD-environments). This architecture consists of a project-invariant (system kernel) and -variant part, e.g. tools or the documents data models. This dissertation examines the concepts of the architecture in relation to the *ECMA reference model*. SWD-environments which are based on this architecture, free users from the necessity of routine tasks, such as managing project data or controlling the software development process.

This dissertation also describes how the project-variant part of the GIPSY-based SWD-environments are constructed. The construction process starts with the formal definition of a software development model. SWD-models define valid development processes through a partial order of activities. They also lay down the properties of the project data (documents), which must hold before and after a particular activity. Thus, activities are interpreted formally as *document transformations*. This means, that at the beginning of an activity, the software engineer receives a set of previously written documents. In order to conclude his activity correctly, he has to produce a set of new documents, which are consistent with the received ones.

Formal language definitions written in GIPSY/L are used to describe the documents' properties. These definitions combine the documents' syntactical structure and semantical properties. From these definitions, GIPSY generates object-oriented data models (attributed syntax trees), which enable tools to operate on common documents. Object-oriented languages introduce the concept of extensibility. The same concept has been built into GIPSY/L. Thus, complex language definitions are decomposed into several smaller definitions. This dissertation demonstrates, that extensibility is an important scheme in which to integrate SWD-tools into one homogeneous working environment.

Finally, the document architecture is presented. The data models which are generated by GIPSY are important components of each document. They describe attributed syntax trees, which define the grammatical structure and semantics of the document

texts. They also define a text-oriented editing model, which is implemented with a novel incremental scanning and parsing scheme.

Kurzfassung

Im GIPSY-Projekt (**G**enerator für **I**ntegrierte **P**rogrammentwicklungs-**S**ysteme) des Instituts für Technische Informatik und Kommunikationsnetze wird eine neue Architektur für verteilte integrierte Softwareentwicklungssysteme (SWE-Systeme) entwickelt. Diese Architektur besteht aus einem projekt-invarianten Systemkern und einem -varianten Teil, z.B. Entwicklungswerkzeuge oder die Datenmodelle der Dokumente. Die vorliegende Arbeit beschreibt anhand des *ECMA-Referenzmodells* die Konzepte des Systemkerns. Die Softwareentwicklungssysteme, die auf der Basis des Systemkerns gebaut werden, sollen die Projektbeteiligten von unnötigen Arbeitsschritten, wie zum Beispiel dem Verwalten der Projektdaten oder der Ablaufkontrolle des Projekts, befreien.

Die vorliegende Arbeit zeigt aber auch den Ansatz auf, wie die projekt-varianten Teile GIPSY-basierter SWE-Systeme entwickelt werden können. Ausgangspunkt dafür ist die Formalisierung der Softwareentwicklungsmodelle. Diese Modelle beschreiben gültige Abläufe eines Projekts durch eine partielle Ordnung von Tätigkeiten und definieren die Eigenschaften der erstellten Projektdaten (Dokumente). Tätigkeiten werden darin als *Dokumententransformationen* verstanden. Zu Beginn einer Tätigkeit werden dem Softwareentwickler eine Menge von bereits erstellten Dokumenten zur Verfügung gestellt, worauf er der Tätigkeit entsprechend neue, zu den erhaltenen konsistente Dokumente anfertigen muss.

Die erforderlichen Eigenschaften dieser Dokumente werden durch formale Sprachdefinitionen in GIPSY/L festgelegt. Diese Definitionen umfassen sowohl die syntaktischen wie auch die semantischen Eigenschaften dieser Dokumente. Aus ihnen erzeugt GIPSY objektorientierte Datenmodelle (attributierte Syntaxbäume), auf welchen die im SWE-System installierten Werkzeuge, z.B. Dokumenteneditoren, gemeinsam arbeiten können. Wie dies von objektorientierten Sprachen her bekannt ist, wurde in GIPSY/L das Prinzip der Erweiterbarkeit eingeführt. Damit ist es möglich, komplexe Sprachdefinitionen in mehrere voneinander abgeleitete Definitionen zu zerlegen. Wie in der Dissertation gezeigt wird, ist die Erweiterbarkeit aber auch ein wichtiges Hilfsmittel, Werkzeuge zu einer interaktiven Arbeitsumgebung integrieren zu können.

Abschliessend wird die Dokumentenarchitektur besprochen. Ein wichtiger Bestandteil davon sind die von GIPSY erzeugten Datenmodelle. Sie beschreiben attributierte Syntaxbäume, welche den grammatikalischen Aufbau und die Semantik der Dokumententexte definieren. Auf diesen Datenmodellen sind textorientierte Editieroperationen definiert. Anhand eines neuen inkrementellen Scan- und Parsealgorithmus wird gezeigt, wie diese Operationen direkt auf den Syntaxbäumen ausgeführt werden.