

Ausgabe 2003 / 2

Erscheinungsart: ca. 4 x jährlich in elektronischer Form

Vorgehensmodelle

weitere in dieser Ausgabe ...

- ⇒ Vorgehensmodelle - Eine Leine für den 'Werwolf' Software?
- ⇒ Das Spiralmodell von Pomberger und Pree
- ⇒ Vorgehensmodelle im Überblick
- ⇒ Interessante Links zum Thema

Kurzdefinition / Glossar ...

- ⇒ Vorgehensmodelle sollen den Arbeitsprozess vereinheitlichen, um eine verlässliche Arbeitsgrundlage zu schaffen. [Informatik-Handbuch]
- ⇒ Vorgehensmodelle sind eine einheitliche und verbindliche Vorgabe für Aktivitäten und deren Ergebnisse. [V-Modell97]

Vorgehensmodelle: 'Kochrezepte' für die Software-Entwicklung ...

Man kann natürlich auch ohne Kochrezepte kochen. Mit entsprechendem Wissen und praktischer Erfahrung wird dies auch meist mehr oder weniger gut gelingen. Das Ergebnis ist jedoch oft nicht vorhersagbar (manchmal gelingt es, manchmal ist es etwas 'versalzen' und teilweise ist es auch ungenießbar). Mit einem Kochrezept, das je nach Bedarf ev. auch bewusst etwas abgewandelt werden kann, wird ein gleichmäßigeres Ergebnis erzielt werden können (auch noch nicht so erfahrene Köche werden damit zumeist Erfolg haben).

Welchen Nutzen bringt es nun, wenn man in der Software-Entwicklung Vorgehensmodelle definiert und anwendet?

- ⇒ **Eindämmung der Kosten über den Lebenszyklus eines Software-Produkts** (reduzierter Ressourceneinsatz - frühes Erkennen von Fehlentwicklungen - geringere Schulungskosten - transparentere Kalkulation - geringeres Kostenrisiko - ...)
- ⇒ **Qualitätsverbesserung** (Standardisierung - einfachere Prüfung - einfachere Wartbarkeit - ...)
- ⇒ **Verbesserung der Kommunikation** (einheitliche Begriffe - standardisierte Zwischenergebnisse - detailliertere Beschreibungen - weniger Missverständnisse - ...)
- ⇒ **Zusatznutzen** (oft schon gute Grundlage für angestrebte Zertifizierungen oder Auditierungen wie z.B. ISO9000, CMM, ...).

Nutzen und Tipps für Software-Käufer / Kunden ...

Auch Software-Käufer bzw. Interessenten können das Wissen über Vorgehensmodelle zum eigenen Nutzen verwenden:

- ⇒ **Indiz für Qualität:** wenn ein Lieferant ein adäquates Vorgehensmodell verwendet, ist dies zumindest schon ein Indiz für Professionalität. (Einen Vergleich gängiger Vorgehensmodelle finden Sie weiter hinten.)
- ⇒ Lassen Sie sich das Vorgehensmodell Ihres Lieferanten vor einer Entscheidung genau erklären und ziehen Sie wenn notwendig einen IT-Spezialisten hinzu. Sie werden einerseits Ihre Lieferanten und auch deren Preisbildung besser verstehen und andererseits sehr schnell die **Spreu vom Lieferanten-Weizen trennen können**.
- ⇒ **Gute Basis für Zusammenarbeit:** der Lieferant und auch Sie als Käufer wissen, wie und in welchen Schritten die Leistungen erbracht werden und wann welches Ergebnis zu erwarten ist.



**Theorie und Praxis
bei Vorgehensmodellen**

"Eine Entscheidung ist besser als keine Entscheidung", "Ein Businessplan ist besser als kein Businessplan", ...

Man könnte noch unzählige Beispiele finden, in denen die grundsätzliche Notwendigkeit nach einer Planung und systematischen Vorgehensweise hervor-gehoben und unterstrichen wird.

Im IT-Bereich ist es nicht anders.

Leider jedoch wird in IT-Unternehmen oder auch IT-Abteilungen oft auf klare Vorgehensmodelle verzichtet.

Es wird dem Einzelnen überlassen, sich mehr oder weniger gut zu organisieren.

Manchmal funktioniert dies auch erstaunlich gut. Zumeist jedoch verursacht eine sehr individuell gesteuerte Vorgehensweise unmittelbar erheblich erhöhte Prozesskosten durch Reibungsverluste, Wissensverluste bei Weggang/Neuzugang von Mitarbeitern, ... ganz zu schweigen von den erhöhten Wartungs-, Support- und Betriebskosten von qualitativ mangelhaften Systemen.

Viele Verantwortliche meinen, sich kurzfristig einen Kostenvorteil verschaffen zu können, wenn sie auf klare Vorgehensweisen und Regelungen verzichten.

Dass dies nicht so ist, zeigt die Praxis und wird auch in vielen Studien belegt, in denen die im IT-Bereich dramatisch hohe Anzahl fehlgeschlagener Projekte aufgezeigt wird.

Dipl.-Ing. Johannes Bergmann

allgemein gerichtlich beideter und
zertifizierter Sachverständiger für Informatik

Der Quality-Newsletter ist ein periodisches Informationsmedium von Software Quality Lab und dessen Partnern mit Schwerpunkt Software-Qualitätsmanagement.

Inhalt: fachliche Beiträge und Schwerpunktthemen, Vorstellung neuer Produkte und Leistungen, neue wissenschaftliche Erkenntnisse, Branchen-News, ...

Aktuelle Fach- und Forschungsbeiträge sind willkommen.

Einsendungen an info@software-quality-lab.at.

Weitere Infos zu diesem und anderen Themen finden Sie auf <http://www.software-quality-lab.at>.

Vorgehensmodelle: Eine Leine für den 'Werwolf' Software?

von Univ.-Prof. Dr. Gerhard Chroust

Abteilung für Systemtechnik und Automation, Institut für Systemwissenschaften, Johannes Kepler Universität Linz

Software hat einige Eigenschaften, die die Sicherstellung ihrer Qualität schwieriger machen als bei vielen anderen Industrieprodukten: Software ist primär ein Gedankenprodukt, ist unsichtbar, sehr komplex, leicht änderbar und soll sich jeder Umgebung leicht anpassen (Fred Brooks, 1968).

Kleinste Veränderungen im Code (z.B. ein fehlerhafter Strichpunkt) **können bereits schwerwiegendste Änderungen des Verhaltens hervorrufen** (wie zum Beispiel den Absturz der Ariane-Rakete).

Fred Brooks sprach deshalb auch vom 'Werwolf Software', den man mit keiner noch so magischen Silberkugel erlegen kann.

Auf Grund der Komplexität der Software-Produkte ist es chancenlos, die Korrektheit von Software am End-Produkt durch Tests u.ä. festzustellen. Man kann den Werwolf Software nur an die Leine legen, indem man den Entwicklungsprozess definiert, den die Entwickler anwenden müssen: wie man ein Kind an der Hand über eine gefährliche Straße führt.

Vorgehensmodelle sind eine Abstraktion von bestehenden, erfolgreichen Entwicklungsprozessen (ähnlich einer Klasse in der objekt-orientierten Programmierung), die den Ablauf von zukünftigen Software-Entwicklungsprozessen festlegen, wobei aber der Prozess für jedes individuelle Projekt an die äußeren Projektparameter (Aufwand, Termin, Können der Entwicklungsmannschaft, Qualitätsanforderungen, Risiko) anzupassen ist.

Die bindende Festlegung eines definierten Vorgehensmodells bietet einige **wesentliche Vorteile**:

- Ein Vorgehensmodell abstrahiert vom individuellen Projekt und legt somit die generelle Entwicklungsmethodik (die 'Software-Kultur') fest.
- Es erzwingt die korrekte Abfolge der notwendigen Entwicklungsschritte und sichert die Einheitlichkeit von Methoden und Werkzeugen im Projekt.
- Es erlaubt eine verbesserte vorausschauende Planung.
- Es macht die Entwicklung vom individuellen Entwicklungsingenieur unabhängig.
- Die Lehr- und Lernbarkeit des in einem Unternehmen vorgegebenen Entwicklungsprozesses wird erleichtert.
- Der Prozess kann (unabhängig vom konkreten Produkt) theoretisch betrachtet, analysiert und durch Erfahrung zu verbessert werden.

- Es dient auch zur Bewertung und Zertifizierung eines Software-Hauses im Sinne von ISO 9000, CMM oder Bootstrap.

In ihrer Grundkonzeption bestehen Vorgehensmodelle aus der Beschreibung der Aktivitäten und der von den Aktivitäten erzeugten und verwendeten Daten, sowie in der Festlegung der Ausführungsreihenfolge dieser Aktivitäten.

Dadurch wird auch im wesentlichen die 'Strategie' der Software-Entwicklung festgelegt: in welcher Reihenfolge die Aktivitäten (eventuell auch mehrfach) durchlaufen werden.

Die aktuell propagierten **Vorgehensmodelle unterscheiden sich**

- durch den **Detaillierungsgrad**: vom einfachen Boehm'schen Wasserfall-Phasenmodell bis zu detailliert ausgearbeiteten Modellen wie V-Modell oder Hermes
- durch die Festlegung der **Sequentialisierungsreihenfolge** der Entwicklungsschritte: vom linearen Durchlauf wie z.B. beim Wasserfall-Modell bis zu komplexen, iterativen Ansätzen wie bei Spiralmodell und Rational Unified Process (RUP).
- durch die Berücksichtigung der **begleitenden Prozesse**: keine bei Wasserfall-Modell und RUP, Qualitätsmanagement und Projektmanagement bei IBM's ADPS und zusätzlich Konfigurationsmanagement beim V-Modell.

Heute eingesetzte Vorgehensmodelle müssen vor ihrem effektivem Einsatz an die Entwicklungsorganisation und ihre etablierten Konventionen und Usancen (die 'Software-Kultur') angepasst werden.

Sie müssen auch in eine entsprechende Software-Entwicklungsumgebung (Software Engineering Environment) eingebettet werden, die sowohl methoden-kompatible Entwurfswerkzeuge (z.B. UML-Diagramme) als auch entsprechende Verwaltungsmittel für Zwischen- und Endergebnisse bietet.

Vorgehensmodelle sind ein **wesentlicher Beitrag zur qualitativen Verbesserung des Software-Entwicklungs-Prozesses**, indem sie Systematik, Vollständigkeit und korrekte Abfolge der einzelnen Aktivitäten weitgehend sicherstellen.

Vollständiger Knowledge Letter Zugang

Wir freuen uns, dass Sie an diesem Thema Interesse haben und den Knowledge Letter von Software Quality Lab bis hierher gelesen haben.



Dieser Knowledge Letter ist eine Vorschau (gekürzte Version des gesamten Artikels).

Wenn Sie den ungekürzten Knowledge Letter lesen möchten, registrieren Sie sich bitte unter <http://www.software-quality-lab.com/download/knowledge-letter/anfrage-knowledge-letter/>

Sie erhalten nach der Registrierung vollen Zugang zu allen bisherigen Knowledge Letters von Software Quality Lab und erhalten automatisch künftige Knowledge Letter per E-Mail.

Software Quality Days — Die größte Konferenz zum Thema „Software Qualität“ in Europa!



Besuchen Sie die Top-Konferenz mit allen Infos rund um Software Qualität.

Beste Qualität der Vorträge und Tutorials sowie eine Mischung aus praktischen und wissenschaftlichen Beiträgen machen die Software Quality Days zum Top-Event.

In den 3 praktischen Tracks werden anwendungsorientierte Vorträge präsentiert. Der wissenschaftliche Track zeigt Beiträge mit hohem Innovationsgrad und praktischer Anwendbarkeit, basierend auf Forschungsergebnissen. Im Solution Provider Forum präsentieren Aussteller ihre neuesten Tools mit Praxis-Beispielen.

Nähere Infos unter

www.software-quality-days.com

