

Exzerpt 1:

Dokument:

hartzendorf-ausarbeitung-parnas.pdf

| | |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Preview: | <p>Überschriften:</p> <ol style="list-style-type: none">1. Biografie2. Stand der Modularisierung bevor es das Geheimnisprinzip gab3. Der Weg zum Geheimnisprinzip4. Geheimnisprinzip<ol style="list-style-type: none">4.1 Das Geheimnisprinzip ist schwieriger als es aussieht4.2 Hierarchische Modulare Struktur für Komplexe Systeme4.3 Es gibt keine Abstraktionsebenen5. Folgen des Konzepts und Ausblick<ol style="list-style-type: none">5.1 Das Geheimnisprinzip in drei neueren Konzepten5.2 Zukünftige Forschung: Standard-Information-Hiding-Designs <p>Abbildungen und Diagramme:</p> <ul style="list-style-type: none">- Abb. Struktogramm eines strukturierten Codes (jedoch nur Kontrollfluss)- Abb. Struktogramm eines strukturierten Codes mit eingezeichnetem Datenfluss- Abb. abstrahierter Datenaustausch (ohne Geheimnisprinzip)- Abb. realer Datenaustausch (ohne Geheimnisprinzip) |
| Question: | <ol style="list-style-type: none">[1] Wer war Parnas?[2] Wie sahen Programmkonstrukte vor dem Geheimnis-, Modularisierungs-Prinzip aus?[3] Wie sieht das Geheimnisprinzip aus?[4] Was sind die Folgen und Auswirkungen?[5] Heutige Anwendung?[6] Zukünftige Anwendung? |
| Read: | <ul style="list-style-type: none">- Zu [1]: David Lorge Parnas *10.02.1941 in New York, Ph.D. in Electrical Engineering, heute Professor für Software Engineering an der Universität von Limerick in Irland, bekannt durch Kritik an SDI, viele Veröffentlichungen über Software Engineering,- Zu [2]: „Goto-Spagetti“- Code- Zu [3]: Zugriffe auf Teile einer Programmeinheit, die für die „reguläre Benutzung“ nicht erforderlich sind, sollten verboten sein“- Zu [4]: Modularer Aufbau (Klassen/Objekte), Zugriffsschnittstellen, OOP, OOD, es gibt keine Abstraktionsebenen, Vererbung ist kontraproduktiv- Zu [5]: Abstrakte Datentypen, Objektorientierte Sprachen, Komponentenorientierte Programmierung, Frameworks- Zu [6]: Standardstrukturen für Programmklassen, welche Modulstrukturen und Modulschnittstellen für bestimmte Geräte, Datenstrukturen, notwendige Klassen und Algorithmen enthalten |

| | |
|-----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | sollen |
| Reflect: | <p>Bekannte Aspekte:</p> <ul style="list-style-type: none"> - Geheimnisprinzip an sich (wird hier jedoch tiefer beleuchtet) - Auswirkungen auf Heute und Zukunft (wird im Text jedoch vertieft) <p>Unbekannte Aspekte:</p> <ul style="list-style-type: none"> - Biografie Parnas - Weg zum Geheimnisprinzip (Parnas Ideen) |
| Recite: | <p>Der Text beginnt mit einer Biografie von Parnas. Danach geht er auf den Stand vor dem Geheimnisprinzip ein (sog. Goto-Spagetticode). Die damalige Unstrukturiertheit zeigt er auch anhand eines Struktogramms. Des Weiteren veranschaulicht er den Weg über Modularisierung und globalen Variablen hin zum Geheimnisprinzip. Dies wird auch mit Parnas Gedankengängen veranschaulicht (Clean Design, Datenkapselung, Schnittstellenspezifikation). Der Text stellt auch kurz die zwei wichtigsten Artikel vor, welche Parnas zum Thema „Information Hiding“ verfasst hatte. Darauf folgend gibt der Text Gründe für das Scheitern des Geheimnisprinzips in Programmierprojekt an (Flussdiagramminstinkt, ...). Am Ende der Arbeit stellt der Autor noch Folgen des Konzepts, so wie einen Ausblick auf die Zukunft, dar (Heute: Abstrakte Datentypen, Objektorientierte Programmierung, Objektorientierte Sprachen, Vererbung, Komponenten-Orientierte Programmierung / Zukunft: Standardstrukturen für Programmklassen, welche Modulstrukturen und Modulschnittstellen für bestimmte Geräte Datenstrukturen, notwendige Klassen und Algorithmen enthalten.</p> |
| Review: | <p>In diesem Text wird der Weg hin zum Geheimnisprinzip, das Geheimnisprinzip selbst und ein Ausblick auf die Zukunft dargestellt. Diese Arbeit beschäftigt sich sehr in die Breite mit dem Geheimnisprinzip. Bietet somit einen sehr guten Überblick. Bei manchen Punkten sollte man mit zusätzlicher Literatur mehr in die Tiefe gehen.</p> |

Keywords:

Abstrakte Datentypen,
 abstrakte Schnittstelle (Java),
 Abstraktionsebenen,
 Aggregation,
 Auswechselbarkeit,
 Biografie Parnas,
 Design, Flussdiagramm Instinkt,
 Geheimnisprinzip,
 Goto-Spagetticode,
 Informationsverteilung,
 Komponenten-Orientierte Programmierung,
 Modularisierung,

Objektorientierte Sprachen,
Schnittstellenspezifikationen,
strukturierter Code,
Uses-Relation,
Vererbung,
Wiederverwendbarkeit