

# Zur Information

- Im Semester 2024 ist das Kapitel 11.  
Softwarewartung **nicht Klausurrelevant!**



Vorlesung Softwaretechnik I (SS 2024)

# 11. Softwarewartung

Prof. Dr. Jens Grabowski

Tel. 39 172022

[grabowski@informatik.uni-goettingen.de](mailto:grabowski@informatik.uni-goettingen.de)



# Inhalt

- **Einführung und Definitionen**
- Modell für die Softwarewartung
- Organisation der Softwarewartung
  - Issue & Bug Tracking Systeme
- Lernziele

# Verschleiß und Wartung

- Wartung wird wg. Verschleiß durch Abnutzung durchgeführt!
- ... nur Software ist ein immaterielles Produkt und nutzt sich nicht ab.
- Softwarewartung beschreibt notwendige Änderungen aufgrund von
  - ausgelieferten Fehlern,
  - externen Einflüssen (z.B. Treiber, Betriebssysteme, gesetzlicher Rahmen),
  - neue bzw. geänderte Wünsche oder Bedürfnisse der Nutzer.





# Betriebs- und Wartungsphase

- Definitionen (IEEE Std 610.12, 1990):
  - **Operation and maintenance phase** – The period of time in the software life cycle during which a software product is employed in its operational environment, monitored for satisfactory performance, and modified as necessary to correct problems or to respond to changing requirements.
  - **Maintenance** – The process of modifying a software system or component after delivery to correct faults, improve performance or other attributes, or adapt to a changed environment.

Synonym: software maintenance

See also: adaptive maintenance, corrective maintenance, perfective maintenance.



# Arten der Wartung

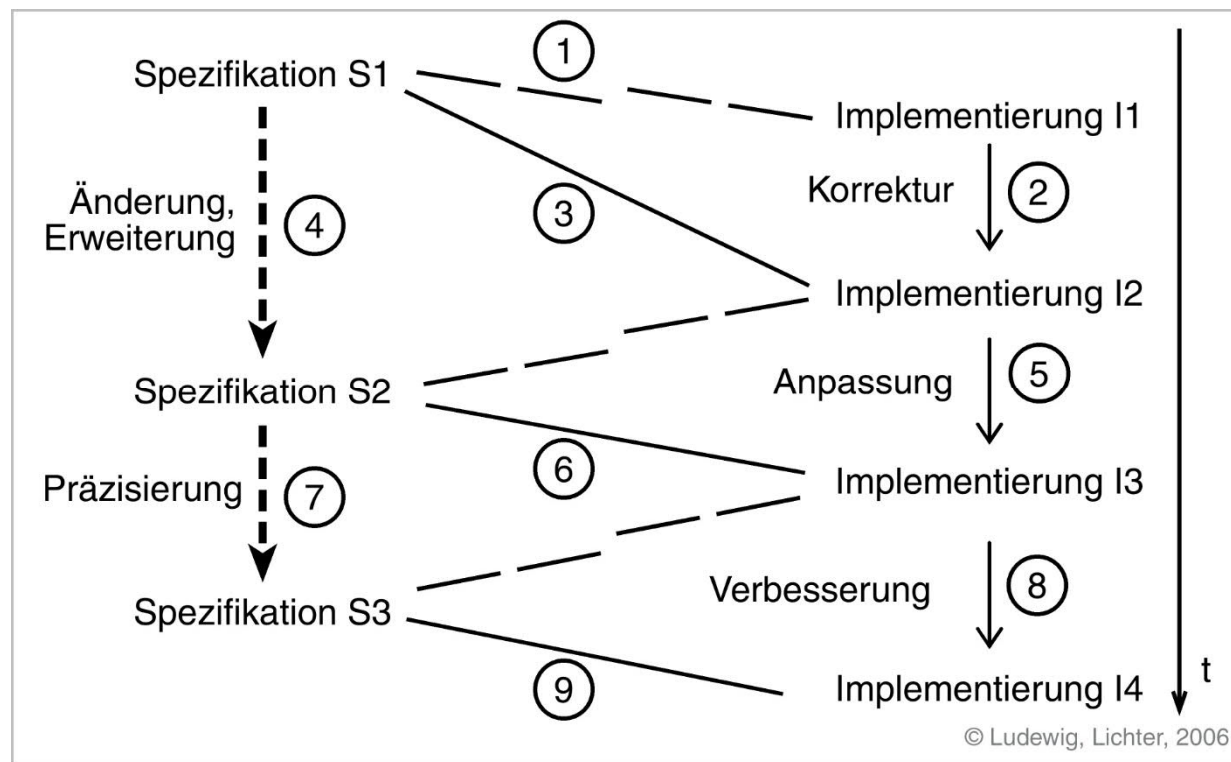
- Definitionen (IEEE Std 610.12, 1990) cont.:
  - **Adaptive maintenance** – Software maintenance to make a computer program usable in a changed environment.
  - **Corrective maintenance** – Maintenance performed to correct faults in software.
  - **Perfective maintenance** – Maintenance performed to improve the performance, maintainability, or other attributes of a computer program.
  - **Preventive maintenance** – Maintenance performed for the purpose of preventing problems before they occur.



# Inhalt

- Einführung und Definitionen
- **Modell für die Softwarewartung**
- Organisation der Softwarewartung
  - Issue & Bug Tracking Systeme
- Lernziele

# Modell für die Softwarewartung







# Modell für die Softwarewartung

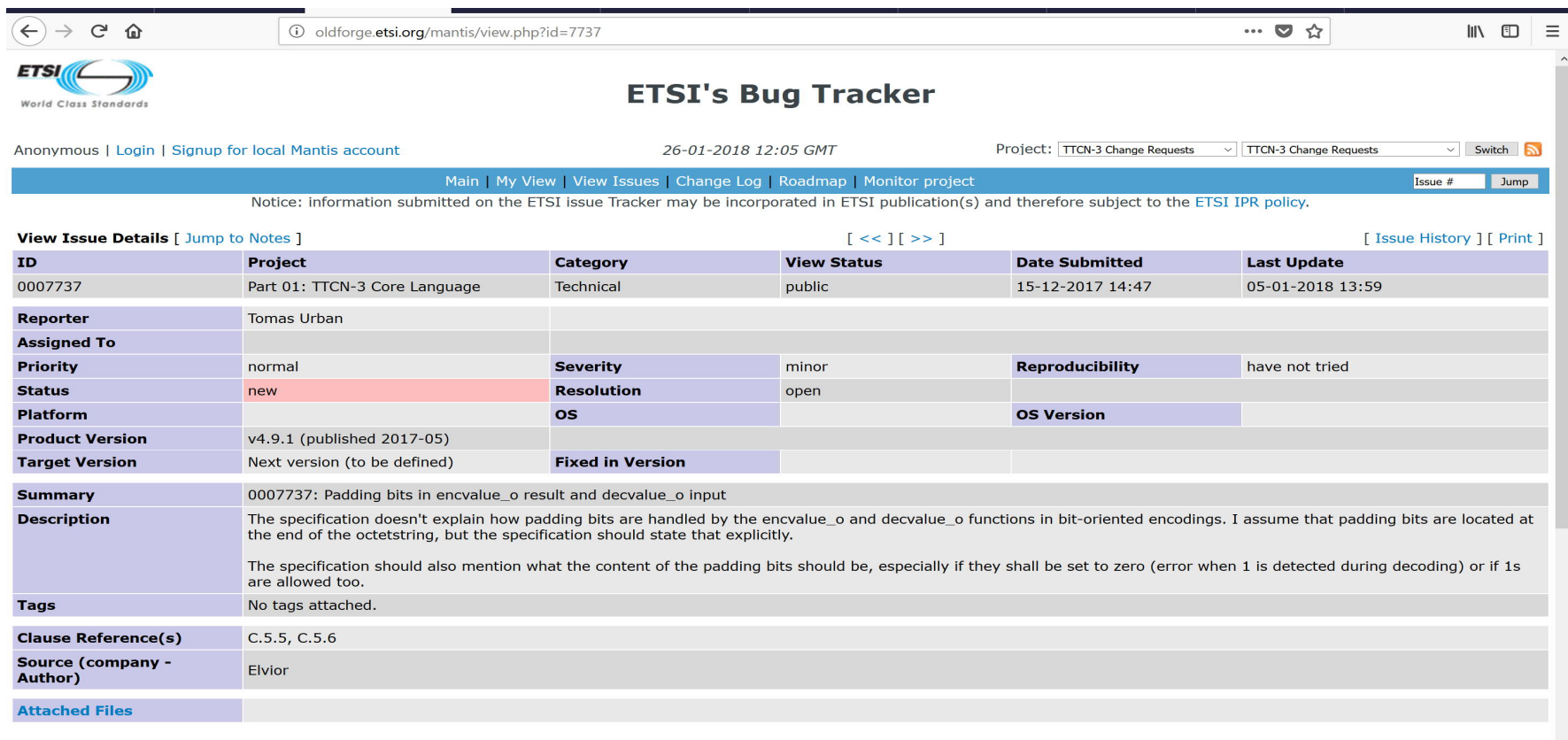
- Implementierungsfehler:
  - (1) Implementierung entspricht nicht der Spezifikation.
  - (2) Korrektur der Implementierung stellt
  - (3) Konsistenz mit Spezifikation wieder her.
- Änderung/Erweiterung der Anforderungen:
  - (4) Spezifikation muss geändert/erweitert werden.
  - (5) Anpassung der Implementierung stellt
  - (6) Konsistenz mit Spezifikation wieder her.
- Verbesserungswunsch des Kunden
  - (7) Präzisierung der Spezifikation.
  - (8) Verbesserung der Implementierung stellt
  - (9) Konsistenz mit Spezifikation wieder her.



# Inhalt

- Einführung und Definitionen
- Modell für die Softwarewartung
- **Organisation der Softwarewartung**
  - Issue & Bug Tracking Systeme
- Lernziele

# Wartungsorganisation – Issue & Bug Tracking Systeme



The screenshot displays the ETSI's Bug Tracker web interface. At the top, the ETSI logo and "World Class Standards" tagline are visible. The page title is "ETSI's Bug Tracker". Below the header, there is a navigation bar with links: "Main", "My View", "View Issues", "Change Log", "Roadmap", and "Monitor project". A notice states: "Notice: information submitted on the ETSI Issue Tracker may be incorporated in ETSI publication(s) and therefore subject to the ETSI IPR policy." The main content area shows the details for issue ID 0007737. The issue is titled "Part 01: TTCN-3 Core Language" and is categorized as "Technical". It was submitted on 15-12-2017 14:47 and last updated on 05-01-2018 13:59. The reporter is Tomas Urban. The issue status is "new" (highlighted in red). The priority is "normal" and the severity is "minor". The resolution is "open". The platform is not specified. The product version is "v4.9.1 (published 2017-05)" and the target version is "Next version (to be defined)". The summary states: "0007737: Padding bits in encvalue\_o result and decvalue\_o input". The description explains that the specification doesn't explain how padding bits are handled by the encvalue\_o and decvalue\_o functions in bit-oriented encodings. It also mentions that the specification should state that explicitly. The tags section shows "No tags attached." The clause reference is "C.5.5, C.5.6" and the source (company - Author) is "Elvior". There are no attached files.

ETSI's Bug Tracker

Anonymous | [Login](#) | [Signup for local Mantis account](#) 26-01-2018 12:05 GMT Project: [TTCN-3 Change Requests](#) [TTCN-3 Change Requests](#) [Switch](#)

[Main](#) | [My View](#) | [View Issues](#) | [Change Log](#) | [Roadmap](#) | [Monitor project](#) [Issue #](#) [Jump](#)

Notice: information submitted on the ETSI Issue Tracker may be incorporated in ETSI publication(s) and therefore subject to the ETSI IPR policy.

[View Issue Details](#) [ [Jump to Notes](#) ] [ << ] [ >> ] [ [Issue History](#) ] [ [Print](#) ]

ID	Project	Category	View Status	Date Submitted	Last Update
0007737	Part 01: TTCN-3 Core Language	Technical	public	15-12-2017 14:47	05-01-2018 13:59
<b>Reporter</b>	Tomas Urban				
<b>Assigned To</b>					
<b>Priority</b>	normal	<b>Severity</b>	minor	<b>Reproducibility</b>	have not tried
<b>Status</b>	new	<b>Resolution</b>	open		
<b>Platform</b>		<b>OS</b>		<b>OS Version</b>	
<b>Product Version</b>	v4.9.1 (published 2017-05)				
<b>Target Version</b>	Next version (to be defined)	<b>Fixed in Version</b>			
<b>Summary</b>	0007737: Padding bits in encvalue_o result and decvalue_o input				
<b>Description</b>	The specification doesn't explain how padding bits are handled by the encvalue_o and decvalue_o functions in bit-oriented encodings. I assume that padding bits are located at the end of the octetstring, but the specification should state that explicitly.  The specification should also mention what the content of the padding bits should be, especially if they shall be set to zero (error when 1 is detected during decoding) or if 1s are allowed too.				
<b>Tags</b>	No tags attached.				
<b>Clause Reference(s)</b>	C.5.5, C.5.6				
<b>Source (company - Author)</b>	Elvior				
<b>Attached Files</b>					



# Issue & Bug Tracking Systeme - Problem- oder Fehlermeldung

- Verwendung eines Issue & Bug Tracking Systems
  - sorgt für ein projektweit/organisationsweit einheitliches Schema für Problem- und Fehlermeldungen.
  - sorgt für eine reibungslose und transparente Kommunikation von Problem- und Fehlermeldungen.
  - erlaubt eine statistische Auswertung aller Meldungen.
  
- Typischerweise besteht eine Problemmeldung aus 3 Teilen:
  - Identifikation des Problems
  - Klassifikation des Problems
  - Problembeschreibung



# Issue & Bug Tracking Systeme - Problem- oder Fehlermeldung

Identifikation des Problems

Attribut	Bedeutung
Nummer	laufende, eindeutige Meldungsnummer
Produkt	ID des Produkts mit dem beobachteten Problem
Version	ID der genauen Version des Produkts
Plattform	Identifikation der HW-/SW-Plattform bzw. der Produktumgebung in der das Problem auftritt
Entdecker	ID des Meldenden
Entwickler	ID des für das Produkt verantwortlichen Entwicklers oder Teams
Erfassung	Datum, ggf. Uhrzeit an dem das Problem beobachtet wurde



# Issue & Bug Tracking Systeme - Problem- oder Fehlermeldung

## Klassifikation des Problems

Attribut	Bedeutung
Status	Bearbeitungsfortschritt der Meldung; möglichst mit Kommentar und Datum des Eintrags
Klasse	Klassifizierung der Schwere des Problems
Priorität	Klassifizierung der Dringlichkeit einer Korrektur
Anforderung	Verweis auf die (Kunden-)Anforderung, die wegen der Problemwirkung nicht erfüllt oder verletzt ist
Fehlerquelle	Soweit feststellbar, Projektphase in der die Fehlhandlung begangen wurde (Analyse, Design, Programmierung); nützlich zur Planung prozessverbessernder Maßnahmen



# Issue & Bug Tracking Systeme - Problem- oder Fehlermeldung

## Problembeschreibung

Attribut	Bedeutung
Reproduzierbarkeit	Beschreibung des Testfalls oder der Schritte zur Reproduktion der Problems
Problem	Beschreibung der Fehlerwirkung; erwartete vs. tatsächliche Ergebnisse bzw. Verhalten
Kommentar	Stellungnahmen der Betroffenen zum Meldungsinhalt
Korrektur	Korrekturmaßnahmen des zuständigen Entwicklers
Verweis	Querverweise auf andere zugehörige Meldungen



# Issue & Bug Tracking Systeme - Problem- oder Fehlermeldung

- Wichtig ist, dass alle Informationen erfasst werden, die für das Reproduzieren und Lokalisieren eines potenziellen Fehlers notwendig sind, sowie Informationen, die eine Auswertung der Produktqualität und des Korrekturfortschritts ermöglichen.
- Unabhängig vom vereinbarten Schema gilt
  - Eine Meldung muss so abgefasst sein, dass der zuständige Entwickler mit minimalem Aufwand das Problem nachvollziehen und dessen Ursache finden kann.
  - Die Reproduktion, Lokalisierung und Behebung eines Problems bedeuten für den Entwickler in der Regel nicht eingeplante, zusätzliche Arbeit; in dieser Situation neigen Entwickler dazu, jede unklare Fehlermeldung ohne weitere Analyse abzulehnen oder zurückzustellen.





# Issue & Bug Tracking Systeme - Problem- oder Fehlermeldung

- Ein wichtiges Kriterium zur Beurteilung eines gemeldeten Problems ist dessen **Schwere**.
  - D.h. nach (DIN 66271, 1995) der Grad der Beeinträchtigung des Produkteinsatzes.
- Die **Schwere** eines Fehlers sollte aus Sicht des Anwenders des Produkts vergeben werden.
- Aus dieser Einordnung muss nicht folgen, wie schnell das jeweilige Problem zu beheben ist.
  - Hier gehen zusätzlich Anforderungen des Produkt- oder Projektmanagements ein (z.B. Korrekturaufwand).
- Problem**priorität** legt die Dringlichkeit der Beseitigung fest.




# Issue & Bug Tracking Systeme - Problemklassifikation (Schwere)

Klasse	Bedeutung
1	Systemabsturz mit ggf. Datenverlust; das Testobjekt ist in dieser Form nicht einsetzbar
2	Wesentliche Funktion ist fehlerhaft; Anforderung nicht beachtet oder falsch umgesetzt; das Testobjekt ist nur mit großen Einschränkungen einsetzbar
3	Funktionale Abweichung bzw. Einschränkung (»normaler« Fehler); Anforderung fehlerhaft oder nur teilweise umgesetzt; System kann mit Einschränkungen genutzt werden
4	Geringfügige Abweichung; System kann ohne Einschränkung genutzt werden (Beseitigung hat mit dem nächsten Release zu erfolgen)
5	Schönheitsfehler (z.B. Mangel im Maskenlayout); System kann ohne Einschränkung genutzt werden (Beseitigung kann mit einem der nächsten Releases erfolgen)



# Issue & Bug Tracking Systeme - Problemklassifikation (Priorität)

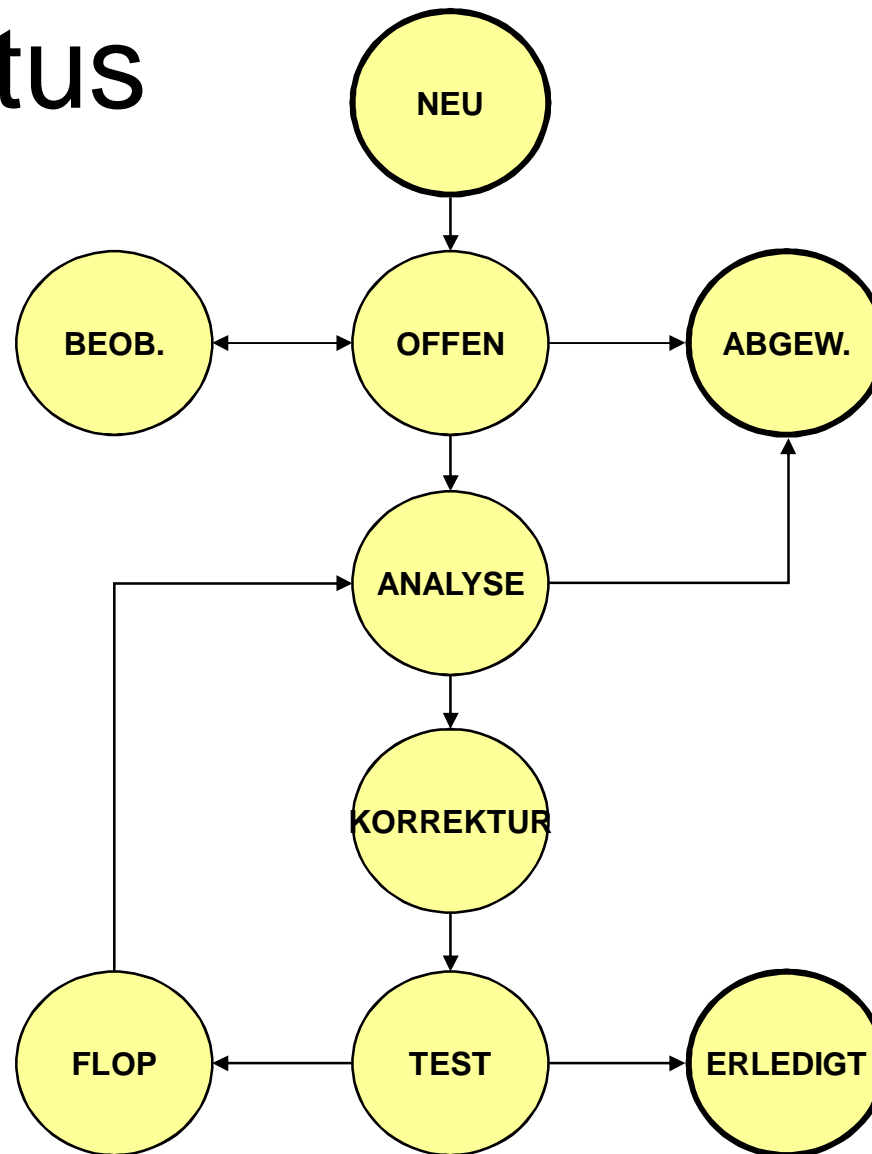
Priorität	Bedeutung
1 – Patch	Der Arbeitsablauf beim Anwender ist blockiert. Das Problem muss unmittelbar, ggf. provisorisch, behoben werden; ein Patch ist zu erstellen
2 – Nächste Version	Die Fehlerkorrektur erfolgt mit der nächsten regulären Produktversion oder der nächsten Testobjektlieferung
3 – Gelegentlich	Die Fehlerkorrektur erfolgt, sobald die betroffenen Systemteile ohnehin überarbeitet werden
4 - Offen	Korrekturplanung ist noch zu treffen



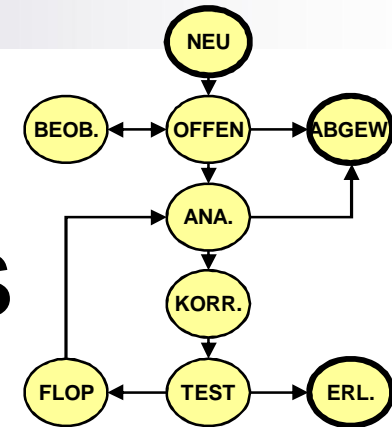
# Issue & Bug Tracking Systeme - Problemstatus

- Das Projektmanagement muss sicherstellen, dass Probleme ordentlich erfasst und verwaltet werden und dafür sorgen, dass Fehlerzustände zügig korrigiert werden und dass diese Korrekturen mit den neuen Versionen des Produkts verfügbar sind.
- Hierzu ist eine kontinuierliche und transparente Verfolgung des Problemanalyse- und Korrekturprozesses über alle Stadien notwendig.
- Diese Verfolgung geschieht anhand des Problemstatus.
- Dabei durchläuft jede Problemmeldung eine Reihe festgelegter Stati, die die erstmalige Erfassung bis zur erfolgreichen Fehlerkorrektur beinhalten.

# Issue & Bug Tracking Systeme - Problemstatus

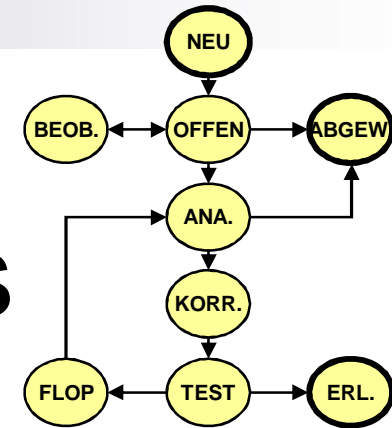


# Issue & Bug Tracking Systeme - Problemstatus



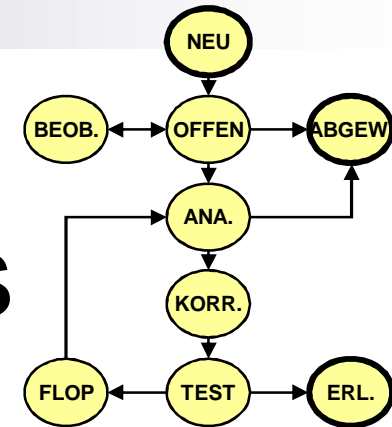
Status (gesetzt durch)	Bedeutung
NEU (Melder)	Neue Meldung wurde erfasst. Der Verfasser hat eine aus seiner Sicht sinnvolle Beschreibung und Klassifizierung eingetragen.
OFFEN (Projektmanager)	<p>Neue Meldungen werden regelmäßig vom Projektmanager gesichtet. Verständlichkeit und vollständige Vergabe der identifizierenden Attribute werden geprüft. Klassifizierende Attribute werden ggf. angepasst, so dass eine projektweit einheitliche Bewertung gegeben ist. Dubletten oder offensichtlich sinnlose Meldungen werden bereinigt bzw. zurückgewiesen.</p> <p>Sofern die Meldung nicht abgewiesen wird, wird sie einem zuständigen Entwickler zugewiesen und auf Offen gesetzt.</p>

# Issue & Bug Tracking Systeme - Problemstatus



Status (gesetzt durch)	Bedeutung
ABGEWIESEN (Projektmanager)	Meldung wird als unberechtigt abgewiesen (kein Defekt im Testobjekt, Änderungswunsch, der nicht berücksichtigt wird)
ANALYSE (Entwickler)	Der zuständige Entwickler setzt die Meldung auf diesen Status, sobald er die Meldung bearbeitet. Das Ergebnis der Problemanalyse (Fehlerursache, Lösungsmöglichkeiten, geschätzter Korrekturaufwand u.Ä.) wird in Kommentaren dokumentiert.
BEOBACHTUNG (Entwickler)	Das geschilderte Problem kann weder nachvollzogen noch ausgeschlossen werden. Die Meldung bleibt unerledigt, bis weitere Informationen oder Erkenntnisse vorliegen.

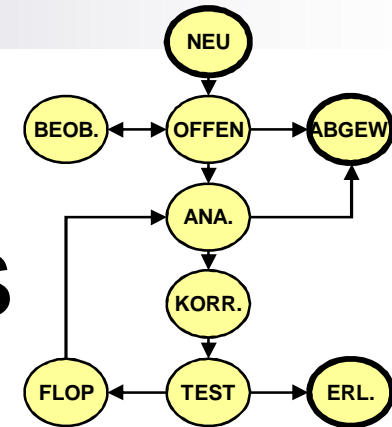
# Issue & Bug Tracking Systeme - Problemstatus



Status (gesetzt durch)	Bedeutung
KORREKTUR (Projektmanager)	Aufgrund der Analyse entscheidet der Projektmanager, dass die Korrektur erfolgen soll. Der zuständige Entwickler führt die Korrekturen durch. Die Art der Korrektur wird in Kommentaren dokumentiert.
TEST (Entwickler)	Der zuständige Entwickler setzt die Meldung auf diesen Status, sobald das Problem aus seiner Sicht behoben ist. Die Softwareversion, in der die Korrektur verfügbar ist, wird angegeben.



# Issue & Bug Tracking Systeme - Problemstatus



Status (gesetzt durch)	Bedeutung
ERLEDIGT (Entwickler, Tester)	Problemlösungen, also Meldungen im Status »TEST«, werden vom zuständigen Entwickler/Tester im nächstmöglichen Testzyklus verifiziert. Dazu wird mindestens der fehleraufdeckende Test wiederholt. Ergibt dieser Fehlernachtest, dass die Fehlerbeseitigung erfolgreich ist, beendet der Tester die Meldungshistorie im Endzustand »ERLEDIGT«.
FLOP (Entwickler, Tester)	Ergibt der Fehlernachtest, dass die Fehlerbeseitigung erfolglos oder ungenügend war, wird »FLOP« gesetzt. Eine erneute Analyse ist notwendig.



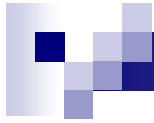
# Issue & Bug Tracking Systeme – Umgang mit Erweiterungswünschen

- Viele Problemmeldungen erfordern gar keine Fehlerkorrekturen, sondern echte (funktionale) Erweiterungen des Produkts.
- Da die Grenze zwischen Fehlermeldung und **Erweiterungswunsch** und die Beurteilung als **berechtigt** oder **unberechtigt** oft Ermessenssache sind, wird im Problemmanagement eine Instanz benötigt, die Änderungsanforderungen genehmigt oder ablehnt.
- Diese Instanz heißt **Änderungskontrollausschuss** (change control board) und wird üblicherweise von Vertretern folgender Interessengruppen gebildet:
  - ☐ Produktmanagement
  - ☐ Projektmanagement (inkl. Testmanagement)
  - ☐ Kundenvertreter



# Inhalt

- Einführung und Definitionen
- Modell für die Softwarewartung
- Organisation der Softwarewartung
  - Issue & Bug Tracking Systeme
- **Lernziele**



# Lernziele

- Was ist Softwarewartung und welche Arten der Softwarewartung gibt es?
- Wie kann sich Softwarewartung als Modell vorstellen?
- Wie kann man die Softwarewartung mit einem Issue & Bug Tracking System organisieren?
- Wie funktionieren Issue & Bug Tracking Systeme?
- Wozu benötigt man einen Änderungskontrollausschuss (change control board) und wie ist er üblicherweise zusammengesetzt?