

Vortragsreihe „ Software Engineering for Everyday Business“

Methoden der Qualitätssicherung

Teil 4b: Reviews, Inspektionen und Audits

Dietmar Winkler

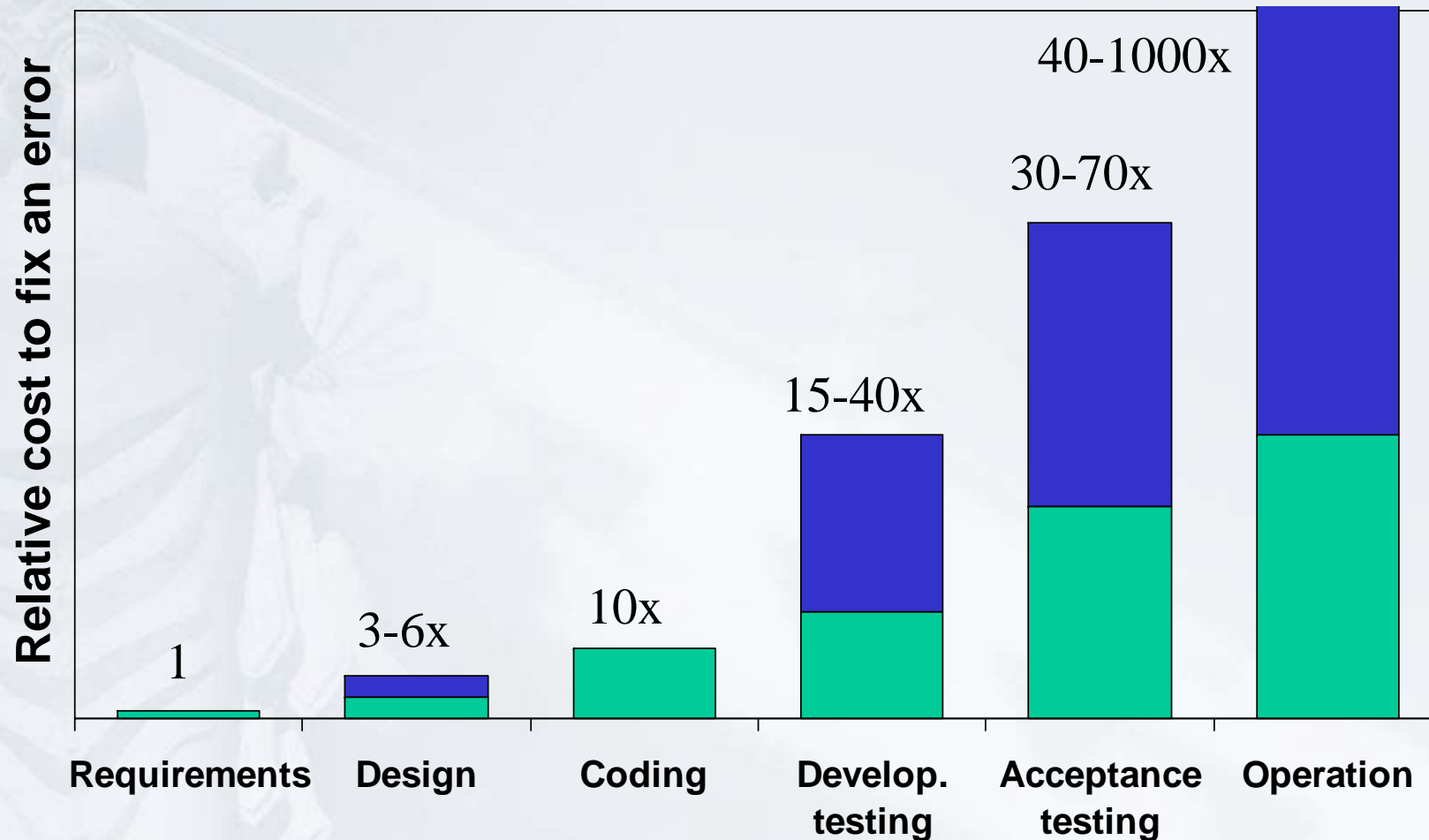
Technische Universität Wien
Institut für Softwaretechnik und Interaktive Systeme

dietmar.winkler@qse.ifs.tuwien.ac.at
<http://qse.ifs.tuwien.ac.at>

- **Teil 4a: Qualitätssicherung und Software Prozess Verbesserung (ca. 45')**
 - Qualität – einige Definitionen und Grundlagen
 - Qualitätssicherung im Software Engineering Prozess
 - Software Prozess Verbesserung
- **Teil 4b: Reviews, Inspektionen, Audits (ca. 45')**
 - Überprüfung der Qualität in frühen Phasen der Softwareentwicklung
 - Inspektionsprozess
 - Audits und Assessments
- **Teil 4c: Software Testen (ca. 60')**
 - Überprüfung der Qualität in „späten Phasen“ der Entwicklung.
 - Teststrategien
 - Testautomatisierung
- **Teil 4d: Best Practice Testing (ca. 30')**

- Die **Korrektur von Fehlern** in Softwareprodukten kann hohe Aufwände und Kosten verursachen.
- **Je später ein Fehler erkannt wird, umso höher sind diese Aufwände.**
- Zielsetzung ist es daher, entweder erst gar keine Fehler zu machen oder gemachte **Fehler frühzeitig zu erkennen und zu beheben.**
- **Analytische Methoden**, wie Reviews, Inspektionen und Audits dienen der Verbesserung der Produkt- und Prozessqualität in allen Phasen der Softwareentwicklung, speziell aber in **frühen Phasen der Softwareentwicklung.**
- **Reviews und Inspektionen** erfordern **keinen ausführbaren Code**, sondern können somit bereits auf z.B. Anforderungsdokumente, Designdokument angewandt werden.
- **Audits** dienen der objektiven Überprüfung von Produkten, Prozessen oder (Management-) Systemen und stellen u.a. auch die Grundlage für Zertifizierungen dar.

Korrekturaufwand von Fehlern im Projektverlauf



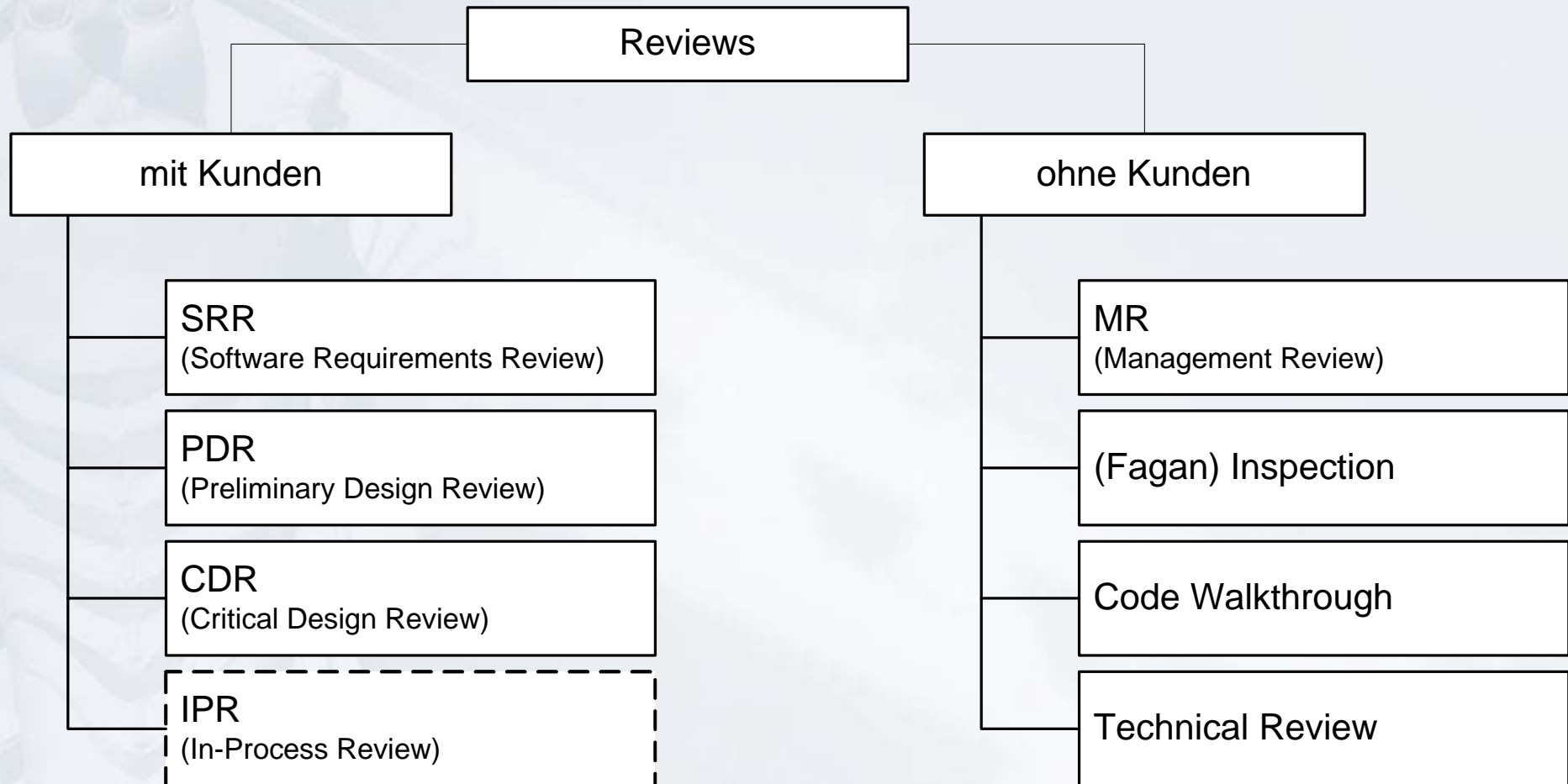
The cost of fixing errors escalates as we move the project towards field use.

From an analysis of sixty-three projects cited in Boehm Barry, „Software Engineering Economics“, Prentice-Hall, 1981

Definition

- „Ein Review ist ein [...] formal geplanter und **strukturierter Analyse- und Bewertungsprozess**, in dem Projektergebnisse einem **Team von Gutachtern** präsentiert und von diesen kommentiert oder genehmigt werden.“
[IEEE Std 610, Wallmüller 2001]
- Reviews dienen vor allem zur **qualitativen Beurteilung** von Produkten und Prozessen, die quantitativ nur schwer oder gar nicht beurteilt werden können (z.B. Modelle, Dokumente)
- Zentral ist die **Beurteilung des Produktes** und nicht des Autors!
- Unterschiedliche Ausprägungen von Reviews zielen auf unterschiedliche Ziele ab.
- In einen Reviewprozess sind definierte Rollen mit zugeordneten Aufgaben involviert.

Arten von Reviews [Thaller, 2000]

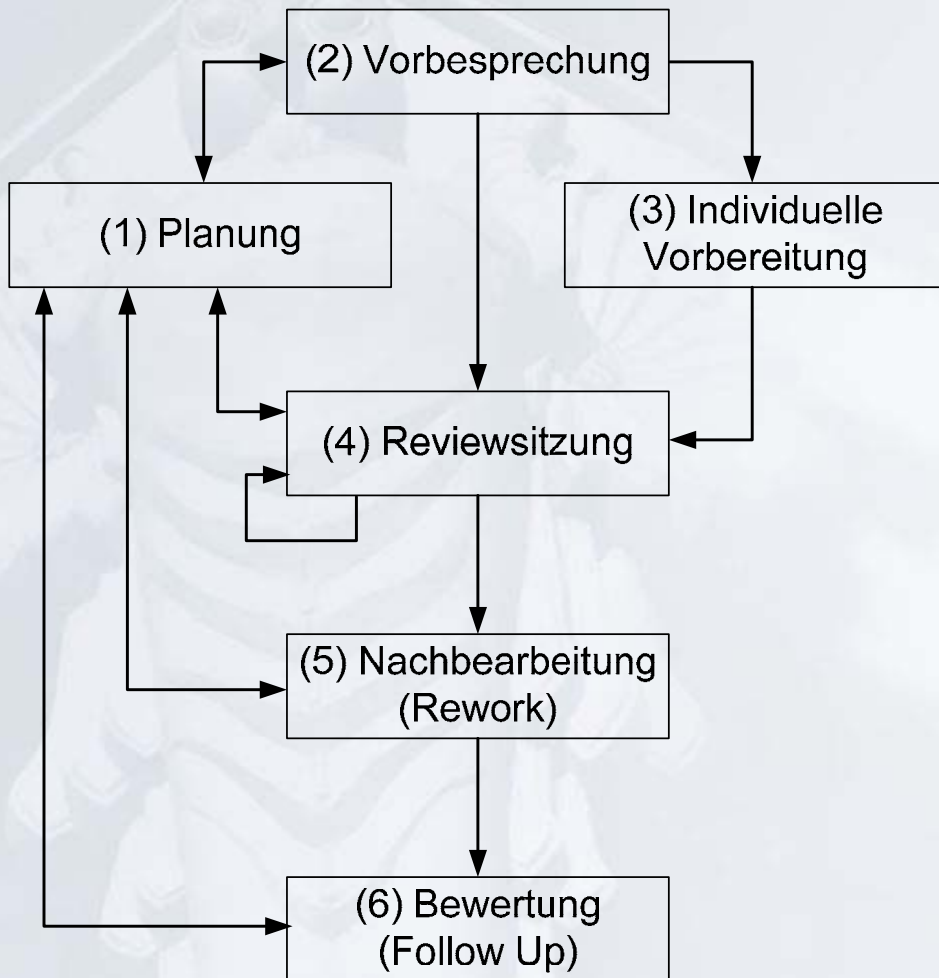


Vergleich Review, Inspektion, Walkthrough

(Technisches) Review	Inspektion (formales Review)	Code Walk-Through (abgeschwächtes Review)
Ziele <ul style="list-style-type: none">▪ Stärken und Schwächen des Prüfobjektes identifizieren▪ (Entwicklungsprozess verbessern)	Ziele <ul style="list-style-type: none">▪ Schwere Defekte im Prüfobjekt identifizieren▪ Entwicklungsprozess verbessern▪ Metriken ermitteln	Ziele <ul style="list-style-type: none">▪ Defekte und Probleme des Prüfobjektes identifizieren▪ Ausbildung von Benutzern und Mitarbeitern
Teilnehmer Moderator, Autor, Gutachter, Protokollführer	Teilnehmer Moderator, Autor, Gutachter, Protokollführer, (Vorleser)	Teilnehmer Autor (=Moderator), Gutachter
Charakteristika <ul style="list-style-type: none">▪ Ausgebildeter Moderator▪ Prüfteam gibt Empfehlung an Manager	Charakteristika <ul style="list-style-type: none">▪ Ausgebildeter Moderator▪ Prüfobjekt wird vom Vorleser Absatz für Absatz vorgetragen▪ Moderator gibt Freigabe	Charakteristika <ul style="list-style-type: none">▪ Prüfobjekt wird vom Autor ablauforientiert vorgetragen▪ Autor entscheidet

1. Das **Produkt** reviewen, nicht den Autor.
2. Verwenden Sie einen **Arbeitsplan**.
3. Diskussionen sachlich und kurz halten.
4. Problembereiche identifizieren, aber nicht jedes Problem gleich lösen.
5. Schriftliche **Aufzeichnungen** führen.
6. Anzahl der Teilnehmer begrenzen; gute Vorbereitung aller Teilnehmer.
7. Für jedes Produkt eine passende **Checkliste** verwenden.
8. Ausreichende **Ressourcen und Zeitbudget** zur Verfügung stellen.
9. Vorab **Training** für alle Reviewer.
10. Reviews im Nachhinein beurteilen für **Verbesserung der Reviews**.

- Die typische Größe eines Reviewteams ist (abhängig von der Projektart, -größe, usw.) im Bereich 3 bis 6 Personen, die sich auf folgende **Rollen** aufteilen:
 - **Moderator** („keeper of the process“)
Leiter des Reviews
 - **Leser** („keeper of focus and pace“)
 - **Schreiber** („preserver of knowledge“)
Protokollschreiber verfasst Protokoll (Notizen während der Reviewsitzung)
 - **Gutachter** („Reviewer“)
Kommentierung des Reviewobjektes.
 - **Autor** („author“)
Klärung von offenen Fragen.
Keine Kommentierung und Rechtfertigung der Lösungen.
- Häufig ist der Autor kein Mitglied des Reviewteams.
- **Wichtig: Das Produkt soll beurteilt werden – NICHT der Autor!**



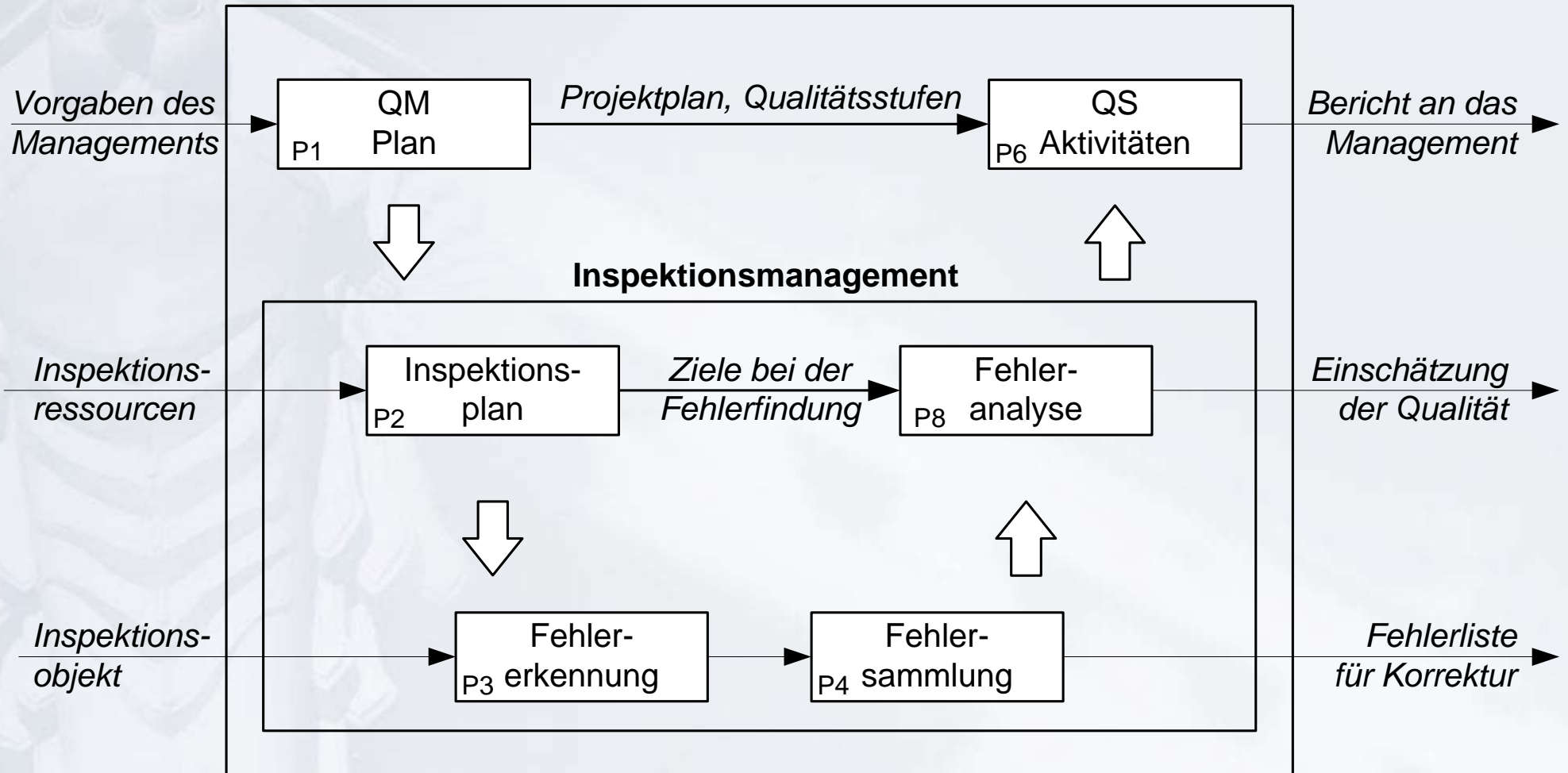
- **Planung:** Objekt, Prüfziele, Auslösekriterien (Einstiegsriterien), Teilnehmer, Ort, Zeit.
- **Vorbesprechung:** Vorstellung des Prüfobjekts bei komplexen und neuen Produkten.
- **Intensive Einzeldurcharbeitung**
- **Durchführung:** Gemeinsames Lesen, Aufzeichnung von Mängeln; während des Reviews sollen **Mängel entdeckt, nicht korrigiert** werden.
- In der **Nachbearbeitung** werden dokumentierte Mängel korrigiert und in der **Bewertung** überprüft.
- Berichterstattung.
- **Wiederholungen von Reviews** sind möglich.
- **Checklisten** unterstützen Reviews.
- Typische Dauer: **2h**

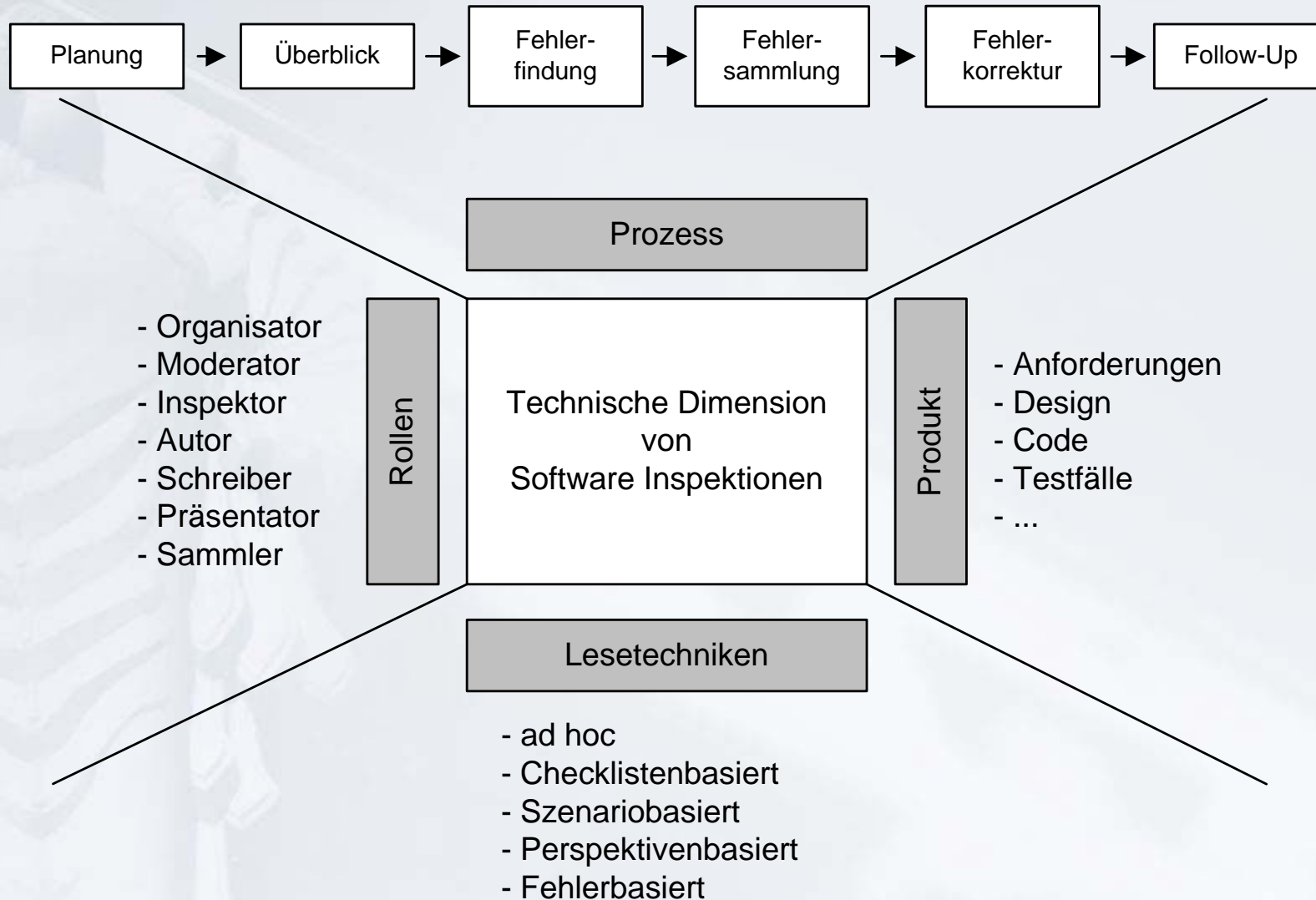
Wer soll an einem Review teilnehmen?

Personen	Produkte							
	Projektplan Analyse	Anforderungen	Analysemodell	Testplan	Projektpl. Realisierung	Entwurf	Implementierung	Installation
Kunde	●	●			●			●
Projektleiter	●	●			●			●
Analytiker		●		●		●		
Integrator						●	●	●
Entwickler						●	●	
Tester				●			●	
Qualitätsicherer				●	●			

- Inspektionen sind **spezielle Reviews**, die speziell in **frühen Phasen** der Softwareentwicklung angewandt werden können.
- Inspektionen sind **formale, effiziente und wirtschaftliche Methoden** um **Fehler im Design und Code** zu finden.
- Inspektionen ermöglichen eine **systematische und strukturierte** Fehlererkennung.
- **Lesetechniken** unterstützen die Inspektoren durch fokussiertes Lesen (Perspektiven, Rollen, Szenarien, Checklisten, usw.)
- **Teammeetings** dienen der Sammlung individueller Fehlerlisten (reale vs. nominelle Teams).
- Wichtige Anwendung: Fehlerfindung in Anforderungs- und Designdokumenten.

Qualitätsmanagement im Projektumfeld





Lesetechniken zur Fehlerfindung (1)

- Lesetechniken sind **strukturierte Ansätze** oder Dokumente, die den Fehlerfindungsprozess während einer Inspektion oder einer Review unterstützen.

Arten von Lesetechniken (Auswahl)

- Ad-hoc (unstrukturiertes Lesen)
- Checklistenbasiertes Lesen
 - Vordefinierte **Fragestellungen** die sequentiell auf das zu untersuchende Objekt angewandt werden.
 - Die Checklisten sind auf zu findende Fehlertypen, Fehlerorte (Design, Code) usw. abgestimmt.
 - In der Praxis meist **generische Checklisten + Projektspezifische Erweiterungen**.
 - **Einfache Anwendbarkeit** durch Inspektoren.

- Perspektivenbasiertes Lesen
 - Fehlersuche aus verschiedenen Perspektiven: **User, Tester, Entwickler**.
 - Zielsetzung ist es, **unterschiedliche Fehler** aus individuellen Sichten zu finden.
 - Erfordert entsprechende Fachspezialisten.

- Usage-Based Reading (UBR) → Best-Practice Inspection
 - Typischerweise existieren **Anwendungsfälle und Szenarien** (z.B. Use Cases)
 - **Priorisierung** von Anwendungsfällen durch Fachleute.
 - Zielsetzung: Fehler in **wichtigen Anwendungsfällen** werden zuerst gefunden.
 - Typische Vorgehensweise:
 - (1) Auswahl des wichtigsten Anwendungsfalls,
 - (2) Überprüfung der relevanten Teile, die diesen Anwendungsfall beschreiben,
 - (3) Reporting von Mängeln und Fehlern,
 - (4) Auswahl des nächsten Anwendungsfalls.

▪ Kosten

- Aufwand Inspektoren (Personenstunden); erhöht sich bei Teammeetings zusätzlich.
- Einplanung im Projektplan.

▪ Nutzen

- Einsparungen durch frühes Finden und Entfernen von Fehlern.
- Genauere **Informationen** über das Produkt und den Entwicklungsprozess zur Planung der Aktivitäten und Ressourcen (PM und QM).
- Beurteilungsgrundlage für die Produktqualität (z.B. durch Restfehlerschätzungen).
- Besseres **Produktwissen** der Inspektoren.
- **Gemeinsame Sicht** des Inspektionsteams auf das Produkt.
- Inspektion als **Lerninstrument** für neue Teammitglieder.

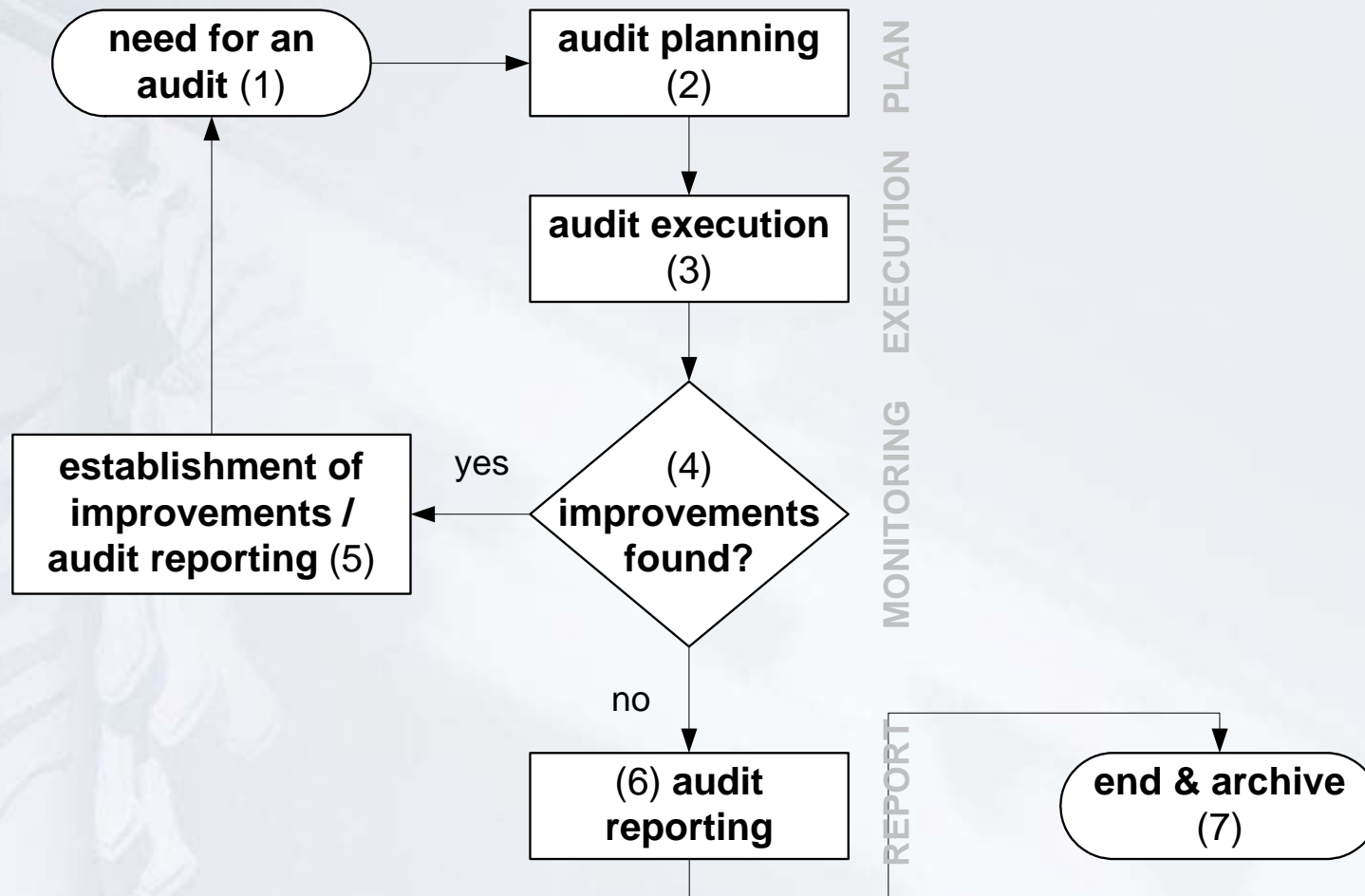
- „Ein Audit ist eine unabhängige Prüfung eines Arbeitsproduktes [..], bei der die Übereinstimmung mit Spezifikation, Standards, vertraglichen Vereinbarungen [..] überprüft wird“ [nach IEEE Std 619; Wallmüller, 2001]
- Audits werden von einer aussenstehenden Organisation bzw. einem Team durchgeführt. Sie sind typischerweise nicht in den Entwicklungsprozess integriert.
- Objektivität der Auditoren.
- Audits stammen aus der ISO 9000 Welt und dienen als Grundlage für Zertifizierungen.

Ziele:

- Praxis vs. Standards und Richtlinien (z.B. Spezifikation, QM-Handbuch)
- Überprüfung der Umsetzung.
- Einschätzung des Prüfobjektes.
- Checkliste über Erfüllung in Theorie (Dokumentation) und Praxis (Umsetzung).

QM-System (Überprüfung)			
Typ	Systemaudit	Prozessaudit	Produktaudit
Aufgabe	Untersuchung des QM-Systems oder ausgewählter Bereiche	Untersuchung eines Prozesses bezüglich Personal, Prozessüberwachung und Arbeitsabfolge	Untersuchung von (Teil-)Produkten im Hinblick auf die Spezifikation
Thema	Struktur des QM-Systems Arbeitsabläufe	Prozessabläufe Mitarbeiter	(Teil-)Produkte
Verbesserungsmassnahmen, Schwachstellenbereinigung			

Planung und Durchführung von Audits



- Bedarf an Audits (1)
 - Meilensteine gemäss Projekt- oder QM-Plan
 - Interne Prozessverbesserung
 - Externe Anforderung (Kunde, Zertifizierungen)
- Auditplanung (2)
 - Inhalt, Audit-Team und Kontaktpersonen
 - Formale Grundlagen (Normen, Checklisten)
 - Ressourcen- und Terminplanung
- Auditdurchführung (3)
 - gemäss Auditplan und formalen Grundlagen
 - Dokumentation der Ergebnisse
- Verbesserungen notwendig ? (4)
 - Diskussion über Stärken / Schwächen des auditierten Bereiches (mit Beteiligten)
 - Festlegung eines möglichen Wiederholungsaudits
- Umsetzung (5) und Berichterstattung bzw. Archivierung (6)

„Disziplinierte Bewertung der Software-Prozesse einer Organisation gegen ein Prozessmodell, wie es mit SPICE oder dem CMM vorliegt.“ [Thaller, 2000]

Ziele:

- Objektive Überprüfung der Organisation durch Externe.
- Praxis vs. Standards und Richtlinien (z.B. QM-Handbuch).
- Praktische Umsetzung von Normenvorgaben in der QM-Dokumentation und der Praxis.
- Audits und Assessments zielen zwar in dieselbe Richtung, unterscheiden sich aber in Umfang und Dauer.
- Basieren auf Interviews auf allen Ebenen (Prozessowner und ausführende Personen).

Assessments stammen aus der CMM/CMMI Welt.

▪ Zusammenfassung

- Analytische Methoden dienen der Verbesserung der Produkt- und Prozessqualität sowie der Einschätzung der Produkt- Prozess- und Projektqualität.
- Im Zentrum dieser analytischen Maßnahmen steht das zu überprüfende Produkt, nicht der Autor!
- Rollen für Reviews und Inspektionen erfüllen definierte Aufgaben.
- Die Durchführung von Reviews und Inspektionen folgt einem definierten Prozess.
- Lesetechniken unterstützen die Reviewer und Inspektoren bei der Fehlerfindung.
- Audits und Assessments stammen aus "unterschiedlichen Welten" und zielen auf eine unabhängige Überprüfung von Produkten und Prozessen im Hinblick auf Spezifikationen und Standards ab.

▪ Ausblick

- Einführung in Software Testen.

- Biffl, Stefan: „*Software Inspection Techniques to support Project and Quality Management*“, Shaker Verlag, 2001, ISBN: 3-8265-8512-7
- Biffl Stefan, Winkler Dietmar, Frast Denis: „Qualitätssicherung, Qualitätsmanagement und Testen in der Softwareentwicklung“, Skriptum zur Lehrveranstaltung, 2004.
<http://qse.ifs.tuwien.ac.at/courses/skriptum/script.htm>
- Laitenberger, Oliver: "Cost-Effective Detection of Software Defects Through Perspective-based Inspections"; PhD Theses in Experimental Software Engineering Vol. 1, 2000
- Software Engineering – Best practices:
<http://best-practice-software-engineering.blogspot.com/>
- Sommerville, Ian; Software Engineering; 6th Edition; Addison-Wesley, 2001; ISBN 0 201 39815 X
- Thaller, Georg Erwin; Software Qualität; der Weg zu Spitzenleistungen in der Software-Entwicklung; VDE-Verlag, 2000; ISBN 3-8007-2497-4
- Wallmüller, Ernest; Software Qualitätsmanagement in der Praxis; Software-Qualität durch Führung und Verbesserung von Software-Prozessen; 2., völlig überarbeitete Auflage, Hanser, 2001, ISBN 3-446-21367-8



Vielen Dank für die Aufmerksamkeit !

Kontakt: Dipl.-Ing. Dietmar Winkler

Technische Universität Wien
Institut für Softwaretechnik und Interaktive Systeme
Favoritenstr. 11/188, A-1040 Vienna, Austria

dietmar.winkler@qse.ifs.tuwien.ac.at

<http://qse.ifs.tuwien.ac.at>