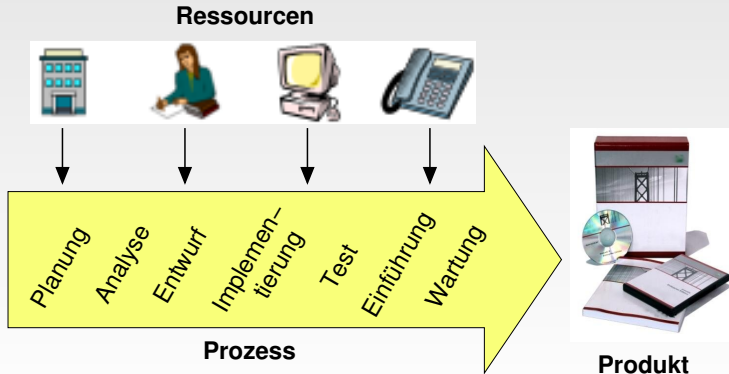
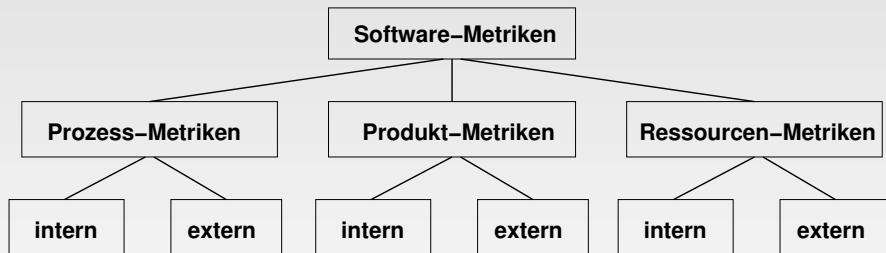


Klassifikation von Softwaremetriken

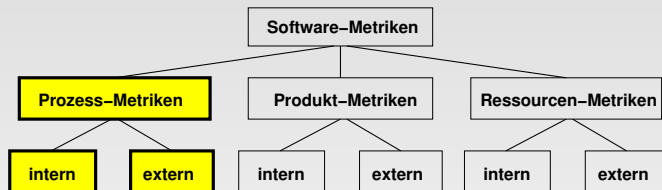
- Was: Ressource/Prozess/Produkt
- Wo: intern/extern (isoliert/mit Umgebung)
- Wann: in welcher Phase des Prozesses
- Wie: objektiv/subjektiv, direkt/abgeleitet



Klassifikation nach Fenton und Pfleeger (1998)



Prozessmetriken

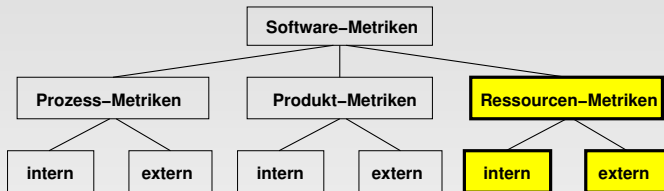


intern:

- Zeit/Dauer
- Aufwand
- Anzahl von Ereignissen
z.B. Fehler, Änderungen

extern:

- Qualität
- Kontrollierbarkeit
- Stabilität
- Kosten



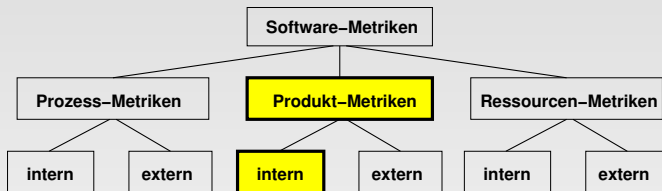
intern:

- Personal (Alter, Lohn)
- Teamgröße/-struktur
- Produktionsmaterialien
- Werkzeuge, Methoden

extern:

- Produktivität
- Erfahrung
- Kommunikation
- ...

Produktmetriken – intern



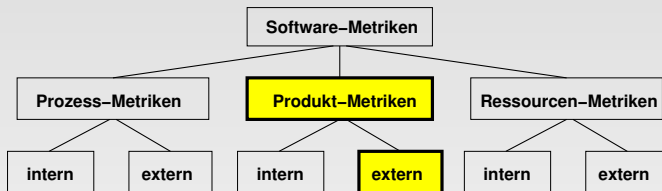
Größe:

- LOC
- Halstead
- Function Points
- Bang (DeMarco)

Komplexität:

- McCabe Cyclomatic Complexity
- Kontrollflussgraph
- Datenfluss
- OO-Metriken

Produktmetriken – extern



- Zuverlässigkeit
- Verständlichkeit
- Benutzerfreundlichkeit
- Performanz
- Portierbarkeit
- Wartbarkeit
- Testbarkeit
- ...

Vorteil: automatische Erfassung

Die Klassiker:

- LOC - Lines Of Code
- Halstead (1977)
- McCabe (1976)
- OO-Metriken (Chidamber und Kemerer 1994)

Lines of code (LOC)

- + relativ einfach messbar
- + starke Korrelation mit anderen Maßen
 - ignoriert Komplexität von Anweisungen und Strukturen
 - schlecht vergleichbar
- abgeleitet: Kommentaranteil

Physical source lines (COCOMO 2.0)

When a line or statement contains more than one type, classify it as the type with the highest precedence.

Statement type	Precedence	Included
Executable	1	✓
Nonexecutable		
Declarations	2	✓
Compiler directives	3	
Comments		
On their own lines	4	
On lines with source code	5	
Banners and non-blank spacers	6	
Blank (empty) comments	7	
Blank lines	8	

Physical source lines (COCOMO 2.0)

How produced	Included
Programmed	✓
Generated with source code generators	
Converted with automated translators	✓
Copied or reused without change	✓
Modified	✓
Removed	

Physical source lines (COCOMO 2.0)

Origin	Include
New work: no prior existence	✓
Prior work: taken or adapted from	✓
A previous version, build, or release	✓
Commercial off-the-shelf software (COTS), other than libraries	
Government furnished software (GFS), other than reuse libraries	
Another product	
A vendor-supplied language support library (unmodified)	
A vendor-supplied operating system or utility (unmodified)	
A local or modified language support library or operating system	
Other commercial library	
A reuse library (software designed for reuse)	✓
Other software component or library	✓

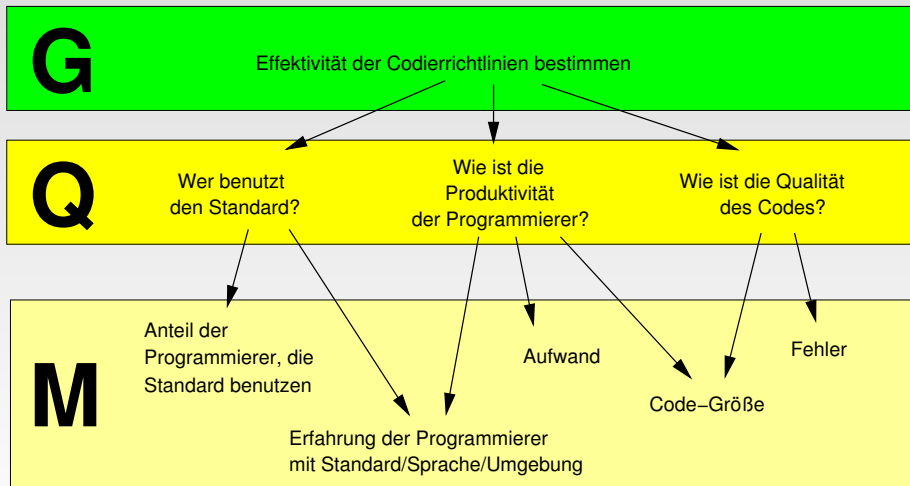
- Beurteilung des aktuellen Zustands
 - Projektüberwachung
 - Produktivität
 - Softwarequalität
 - Prozessqualität (CMM)
- Vorhersage des zukünftigen Zustands
 - Aufwandsabschätzung
 - Prognose für Wartungskosten

- Datenerfassung sehr aufwendig, zunächst wenig Nutzen
- Datenerfassung nicht konsistent
- Teilweise Messungen schwierig durchführbar
- Zweck der Messungen muss klar sein
- Integration der Datenerfassung in den normalen Arbeitsprozess
- Metriken müssen wohldefiniert und validiert sein
- Beziehungen zwischen Metriken müssen definiert sein
- Gefahr der Fehlinterpretation

GQM (Goal-Question-Metric; Basili und Weiss (1984))

Nicht das messen, was einfach zu bekommen ist,
sondern das, was benötigt wird

- ① Ziele erfassen
- ② zur Prüfung der Zielerreichung notwendige Fragen ableiten
- ③ was muss gemessen werden, um diese Fragen zu beantworten



Beispiel: Prozess

Ziel	Frage	Metrik
Maximiere Kundenzufriedenheit		

Beispiel: Prozess

Ziel	Frage	Metrik
Maximiere Kundenzufriedenheit	Wie viele Probleme treten beim Kunden auf?	
	Wie lange dauert Problembehebung?	
	Wo sind Flaschenhälse?	

Beispiel: Prozess

Ziel	Frage	Metrik
Maximiere Kundenzufriedenheit	Wie viele Probleme treten beim Kunden auf?	<ul style="list-style-type: none">• #Fehler (FR) und #Änderungswünsche (ÄR)• Zuverlässigkeit• Break/Fix-Verhältnis
	Wie lange dauert Problembehebung?	<ul style="list-style-type: none">• Verhältnis und Dauer offener und geschlossener FR/ÄR
	Wo sind Flaschenhälse?	<ul style="list-style-type: none">• Personálnutzung• Nutzung anderer Ressourcen

Beispiel: Produkt

Ziel	Frage	Metrik
Maximiere Verständlichkeit des Codes		

Beispiel: Produkt

Ziel	Frage	Metrik
Maximiere Verständlichkeit des Codes	Wie groß ist das Sys- tem? Wie komplex ist das System?	

Beispiel: Produkt

Ziel	Frage	Metrik
Maximiere Verständlichkeit des Codes	Wie groß ist das System?	<ul style="list-style-type: none">• Anzahl Funktionen, Klassen, Pakete etc.• Lines-of-Code pro Funktion, Klasse, Paket etc.
	Wie komplex ist das System?	<ul style="list-style-type: none">• McCabe-Komplexität pro Funktion, Klasse, Paket etc.• Schachtelungstiefe pro Funktion• Kopplung• Anzahl Funktionsaufrufe