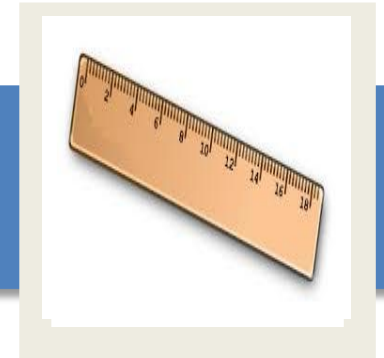


Návrh softwarových systémů - softwarové metriky

Martin Tomášek *Návrh softwarových systémů
(B6B36NSS)*



```
public final void onSensorChanged(SensorEvent event)
{
    m_flightIntensity = event.values[0];
    m_etAmblight.setText("" + m_flightIntensity + " lx");
}
...
resume()
...
light, ... NORMAL);
```

The problem with software testing metrics



"This pie chart shows the bugs I may have found and may not have found while making this chart."

Co je to metrika ?

- Nástroj managementu pro řízení zdrojů
- (lidská práce, peníze, čas atd.)
- Pro jejich správné použití
- Pro splnění cílů
- Zprostředkovávají zpětnou vazbu (feedback)-> **velmi důležité**
- Na základě feedbacku můžeme analýzu a následně zefektivnit proces
- Zlepšení přehlednosti do budoucna (jsme v polovině projektu ?
Na konci ?)



Co je to metrika ?

- U SW projektů se hůře odhadují některé
- atributy
- Zvýšíme motivaci, produktivitu
- (lidé vidí co funguje a co ne..)
- Zlepšení komunikace a předvídatelnost
- Metriky jsou nezbytné pro QA (Quality assurance)



Metrika čeho ?

- V SI se dlouhou dobu zaměřovalo pouze na metriku **zdrojového kódu**
- Kvalita zdrojového kódu je závislá na :
 - ***Kvalita SI procesu (vývojové metodiky...)***
 - ***Výběr a rozvoj lidských zdrojů***
- → Vznik odvětví **software craftsmanship**

Metriky

- Softwarové firmy sbírají metriky, protože jsou jediným nástrojem jak získat manažerské informace pro vývoj software a zajištění kvality.
- Důležité :
- Nemusíme měřit vše, ale jen to co nám pomáhá zlepšit proces a produkt/službu

Kategorie metrik

Category	Description
Cost	Cost metrics measure the amount of money invested in a project. Examples of cost metrics include effort expended or investment in training.
Productivity	Productivity metrics measure the effectiveness of the organization's infrastructure. Examples of productivity metrics include the number of reused infrastructure items and trend analysis of effort expended over given periods of time.
Quality	Quality metrics measure the effectiveness of the quality assurance efforts. Examples of quality assurance metrics include the percentage of deliverables reviewed and defect type recurrence.
Requirements	Requirements metrics measure the effectiveness of the requirements definition and validation efforts. Examples of requirements metrics include requirements instability and number of use cases.
Schedule	Schedule metrics measure the accuracy of your proposed schedule to the actual schedule. Examples of schedule metrics include the calendar time expended to perform a task or project phase.
Size	Size metrics measure, as the name suggests, the size of the development efforts. Examples of size metrics include the number of methods of a class and the function/feature point count.
Testing	Testing metrics measure the effectiveness of the testing efforts. Examples of testing metrics include the defect severity count and the defect source count.

Metriky podle fází sw procesu

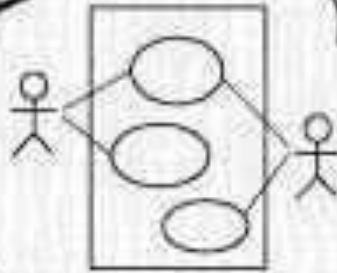
Stage	Potential Metrics
Define Requirements and Justify	Number of use case Function/feature points Level of risk Project size
Define Management Documents	Cost/benefit breakpoint Number of reused infrastructure artifacts Number of introduced infrastructure artifacts Requirements instability
Model	Procedure/Operation count of a module Number of variables per data structure Size of procedures/operations Number of data types per database Number of data relationships per database Number of interfaces per module Requirements instability
Program, Generalize and Test	Procedures/Operations size Procedures/Operations response Comments per procedure/operation Percentage of commented procedures/operations Global usage Number of candidate items for generalization Percentage of items generalized Effort required to generalize an item Percentage of deliverables reviewed Time to fix defects Defect recurrence Defect type recurrence

Metriky podle fází sw procesu

Stage	Potential Metrics
Deliver and Test	<ul style="list-style-type: none">Time to fix defectsDefect recurrenceDefect type recurrenceNumber of defectsDefect source countWork effort to fix a defectPercentage of items reworkedEnhancements implemented per releaseAmount of documentationPercentage of customers trainedAverage training time per user trainer
Assess	<ul style="list-style-type: none">Number of lessons learnedPercentage of staff members assessed
Support	<ul style="list-style-type: none">Average response timeAverage resolution timeSupport request volumeSupport backlogSupport request agingSupport engineer efficiencyReopened support requestsMean time between failuresSoftware change requests opened and closed

WHY SOFTWARE ENGINEERS CAN'T FALL ASLEEP COUNTING SHEEP

```
class Sheep {  
  private int ctr;  
  public Sheep() {  
    ...  
    ...  
  }  
}
```



```
class SheepDriver {  
  private Sheep = null;  
  public static void main  
  ...  
  ...  
}
```

```
Sheep Requirements Doc  
Abstract  
  Program to count sheep  
  ...
```

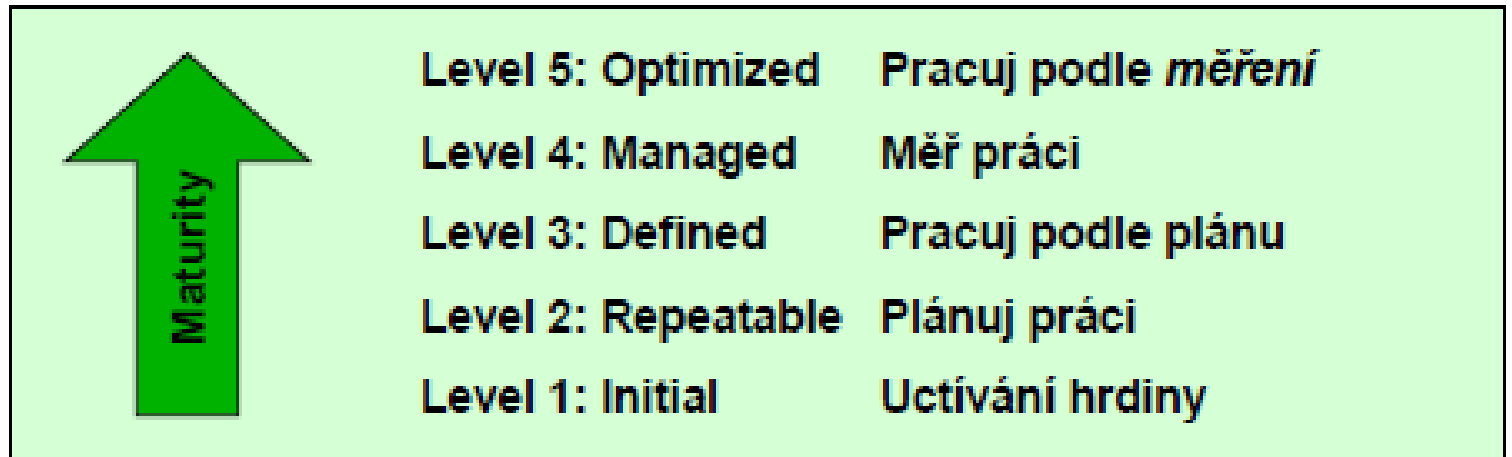


CMM

- Jak zjistíme jaké máme schopnosti dělat kvalitní projekty?
- -> pomocí CMM modelu
- CMM = CAPABILITY MATURITY MODEL (MODEL ZRALOSTI SCHOPNOSTÍ)
- Známý model v oblasti SI
- Definuje určitou úroveň firmy dělat kvalitní software

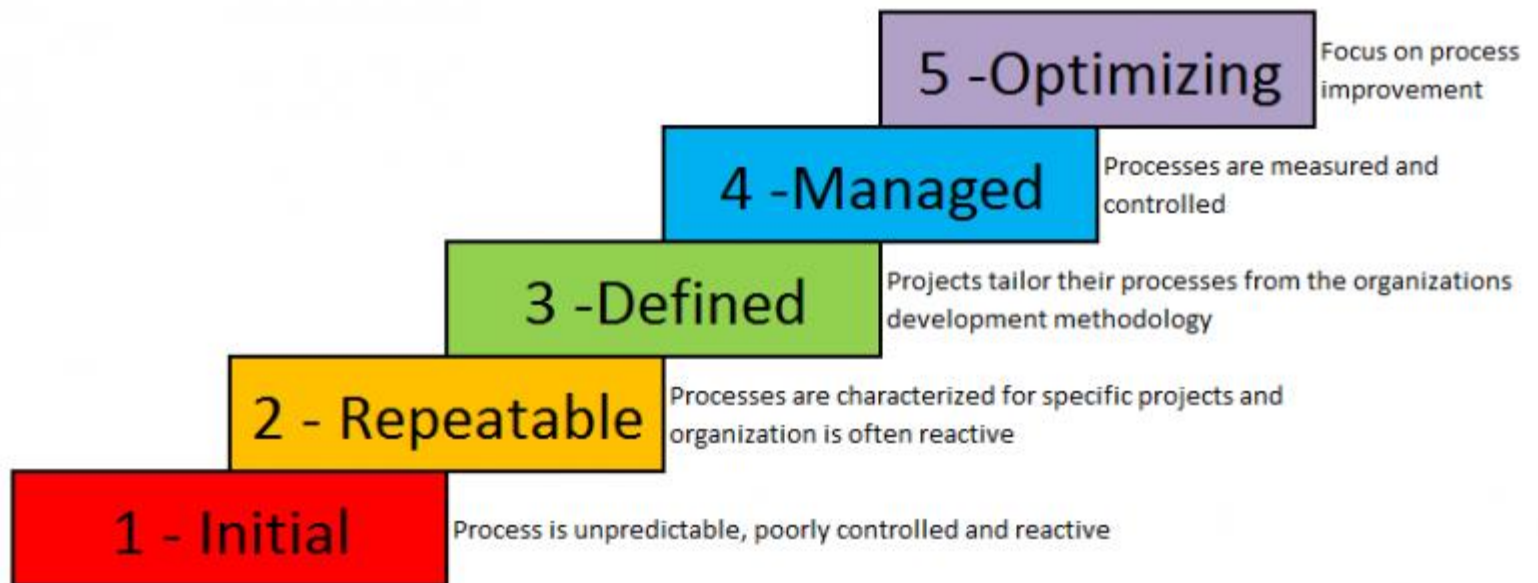
CMM

.CMM definuje 5 úrovní podle kterých poznáme v jaké fázi se nachází firma



CMM

• Jiné definice (EN verze) :



CMM podrobně

Level		Capability	Result
5 Optimizing	Continuous Process Improvement	Organizational Innovation & Deployment Causal Analysis & Resolution	Productivity & Quality
4 Quantitatively Managed	Quantitative Management	Quantitative Process Management Software Quality Management	
3 Defined	Process Standardization	Requirements Development Technical Solution Product Integration Verification Validation Organizational Process Focus Organizational Process Definition Organizational Training Integrated Product Management Risk Management Integrated Teaming Integrated Supplier Management Decision Analysis & Resolution Organizational Environment for Integration	
2 Managed	Basic Project Management	Requirements Management Project Planning Project Monitoring & Control Supplier Agreement Management Measurement & Analysis Product & Process Quality Assurance Configuration Management	
1 Initial	Heroic Efforts	Design Develop Integrate Test	
			Risk & Waste

CMM a metriky

•Potřeba metrik podle úrovní v CMM

CMM úroveň	Doporučený počet metrik
1	0 - 11
2	11 - 22
3	22 - 30
4	30 - 36
5	36 - 50

ITSM (IT Service Management)

- IT Service Management je souhrn nejlepších praxí a referenčních modelů procesů řízení služeb IT.
- Umožňuje: ekonomický růst, řešení požadavků uživatelů, inovace
- Činnosti: plánování, řízení, controlling
- ITSM se nesoustředí na detaily, ale zaměřuje se na poskytnutí rámce pro IT aktivity a vztahy mezi IT a zákazníkem

ITIL (Information Technology Infrastructure Library)

- Rámec soupisů nejlepších praktických přístupů, jak poskytovat služby IT v nejlepší kvalitě (Manažerské postupy, podpora business procesů, finanční hlediska).
- ITIL pokrývá tyto oblasti: Podnikání, Správa IT aplikací, Dodávka IT služeb, Podpora IT služeb, Správa IT infrastruktury, Řízení IT projektů
- Pro každou z oblastí existuje ve formě publikované série.
- Činnosti: plánování, řízení, controlling
- ITSM se nesoustředí na detaily, ale zaměřuje se na poskytnutí rámce pro IT aktivity a vztahy mezi IT a zákazníkem
- Existuje několik verzí – v2 (8 knih), v3 (13 knih)
- Usnadňuje certifikaci ISO 9000 (management jakosti, ISO 20000 (IT service management))

ITIL

- Stanovení cílů, vstupů, výstupů a aktivit každého procesu
- Stanovení rolí a jejich odpovědností v daném procesu
- Způsob měření kvality poskytovaných IT služeb a účinnosti ITSM
- Vzájemné vazby mezi jednotlivými procesy
- Postupy auditu a zásady reportingu pro každý proces
- Přínosy každého procesu
- Critical Success Factors, možné problémy a vhodná protipatření
- Náklady na implementaci a následný provoz
- Zásady pro řízení podpůrné ICT infrastruktury
- – Zásady bezpečnosti ICT infrastruktury

Děkuji za pozornost

Martin Tomášek

