

Requirements Engineering

Tomáš Krátký

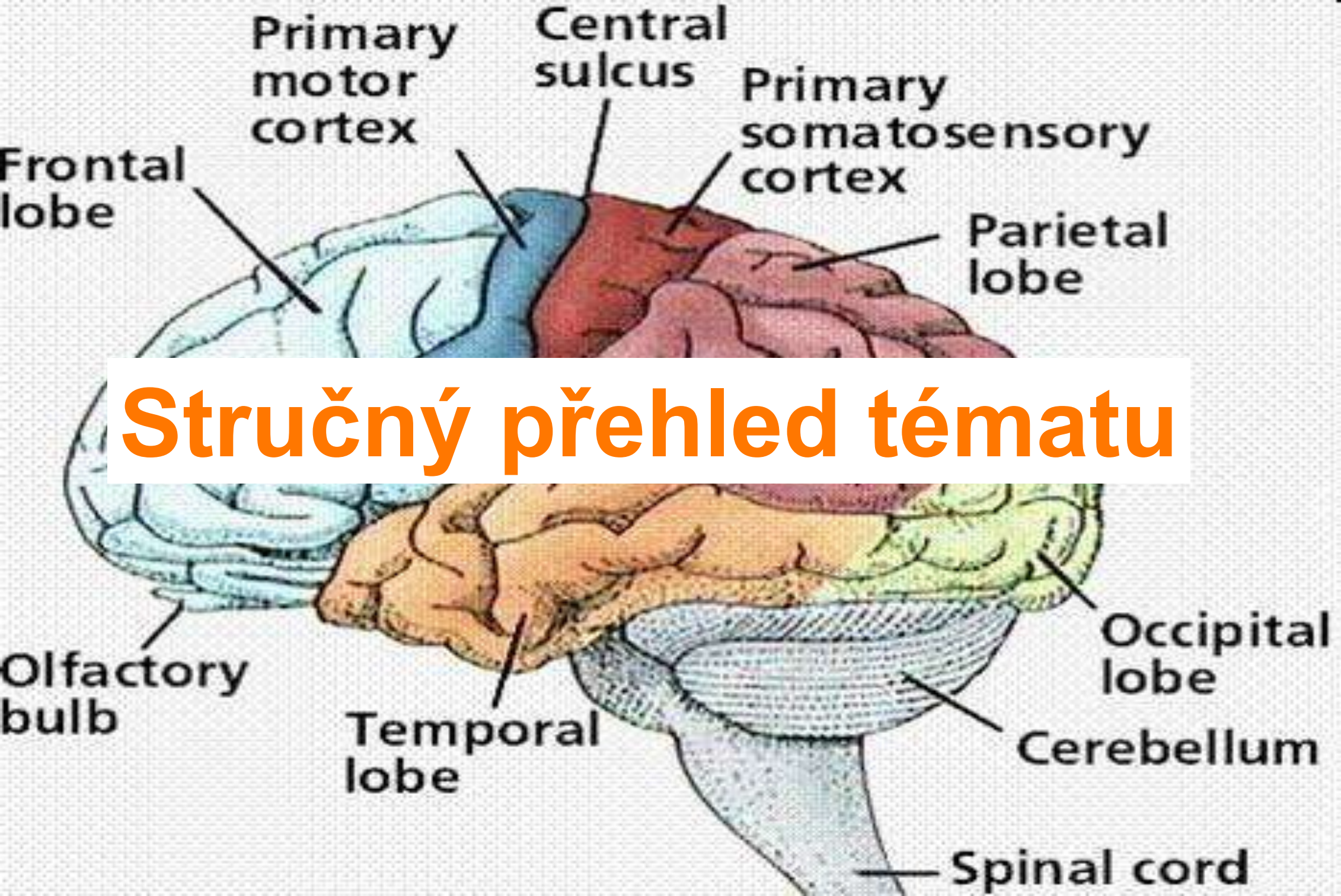
tomas.kratky@profinit.eu





Obsah

- Stručný přehled tématu
- Zásadní otázky
- Poznatky z praxe
- Slepé uličky
- Zajímavá témata
- Goodies – templates, checklists
- Doporučená literatura
- Průběžné ilustrace





Schematický pohled

(Software System) Requirements Engineering

- Elicitation
(schůzky, jednání, připomínkování dokumentů, pozorování uživatelů ...)
- Analysis
(přemýšlení, vymýšlení, debaty, poznámky, ...)
- Specification
(dekompozice, psaní, používání notace)
- Verification
(čtení textu, schůzky, jednání, promítání GUI, ... velké bitvy o rozsah ...)

... to vše i několikrát, promixované v čase, lidech, zaměření ...



Rozsah



Všechny závazky !!!

- metodologie, integrace s jinými systémy, ...
- zákazník potenciálně chce vše co není explicitně "NE"
- dodavatel většinou dělá jen to co je explicitě "ANO,,
- pozor na šedou zónu



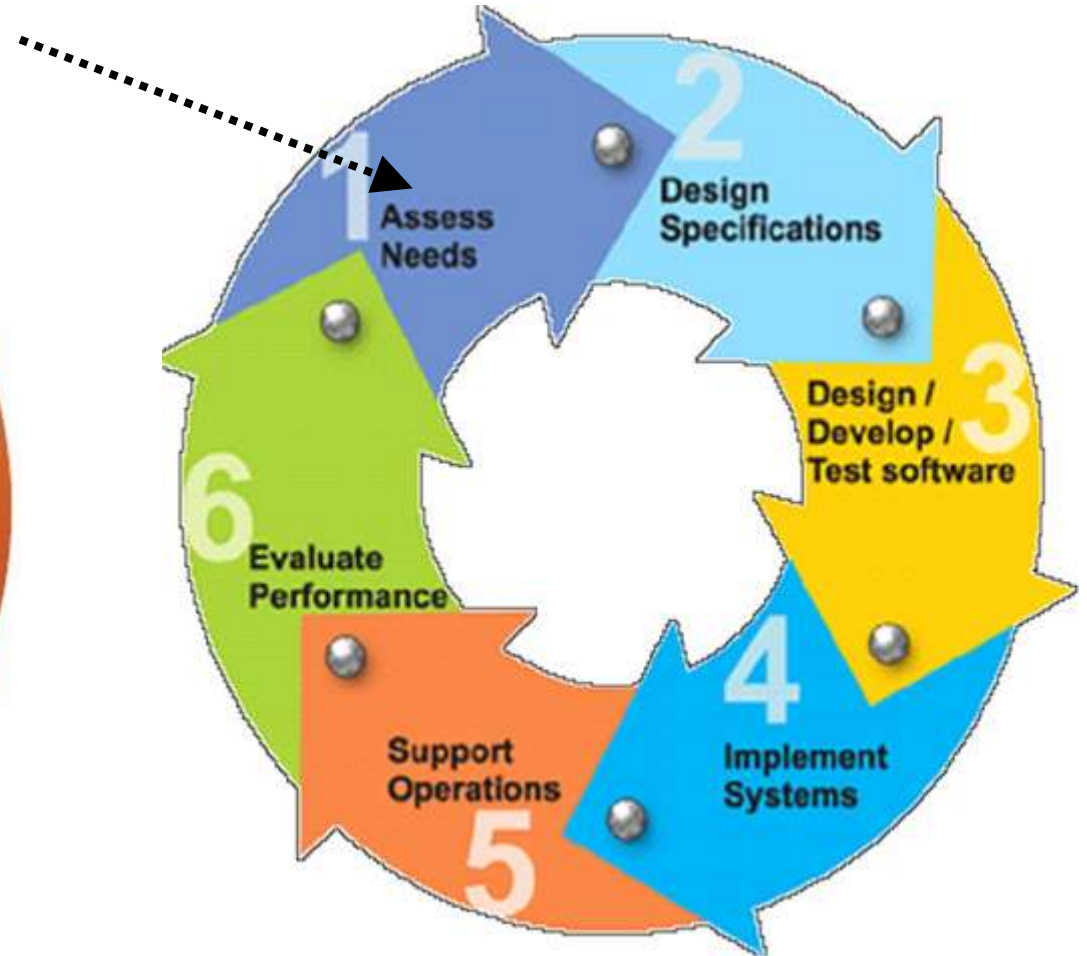
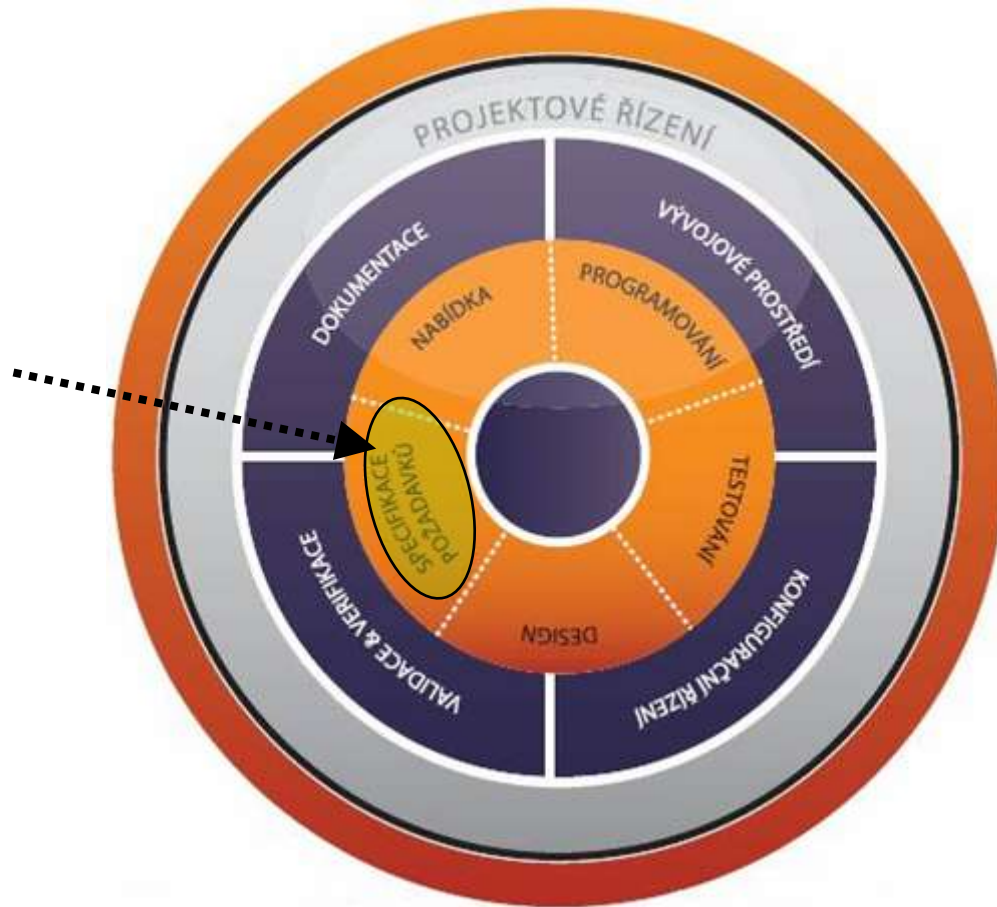
Typy požadavků

- požadavky na vlastní funkce
- požadavky na rozhraní
 - uživatelské, softwarové, HW, komunikační, ...
- nefunkční požadavky
 - výkon, bezpečnost, spolehlivost, ...
- další požadavky
 - legislativní, vícejazyčnost, ...

→ viz šablona pro specifikaci software

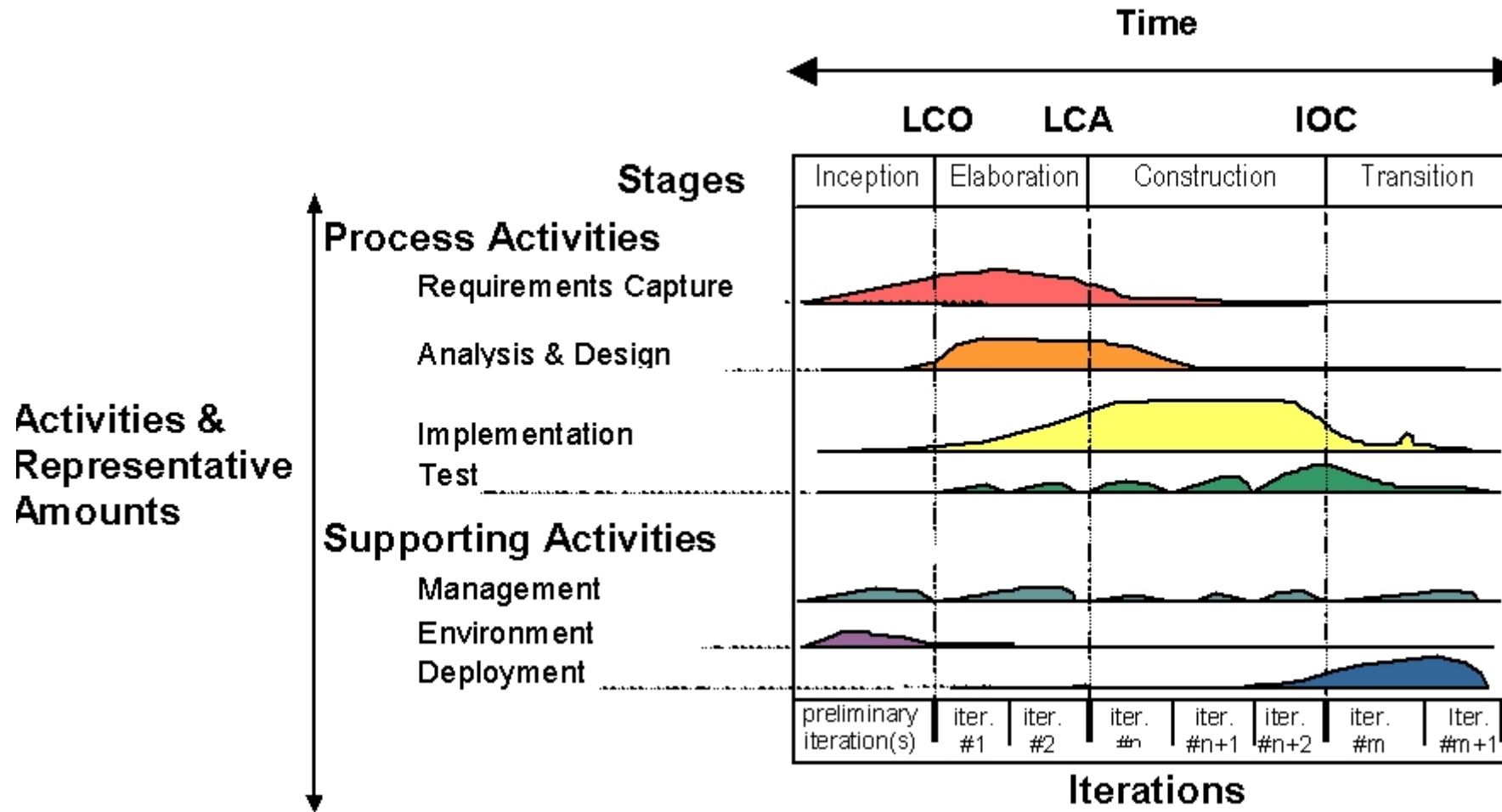


Softwarový proces



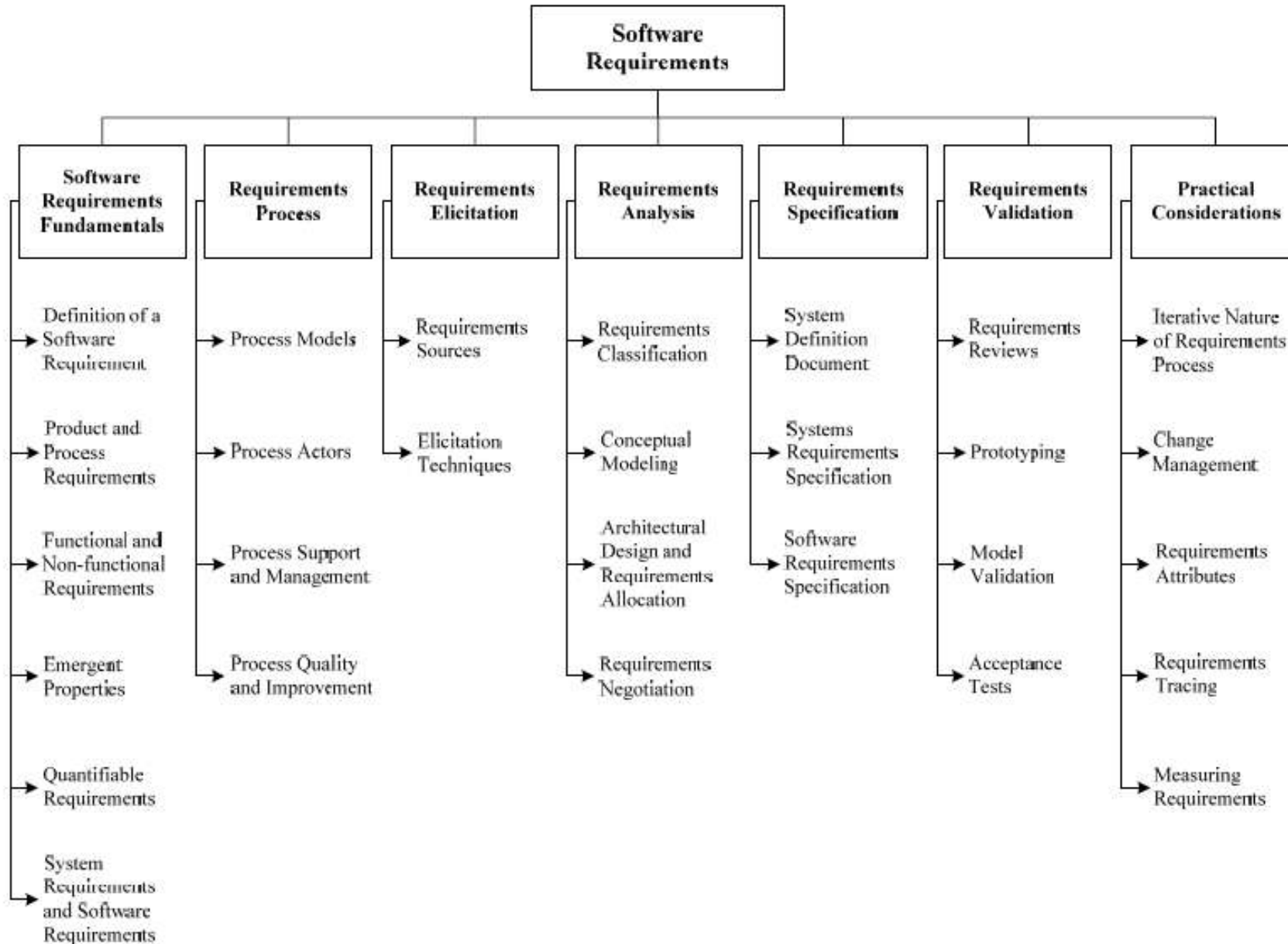


Softwarový proces





SWEBOK





Zásadní otázky



Zásadní otázky – I.

- Co má být výsledek analýzy ?
- Obsah ?
- Forma ?
- Logická a fyzická dekompozice ?
- Míra detailu ?
- Pracnost ?
- Kalendářní čas ?
- Počet lidí ?



Zásadní otázky – II.

- Jak mezi paralelně pracující lidi dekomponovat práci (de facto už ovlivňují výsledek analýzy její předčasnou dekompozicí) ?
- Kdy mohu už začít navrhovat architekturu ?
- Kdy mohu už začít konstruovat ?
- Jak má být analýza/specifikace rozprostřena v čase od psaní nabídky až po údržbu systému?
- Jaké jsou rozdíly mezi specifikací z nuly vs. specifikací změnových řízení během údržby systému



Zásadní otázky – III.

- Jaký je vlastně vztah mezi specifikací a architekturou (co vs. jak) ?
- Jak poznám, že na straně zákazníka se mi věnují ti správní lidé ?
- Jaké vlastnosti má mít dobrý analytik ?
- Jak ověřím, že specifikace je specifikace toho co se skutečně chce, resp. potřebuje ?
- Je rozumné bát se zeptat ?
- Je rozumné nechat si schválit něco o čem mám sám vnitřní pochyby?



Zásadní otázky – IV.

- Jak poznám, že specifikace mi slouží a k čemu vlastně ?
- Co s analytiky po analýze ?
- Jak udržovat výstupy analýzy ?
- Jak se liší analýza při vývoji a při údržbě ?
- Lze vynechat analýzu ?
- Jak rozeznat ty správné okrajové podmínky ?
- Jak se budou (typicky) měnit požadavky a jak se na to připravit ?



Zásadní otázky – V.

- Jak detekovat nárůst rozsahu ?
- Fenomén „*gold-plating*“ ?

... a mnoho dalších ...



Poznatky z praxe

- pracnost: 10 - 30%
- pracnost při údržbě: 1/5
- rozložení v čase je podle "knih"
- osobní předpoklady – ne ghetto analytiků
- forma nesmí zastínit obsah
- obsah musí být „komplexní“
- rozsah musí být analýzou dost popsán
- udržovatelnost je zásadní věc
- čistý princip
- SDLC hraje roli
- rozložení výdajů pracnosti
- nutnost mít odvahu se poučit



Poznátky z praxe

- model GUI a strukturovaný text funguje
- template a checklist funguje
- všechny typy požadavků jsou třeba
- naměřená data jsou třeba (obrazovky, ...)
- život si prosadí, co je skutečně třeba (otázka jak bolestně)
- co se změnou požadavků
 - změna nebo odložené pochopení
 - změna nebo větší míra detailu
 - pozitivní a negativní vymezení
 - boj o rozsah



Slepé uličky



Slepé uličky

- Nedělat analýzu
 - Nemyslet na architekturu
 - Dělat pouze katalog či use cases
 - Ignorovat jiné než funkční požadavky
 - Nepoznat, kdo je skutečně důležitý stakeholder
 - Nevěřit vlastní rozvaze a intuici
 - Neptat se na věci, které „nechci“ slyšet
- ... a mnoho dalších ...



Zajímavá témata





Textová specifikace

- Viktoriánský
- Strukturovaný (viz SVZ)
- Izolovaný (katalog)





Dekompozice požadavků

Kolem čeho ?

- Data – centric
 - Function – centric
 - **Feature – centric** (viz datové schránky)
 - Use case – centric
 - Aspect – centric
 - **Architecture (imposed/proposed) – centric**
 - **Change request – centric** (viz IS VZ US)
- Do LCA
- Při vývoji
- Po loC (údržba)

Jak ?

- UML – centric
- Text – centric



Model GUI

- Funguje skoro cokoli !!!!
 - PowerPoint (viz eSIPO)
 - Excel (viz datové schránky)
 - HTML (potenciálně základ GUI systému)
 - Speciální jazyky pro tvorbu modelu GUI
- Ne vždy je třeba, ne vždy dává smysl
- Pomáhá pochopení systému



UML

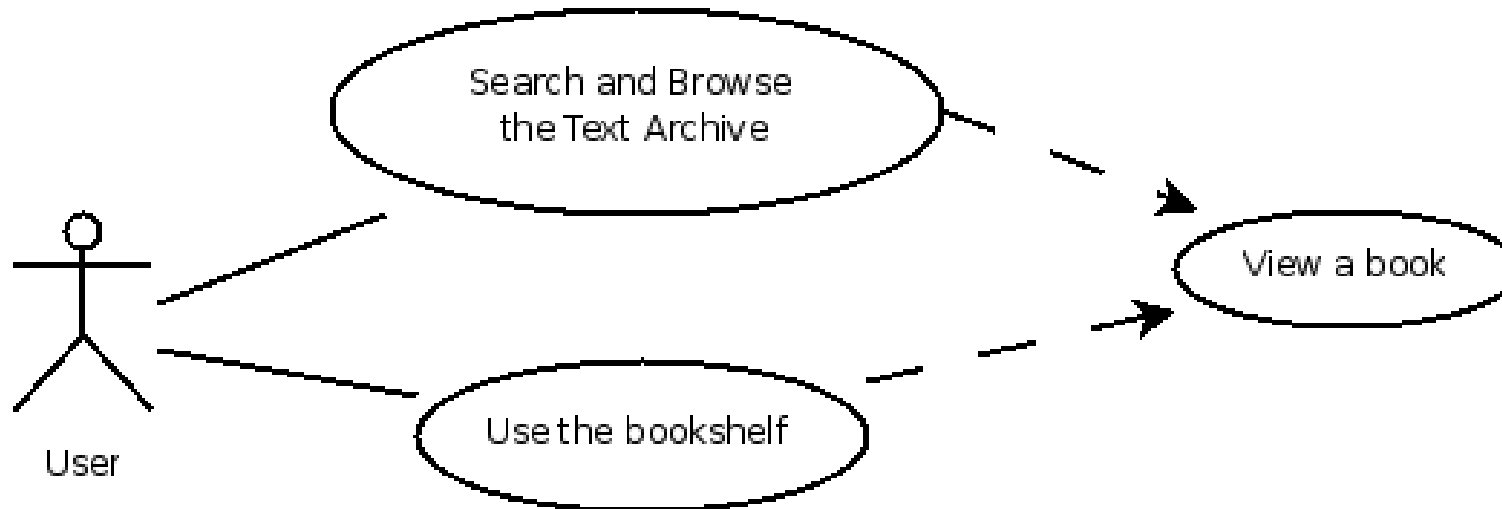
- Prostředek pro reprezentaci vyvíjeného SW
 - na úrovni analýzy,
 - návrhu a
 - částečně i realizace
- Nutné znát (problém u zákazníka)
- Nemá smysl vymýšlet něco jiného
- Ne vždy je použitelný





Use case

- Scénáře
- Dobré jako doplněk (viz datové schránky)
- Nelze použít jako základ pro specifikaci
- Někdy jen útěk před složitostí !





Goodies





Templates, checklists, literatura

Podpora univerzit

Univerzitní výuka

Materiály SwEng

Předmět SW1129

Předmět 1-INF-312

Předmět A4M33SEP

Předmět 1-INF-314

Předmět PDBT

Invited lectures

Odborné konference

ACM SPY



REQUIREMENTS MATERIÁLY

Články

- [When Telepathy Won't Do: Requirements Engineering Key Practices](#)
- [Karl Wiegers Describes 10 Requirements Traps to Avoid](#)
- [Writing Effective Natural Language Requirements Specifications](#)
- [Be Careful With "Use Cases"](#)

Knihy

- Wieger K. Software Requirements: Practical techniques for gathering and managing requirements throughout the product development cycle. Microsoft Press, 1999, resp. 2nd ed.
- [Just Enough Structured Analysis](#) - klasická kniha Ed Yourdona ve wiki formátu !

Checklists

- [CxCheck Requirementst.txt](#) - checklist firmy Construx pro oblast Requirements
- [Requirements review checklist.doc](#)
- [Use case checklist.doc](#)
- [Impact analysis checklist.doc](#)

Templates

- [srs_template.doc](#) - velmi dobrý template pro tvorbu specifikace požadavků
- [use_case_template.doc](#)
- [vision and scope template.doc](#)
- [SAFE BusinessRequirements.doc](#)
- [SAFE SystemRequirements.doc](#)

Všechny odkazované materiály jsou poskytnuty výhradně za účelem výuky softwarového inženýrství.
© Of Respective Parties 2007-2009



Ilustrace - souhrn

SVZ

- Jednoduchá strukturovaná specifikace

IS VZ US

- Dekompozice dle change requests
- Specifikace změnového řízení 4907

eSIPO

- PowerPoint model GUI

Datové schránky

- Funkční uživatelská specifikace, využití Use cases
- XLS model GUI

Diskuse

