

Iterativní vývoj software

KIV/ASWI 2017/2018

► Obsah

► Iterativní vývoj

- struktura a vlastnosti iterace
- globální řízení

► Empirický proces

Q: Jaké můžeme v nejbližší době čekat nové, vzrušující a slibné myšlenky nebo techniky v oblasti software?

A: Myslím, že [nejslibnější myšlenky] jsou už léta známy, jen nejsou správně používány.

– David Parnas, 1999

Kořeny iterativního přístupu ...

Fraser: (from The nature of progress in software production)

»Design and implementation proceeded in a number of stages. Each stage was typified by a period of intellectual activity followed by a period of program reconstruction. Each stage produced a useable product and the period between the end of one stage and the start of the next provided the operational experience upon which the next design was based. In general the products of successive stages approached the final design requirement; each stage included more facilities than the last. On three occasions major design changes were made but for the most part the changes were localised and could be described as ‘tuning’.

Report on a conference sponsored by the NATO SCIENCE COMMITTEE Garmisch, Germany, 7th to 11th October 1968

<http://arialdomartini.wordpress.com/2012/07/20/you-wont-believe-how-old-tdd-is/>

M.Bolton: (...) *Computers don't break down as they used to, so what's the motivation for unit testing and test-first programming today?*

G.Weinberg: We didn't call those things by those names back then, but if you look at my first book (Computer Programming Fundamentals, Leeds & Weinberg, first edition 1961 —MB) and many others since, you'll see that was always the way we thought was the only logical way to do things. I learned it from Bernie Dimsdale, who learned it from von Neumann.

► Jak funguje iterativní vývoj

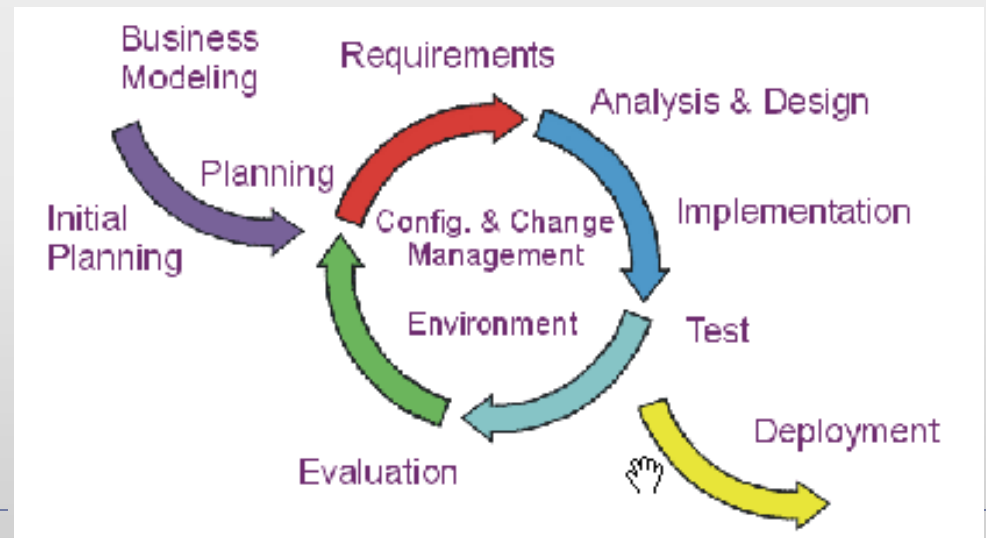
- „Když sekvenční postup funguje pro malé projekty s malou mírou neznáma, proč nerozbít velký projekt do řady malých?“ — P. Kruchten
- Miniaturní úplný projekt
 - cca vodopádový model
 - prolínání aktivit
- Cíl: iterační release (interní či nasazený)
 - produkt funkčně neúplný
 - ale otestovaný a funkční
- Opakovaný postup
 - stále stejné aktivity (téměř)





► Průběh iterace

- Plánování cíle iterace
 - zejména funkčnost
- Doplnění / zpřesnění požadavků
 - základ: plán projektu, vize, předchozí feedback
- (Úprava návrhu)
- Implementace přírůstku funkčností
- Integrace přírůstku
 - ověření, otestování
- (Předání do provozu)
 - validace zákazníkem
- Zhodnocení



► Počet a délka iterací



► Počet

- charakter projektu (rozsah, velikost týmu)
- fáze vývoje
- obvykle alespoň 3 celkem

► Délka

- malá je lepší – menší složitost/riziko, rychlá adaptace
- 1-4 týdny pro malé, 3-6 týdnů velké projekty, zřídka měsíce
- produktivita: blízký cíl, vysoký výkon (až 80 vs 25 FP/měs)
- psychologie: lidé si pamatují překročené termíny, ne opuštěné vlastnosti; nutí včas k těžkým rozhodnutím a kompromisům

SCRUM: 30 dní

XP: 1-2 týdny

► Pravidla pro iterace

- Běžící iterace **uzavřená změnám** zvenčí
 - nutné pro stabilitu projektu
 - potřebuje dobré změnové a projektové řízení
 - zdroje tlaku na změnu: čas, funkčnost, postup
- Vždy pevné datum ukončení
 - plánováno nejpozději na začátku iterace
- **Timeboxované** iterace = délka známa předem
 - omezení plánované funkčnosti možné
 - nelze: nehotový release, změna datumu, přesčasy

► Předání a zhodnocení iterace

► „Customer Demo“

- Předvedení/předání výsledku zákazníkovi
- interní vs externí release
- akceptace (nebo také ne)

„ve čtvrtek“
„no slideware“
max 2h

► Retrospektiva

- co se dařilo = co zachovat
- co se nedařilo a proč => co změnit
- jak můžeme být příště lepší

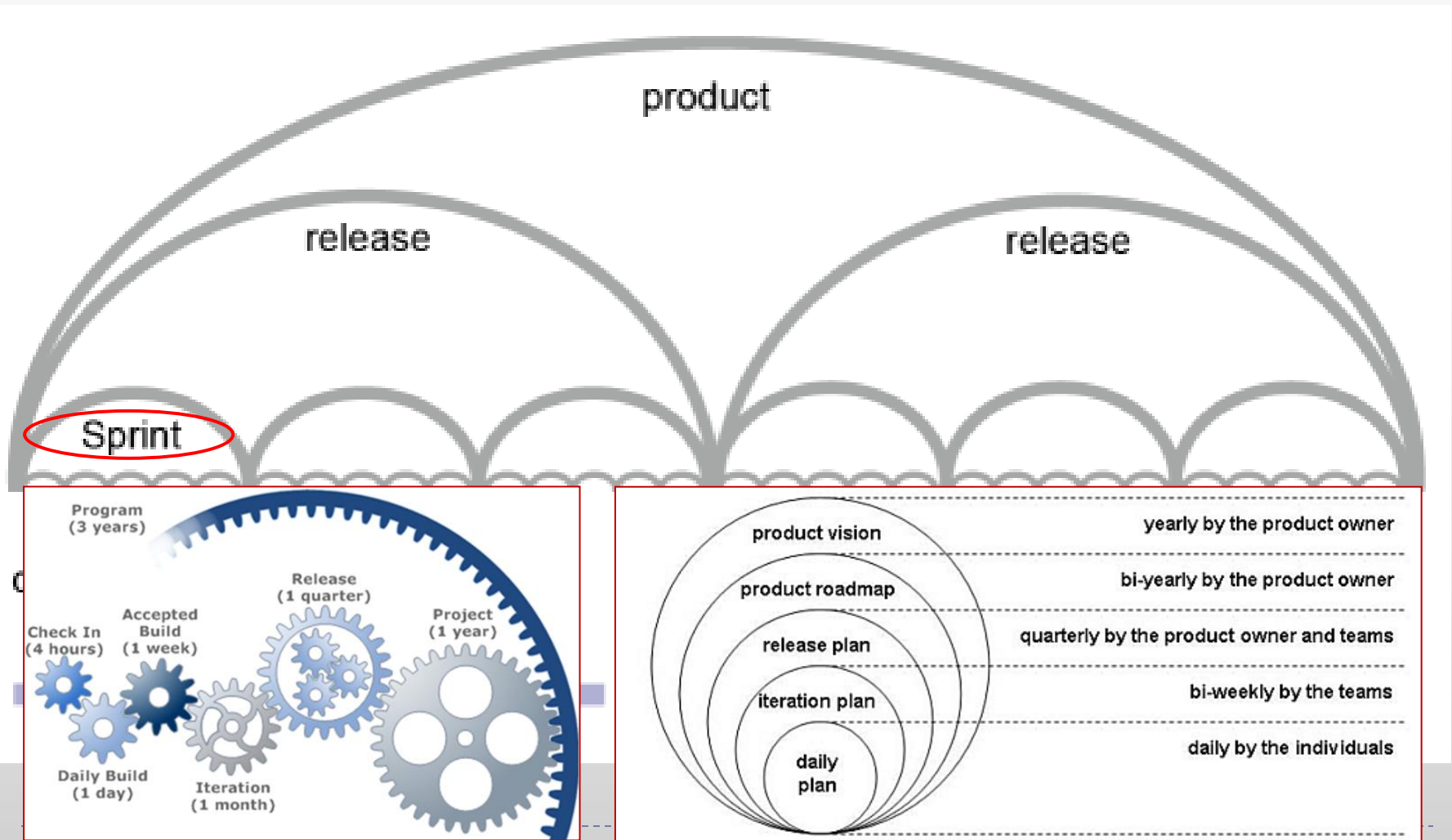
pouze pro tým
max 1h
záznam

Úkoly							
<div> <div>Filtr</div> <div> <input checked="" type="checkbox"/> Vše <div> <input type="checkbox"/> Cílové vzorce <div> AWM 2017 - MM-M-4-tvarova <div></div> </div> </div> </div> <div>Přidat filtr</div> <div>Nastavení</div> </div>							
<div> <div>Prohl.</div> <div>Smažt</div> <div>Ukládt</div> </div>							
#	Fronta	Priorita	Předmět	Přilazeno	Uzavřít do	Odhadovaná doba	
<input type="checkbox"/>	1279	Task	Urgent	Převod 3D modelů z .3ds do .xaml	Jan Řeřicha	2012-05-08	5,00
<input type="checkbox"/>	1262	Support	High	Šchizka týmu	Jan Řeřicha	2012-05-08	4,00
<input type="checkbox"/>	1261	Task	High	Konferenční řešení s Wi-Fi na základě konzultace	Jan Řeřicha	2012-05-03	4,00
<input type="checkbox"/>	1225	Support	High	Šchizka s mentorem - kancelář	Jan Řeřicha	2012-05-02	0,50
<input type="checkbox"/>	1224	Task	High	Přidání odkazu "Get Person" pro pohyb 3D scénou	Lukáš Volf	2012-05-06	20,00
<input type="checkbox"/>	1193	Support	High	Šchizka týmu	Jan Řeřicha	2012-04-27	1,50
<input type="checkbox"/>	1048	Enhancement	High	Uložení sdružení gest do souboru	Jan Řeřicha	2012-04-11	2,00
<input type="checkbox"/>	979	Task	High	Testování prototypu	Petr Altman	2012-05-07	2,00
<input type="checkbox"/>	1278	Task	Normal	Dynamické centrování vstupu detekce uživatelských akcí	Jaroslav Kříž	2012-05-08	2,00
<input type="checkbox"/>	1277	Task	Normal	Vytvoření prototypu 3D aplikace	Petr Altman	2012-05-08	10,00
<input type="checkbox"/>	1265	Task	Normal	Vytvoření rozhraní pro záznam skutečností o Houtkovsku informací	Petr Altman	2012-05-06	5,00
<input type="checkbox"/>	1264	Task	Normal	Vyhledání 3D modelů	Jan Řeřicha	2012-05-04	0,50
<input type="checkbox"/>	1205	Enhancement	Normal	Implementace algoritmu pro vyhodnocení akce zavření diány	Petr Altman	2012-05-08	15,00
<input type="checkbox"/>	1135	Support	Normal	KONICS - schizka	Jan Řeřicha	2012-05-09	2,00
<input type="checkbox"/>	1134	Support	Normal	Šchizka týmu	Lukáš Volf	2012-04-20	2,00
<input type="checkbox"/>	1130	Enhancement	Normal	Implementace Maslovoe odvádění aplikace včetně možnosti volby natrénovaného profilu	Lukáš Volf	2012-04-17	15,00
<input type="checkbox"/>	1072	Task	Normal	Vizuální zobrazení trackballu	Lukáš Volf	2012-04-17	10,00
<input type="checkbox"/>	1071	Task	Normal	Prozkoumat možnost rozpoznávání gest ve 3D	Lukáš Volf	2012-04-30	10,00
<input type="checkbox"/>	1062	Task	Normal	Sepsat dokument o výsledcích výzkumu rozpoznávání prstů	Petr Altman	2012-04-17	10,00
<input type="checkbox"/>	1028	Task	Normal	Vyzkoumat možnost rozpoznávání rukou a prstů	Petr Altman	2012-04-06	10,00
<input type="checkbox"/>	1024	Support	Normal	KONICS - schizka	Jan Řeřicha	2012-04-17	1,50
<input type="checkbox"/>	977	Enhancement	Normal	Autentifikace uživatelů zvednutím ruky.	Lukáš Volf	2012-05-02	21,00
<input type="checkbox"/>	976	Enhancement	Normal	Rozpoznání gest.	Lukáš Volf	2012-04-17	25,00
<input type="checkbox"/>	975	Enhancement	Normal	Implementace Maslovoe vyhodnání akcí.	Michal Karfík	2012-05-01	22,00
<input type="checkbox"/>	973	Enhancement	Normal	Návrh a implementace prvku - seznam.	Petr Altman	2012-04-17	10,00
<input type="checkbox"/>	971	Enhancement	Normal	Návrh a implementace prvku - seznam.	Petr Altman	2012-04-17	10,00
<input type="checkbox"/>	970	Enhancement	Normal	Návrh a implementace prvku - tlačítko.	Petr Altman	2012-04-17	10,00
<input type="checkbox"/>	1223	Task	Low	Dokumentace ke Kinect Controls	Petr Altman	2012-04-23	2,00
<input type="checkbox"/>	1222	Task	Low	Dokumentace ke Kinect Hand Tracking	Petr Altman	2012-04-21	2,00
<input type="checkbox"/>	1221	Task	Low	Dokumentace ke Gesture Recognizeru.	Lukáš Volf	2012-04-24	2,00
<input type="checkbox"/>	1220	Task	Low	Využití timeru pro odvádění textového vstupu.	Jan Řeřicha	2012-05-03	0,50
<input type="checkbox"/>	1133	Support	Low	Schizka týmu	Jan Řeřicha	2012-04-13	1,00
<input type="checkbox"/>	974	Task	Low	Návrh a implementace prvku - testový vstup	Jan Řeřicha	2012-05-18	15,00



Iterace v kontextu

► Kontext iterace v procesu vývoje

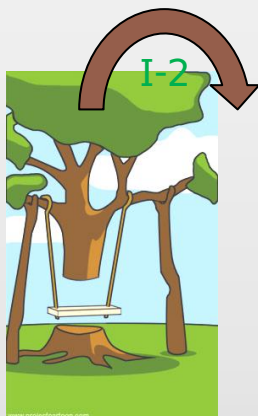


► Globální řízení iterativního vývoje

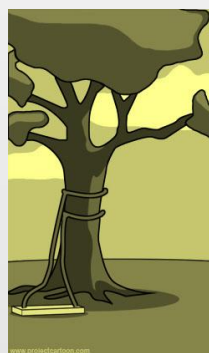
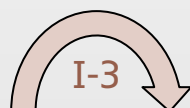
Problém: pro stromy nevidím les



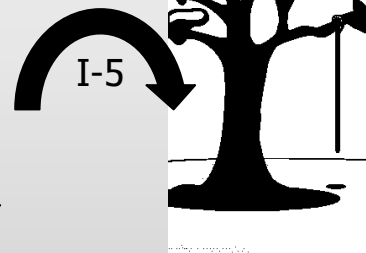
Jak vysvětleno
zákazníkem



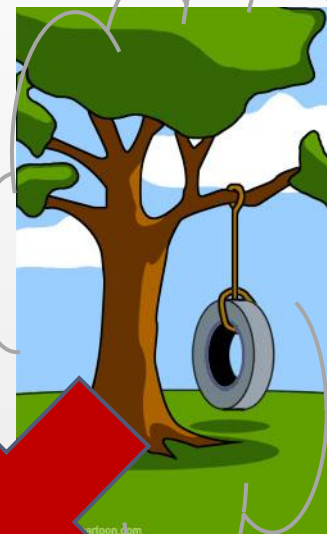
Dobrý nápad
(úprava plánu)
č. 1



Dobrý nápad
č. N



Dodáno na
konci



Co bylo
skutečně
potřeba



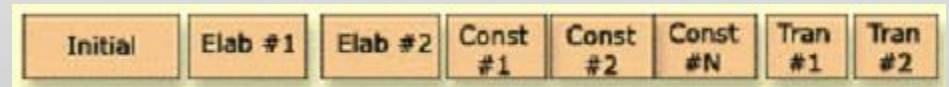
► Globální řízení: milníky => fáze

- Cíl: eliminovat momentálně největší riziko
- **LCO** (Lifecycle Objectives) – *zahájení*
 - definování terče – Vize produktu
- **LCA** (Lifecycle Architecture) – *projektování*
 - určení způsobu řešení – Architektura technického řešení
 - ověření – modely, technické prototypy, testy (executable)
- **IOC** (Initial Operational Capability) – *konstrukce*
 - schopnost efektivně „vyrobit“ řešení – beta verze, all features
 - unit a funkční testy
- **GA** (General Availability) – *nasazení*
 - uvést produkt do rutinního provozu = „krabice“ s produktem, website launch, tisková zpráva a raut :-)
 - support team v provozu

Barry Boehm (1996):
Anchoring the Software Process

► Fáze vývoje a charakter iterací

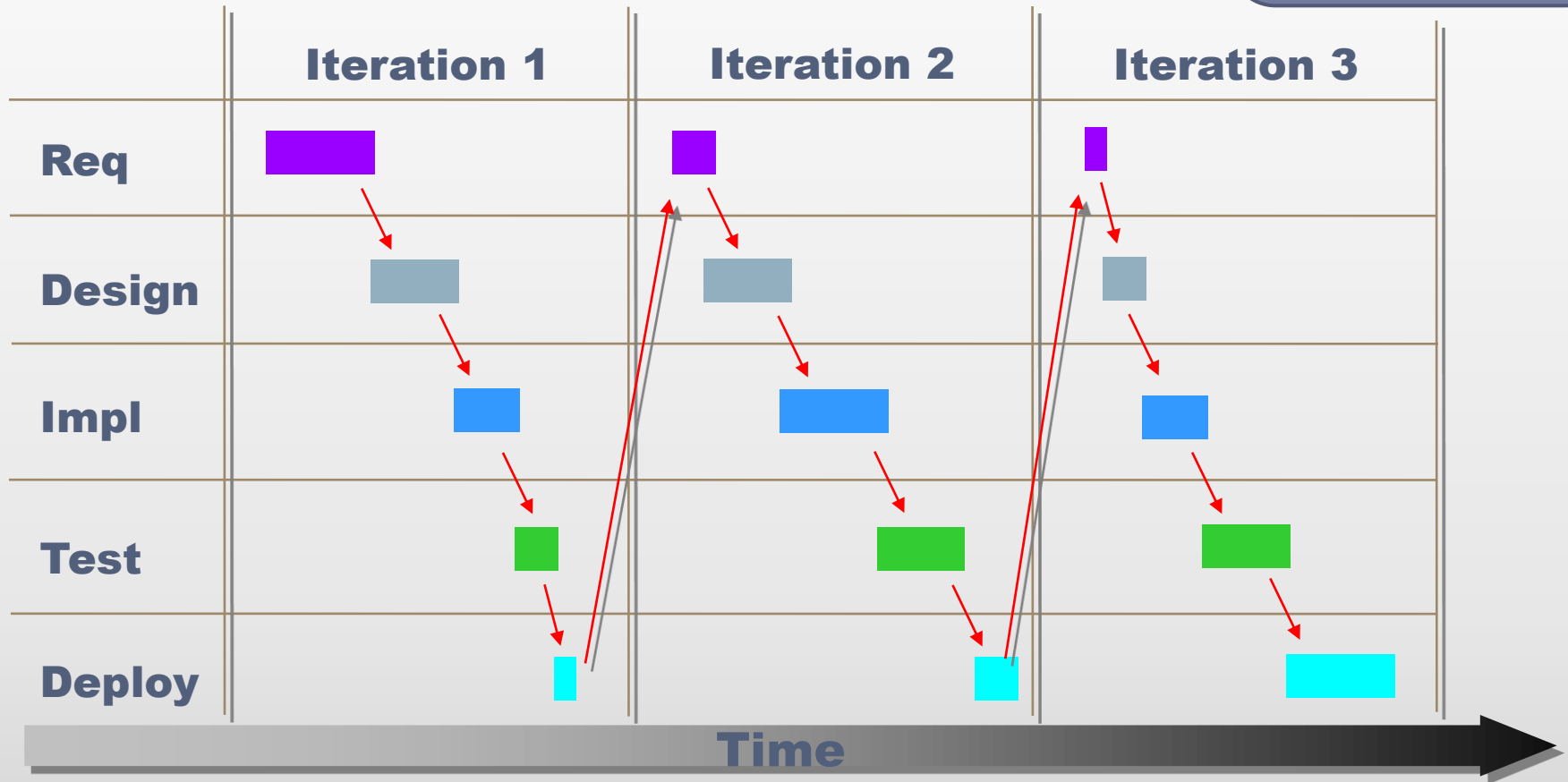
- Základní schema pevné, mění se činnosti a artefakty
- **Zahájení** – analytické činnosti, validace vize zákazníkem
 - 1-2 iterace
- **Projektování** – analytické a designérské činnosti, ověřování prototypy, implementace
 - 2+ iterací
- **Konstrukce** – designérské a programátorské činnosti, změnové řízení, testování a ověřování
 - N iterací,
- **Nasazení** – integrační a konzultační činnosti, ověřování provozem, náběh uživatelské podpory
 - 1-2 iterace



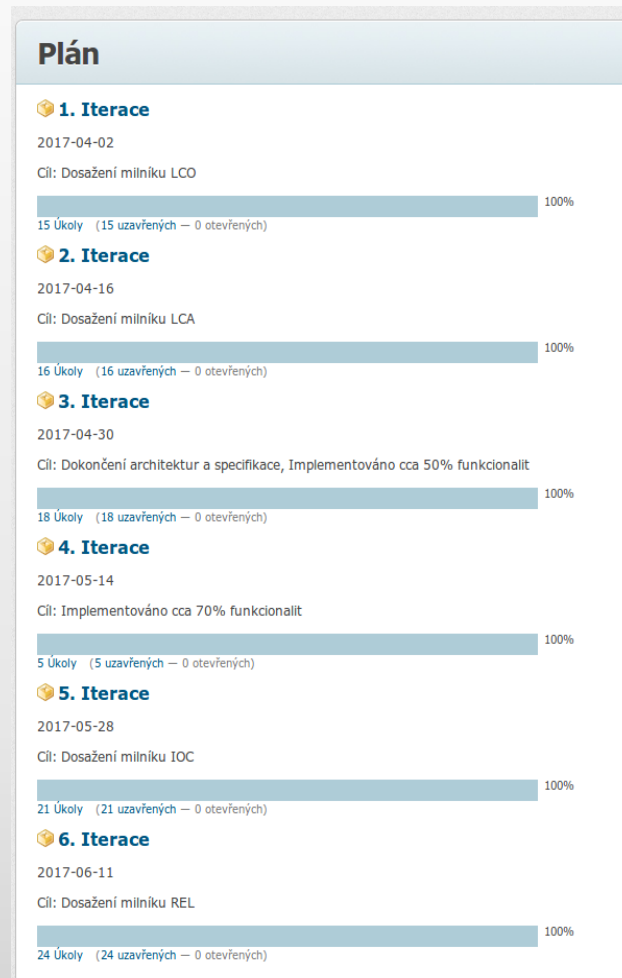
► Podíl činností dle fáze

Simple way:
per-iteration-goal

Better way:
per-workitem



► Příklad průběhu projektu v ALM



Charakteristiky iterativního vývoje

► Přírůstkový vývoj produktu



Vize produktu ...

... a její iterativní naplňování



► Evoluční a adaptivní charakter

► Evoluční

... jeden z 4 nejčastějších faktorů úspěchu sw projektů

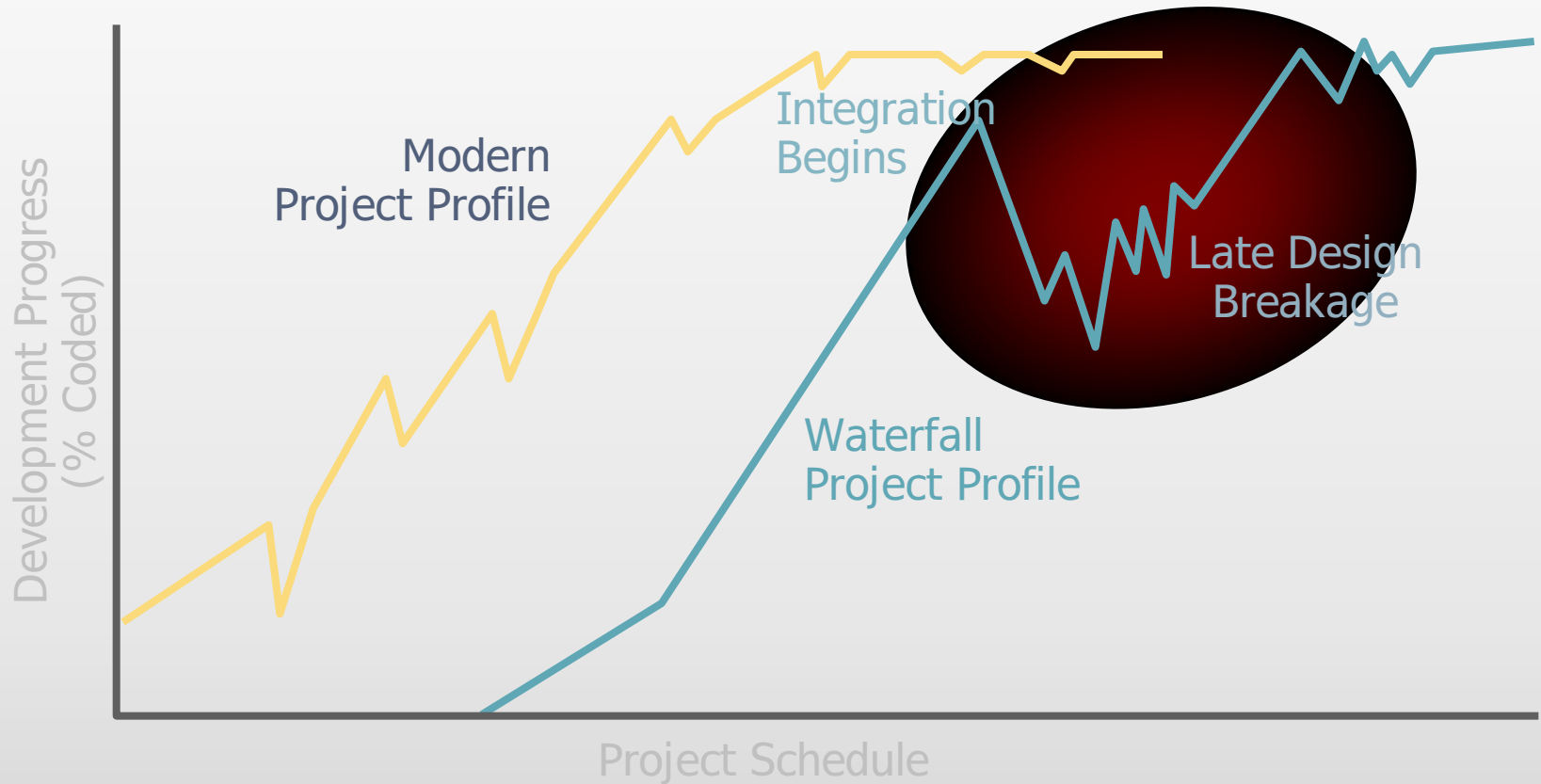
- **znalosti** o požadavcích, návrhu, odhadech a plánu se vyvíjejí a **zpřesňují v průběhu** projektu
 - vs kompletní, dále neměnné specifikace na začátku (20-80)
 - míra změny obvykle klesá s postupujícími iteracemi
- „don't develop software, grow it”

► Adaptivní

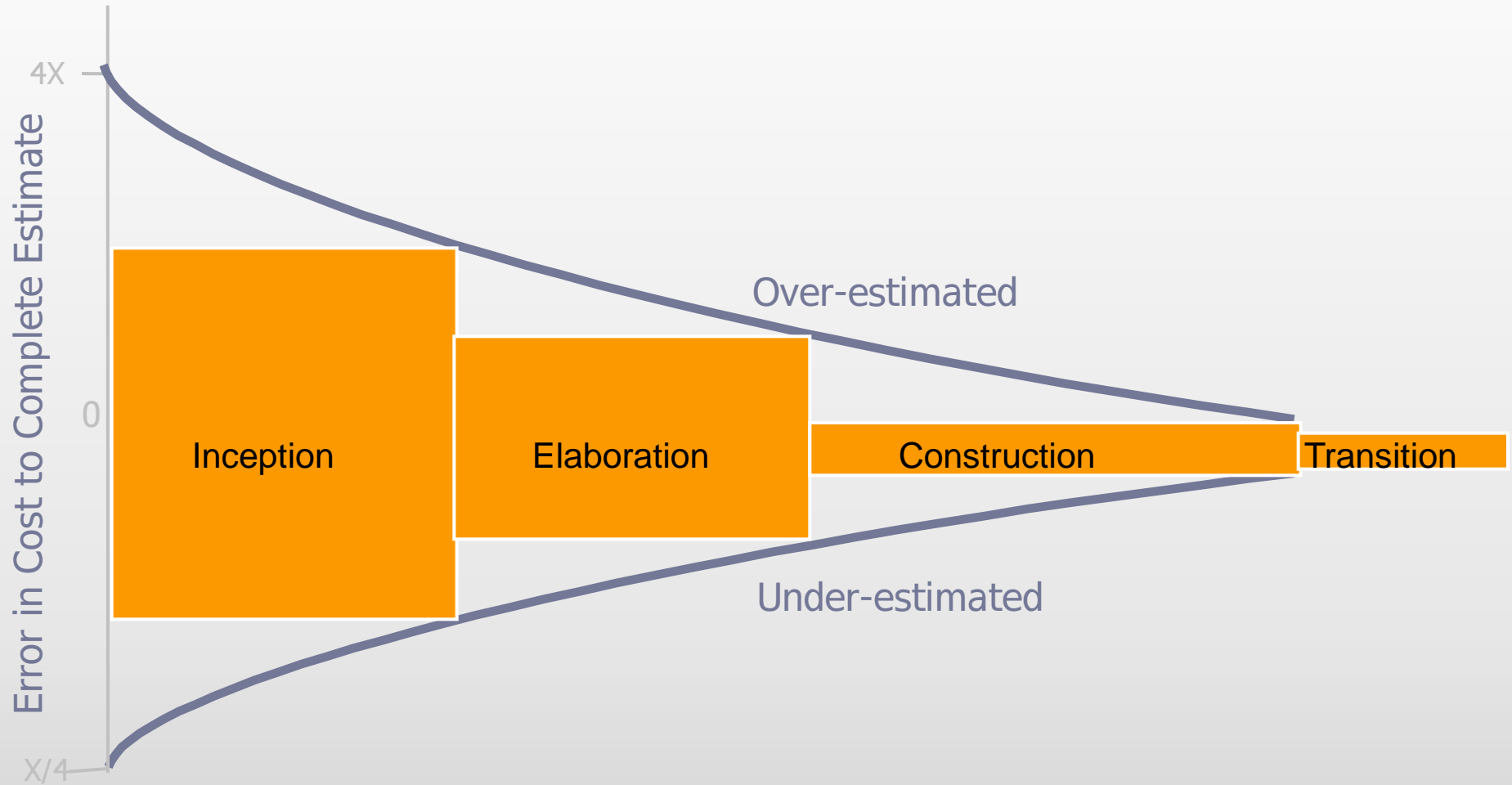
- definice produktu i **postup vývoje reaguje na zjištěné poznatky**
- zdůraznění procesu učení



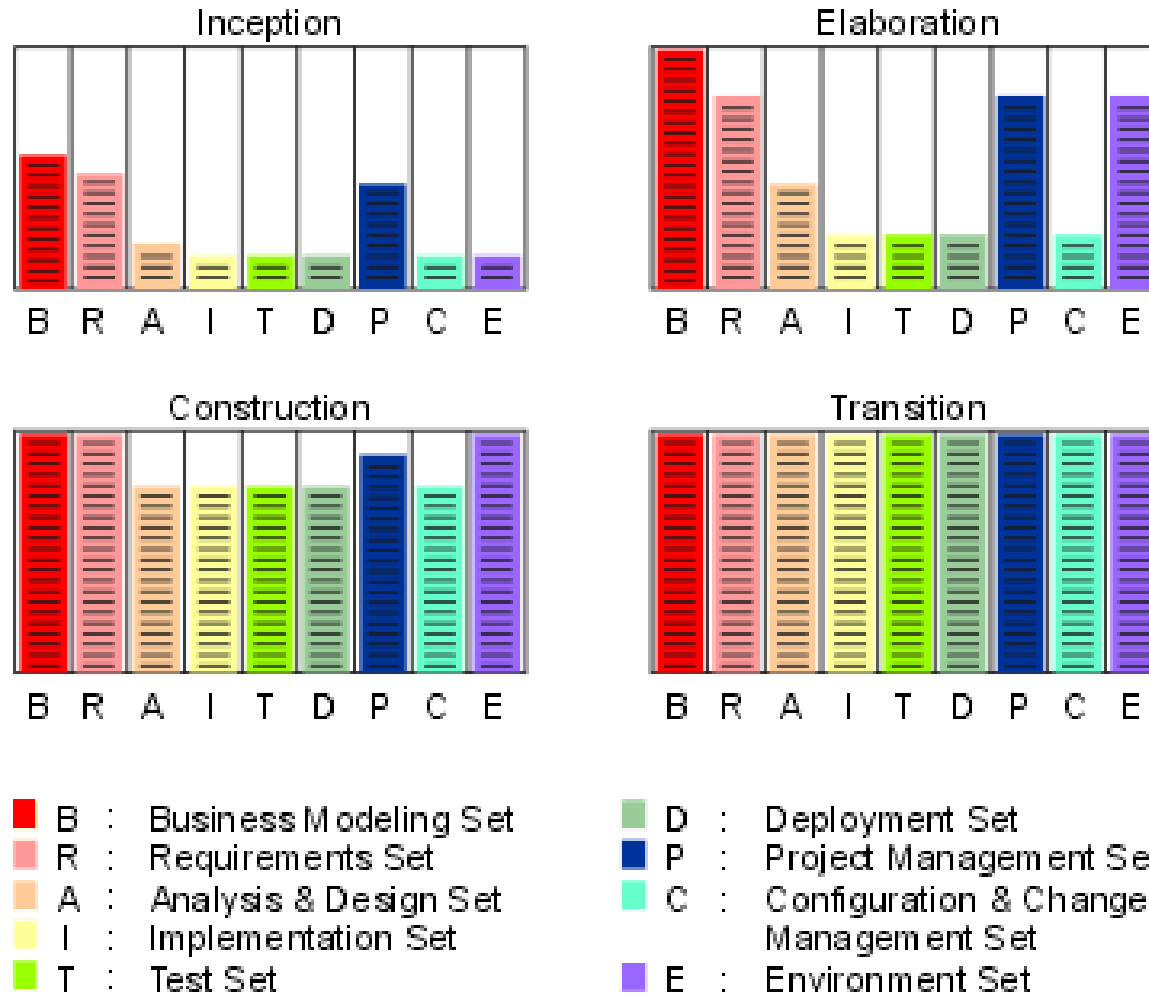
► Better Progress Profile



► Cost Estimate Fidelity



► Vývoj meziproduktů

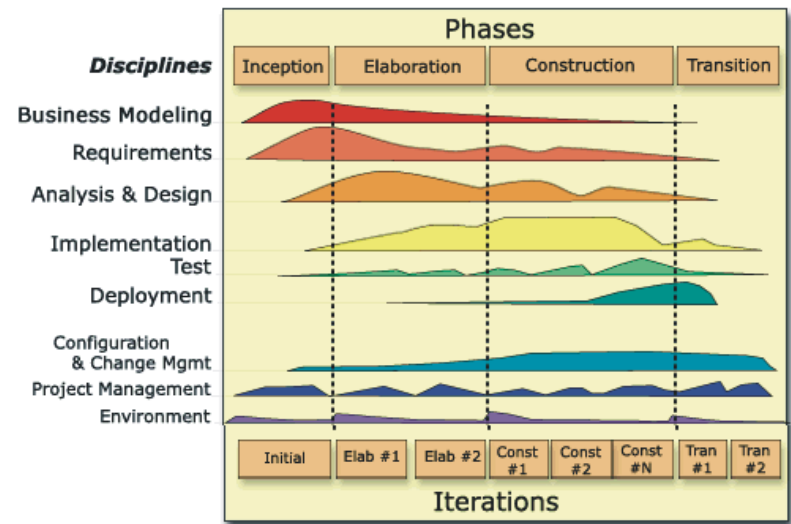




Shrnutí

► Iterativní vývoj

- Risk and user-priority driven
- Process focus on architecture
- Requirements drive design and implementation
- Models abstract the system
- Guidance for activities and artifacts



... but waterfall is not dead

Research at The Standish Group also indicates that smaller time frames, with delivery of software components early and often, will increase the success rate. Shorter time frames result in an iterative process of design, prototype, develop, test, and deploy small elements. This process is known as "growing" software, as opposed to the old concept of "developing" software. Growing software engages the user earlier, each component has an owner or a small set of owners, and expectations are realistically set. In addition, each software component has a clear and precise statement and set of objectives to be less complex. Making the projects simpler causes only confusion and increased cost.

THE STANDISH GROUP REPORT

© The Standish Group 1995. Reprinted here for sole academic purposes with written permission from The Standish Group.

CHAOS

► Varianty dle velikosti projektu

