

9

Agilné metodiky

Marek Tavač

Obsah

- Agilné metodiky: Čo to je?
- Extrémne programovanie
- SCRUM
- Test Driven Development
- Záver

Súčasnoscť

- Hlavný problém: rýchlosť vývoja
- Niektoré metódy už nie tak efektívne
- Nielen dobre, ale hlavne rýchlo
- Zmeny:
 - Väčšia konkurencia
 - Menšie programátorské tímy
 - Čo najrýchlejšie nasadenie systému
- Potrebujeme metodiky na rýchly a kvalitný vývoj „malých“ aplikácií

Agilné metodiky - vznik

- Alistar Cockbun, Kent Bleck, Ward Cunningham, Martin Fowler, Ken Schwaber, Jeff Sutherland, ...
- Utah – 2001: Manifest agilného vývoja
- Aliancia pre agilný vývoj softvéru
- Jedinou cestou ako overiť správnosť navrhnutého systému, je čo najskôr ho vyvinúť, predložiť zákazníkovi a na základe jeho reakcie upraviť
- Nový smer pre SI
- Aké sú základné premenné pre vývoj?

Tradičný verzus agilný prístup

Tradičný prístup

Fixné

Funkcionalita

Agilný prístup

Čas

Zdroje

Čas

Zdroje

Funkcionalita

Variabilné

Základné princípy

- Iteratívny a inkrementálny vývoj
- Priama osobná komunikácia v tíme
- Stále spojenie so zákazníkom
- Rigorózne, opakované, priebežné automatizované testovanie
- Vyžadujú menej formálnych a byrokratických artefaktov
- Najsilnejší dôraz na zdrojový kód

Preferenčné tézy agilného programovania

- Prijat' a umožniť zmenu je omnoho efektívnejšie
- Je treba byť pripravený reagovať na nepredvídané udalosti

Preferenčné skupiny

Prednosť	Pred
Individuality a komunikácia	Procesy a nástroje
Prevádzky schopný softvér	Obsiahla a objemná dokumentácia
Spolupráca so zákazníkom	Uzatváranie rôznych zmlúv
Reakcia na zmenu	Striktné plnenie plánu

Základné tézy

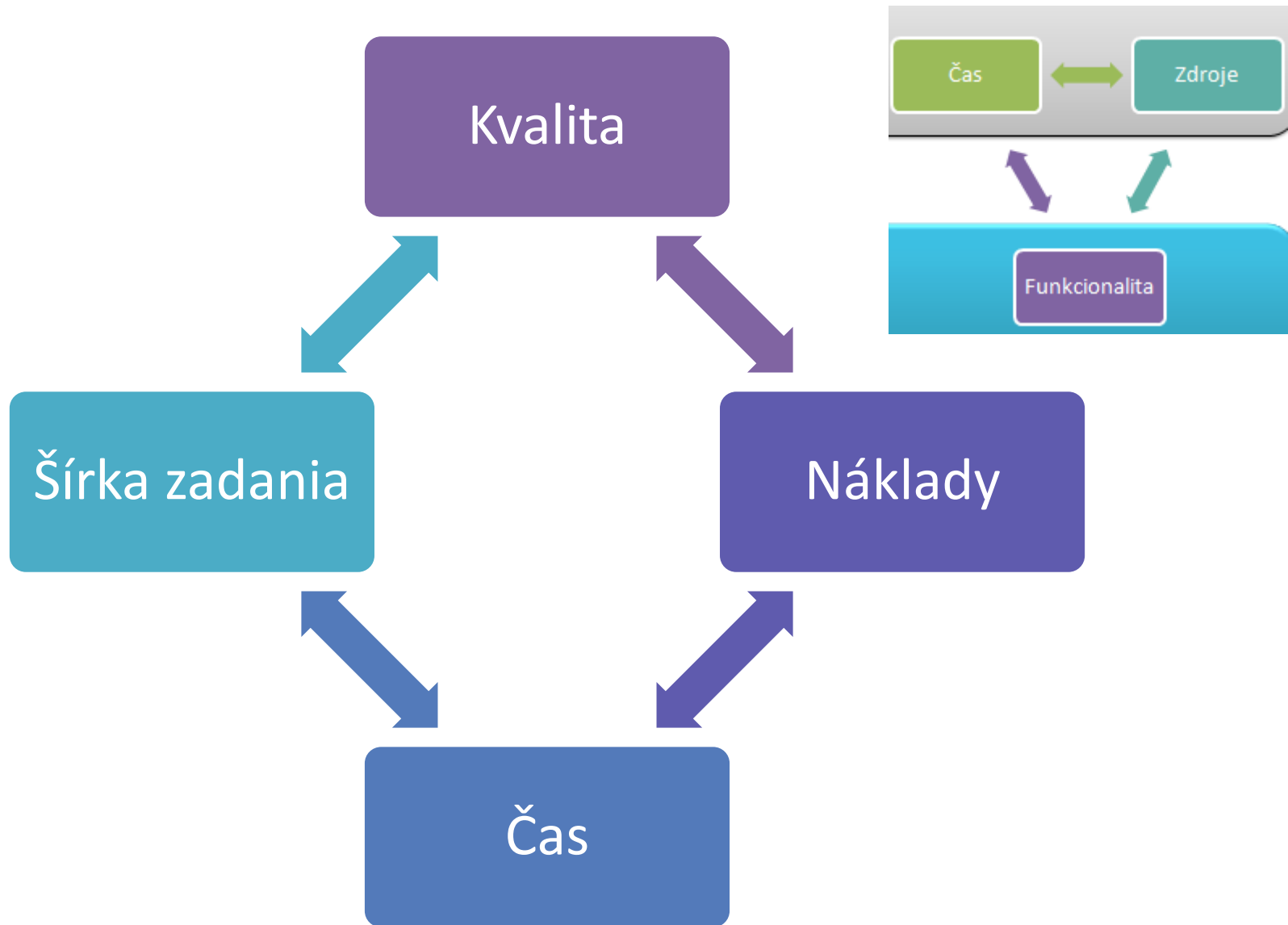
- Hodnota pre zákazníka
- Zmeny sú výhodou
- Časté dodávky
- Zákazníci pracujú s tímom
- Motivácia je kľúčová
- Vzájomná konverzácia
- Úspech posudzovaný podľa fungovania
- Udržateľný vývoj
- Perfektný návrh a riešenie
- Jednoduchosť
- Kreativita
- Ako zvýšiť efektivitu?

Extrémne programovanie

Jediným exaktným, jednoznačným,
zmerateľným, overiteľným
a nezpochybiteľným zdrojom
Informácií je zdrojový kód

Účinný, efektívny, ľahký, flexibilný a
zábavný spôsob vývoja.
Veľmi rozumný.

Premenné vývoja softvéru



XP hodnoty

Komunikácia

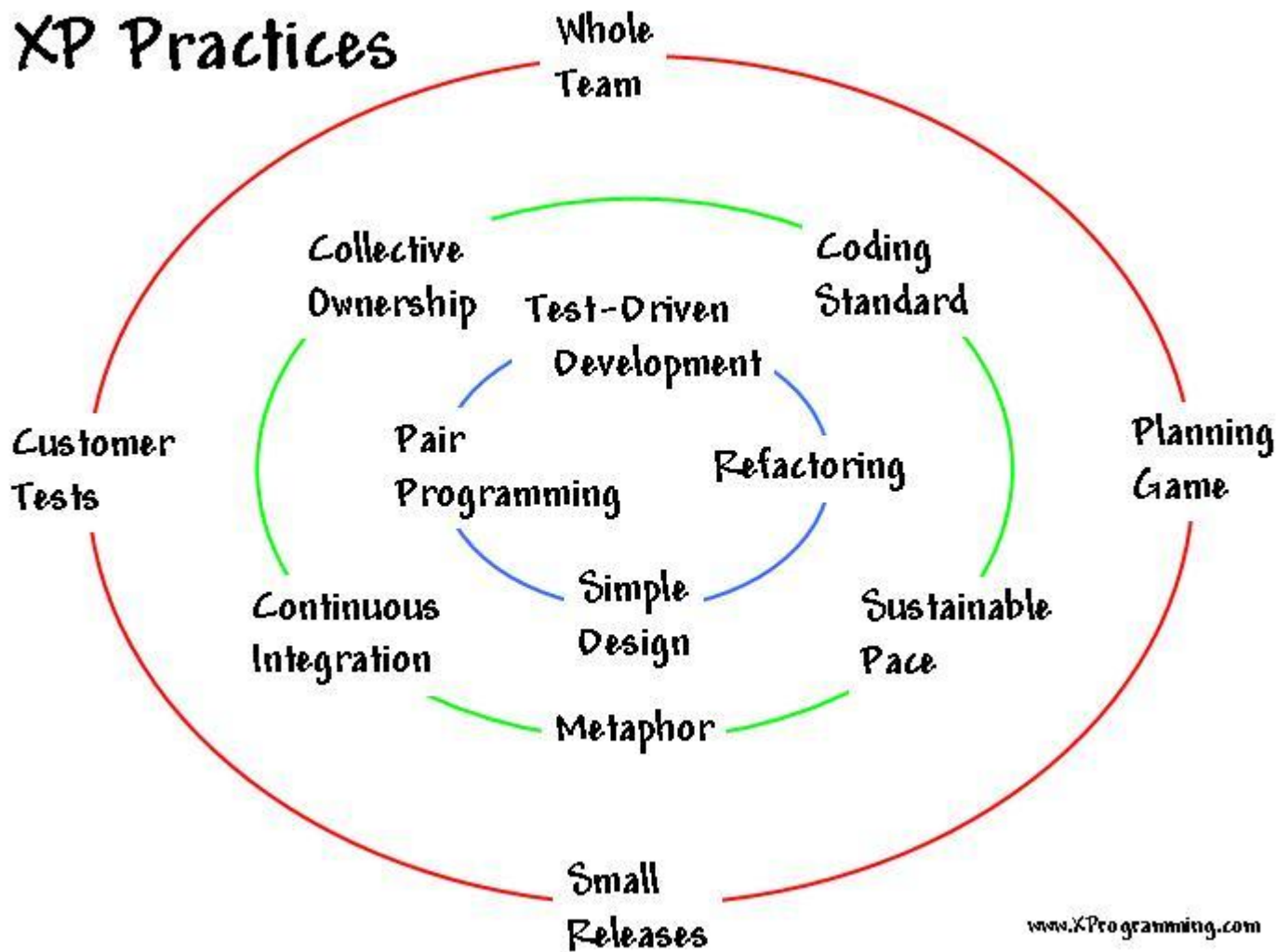
Jednoduchosť

Spätná väzba

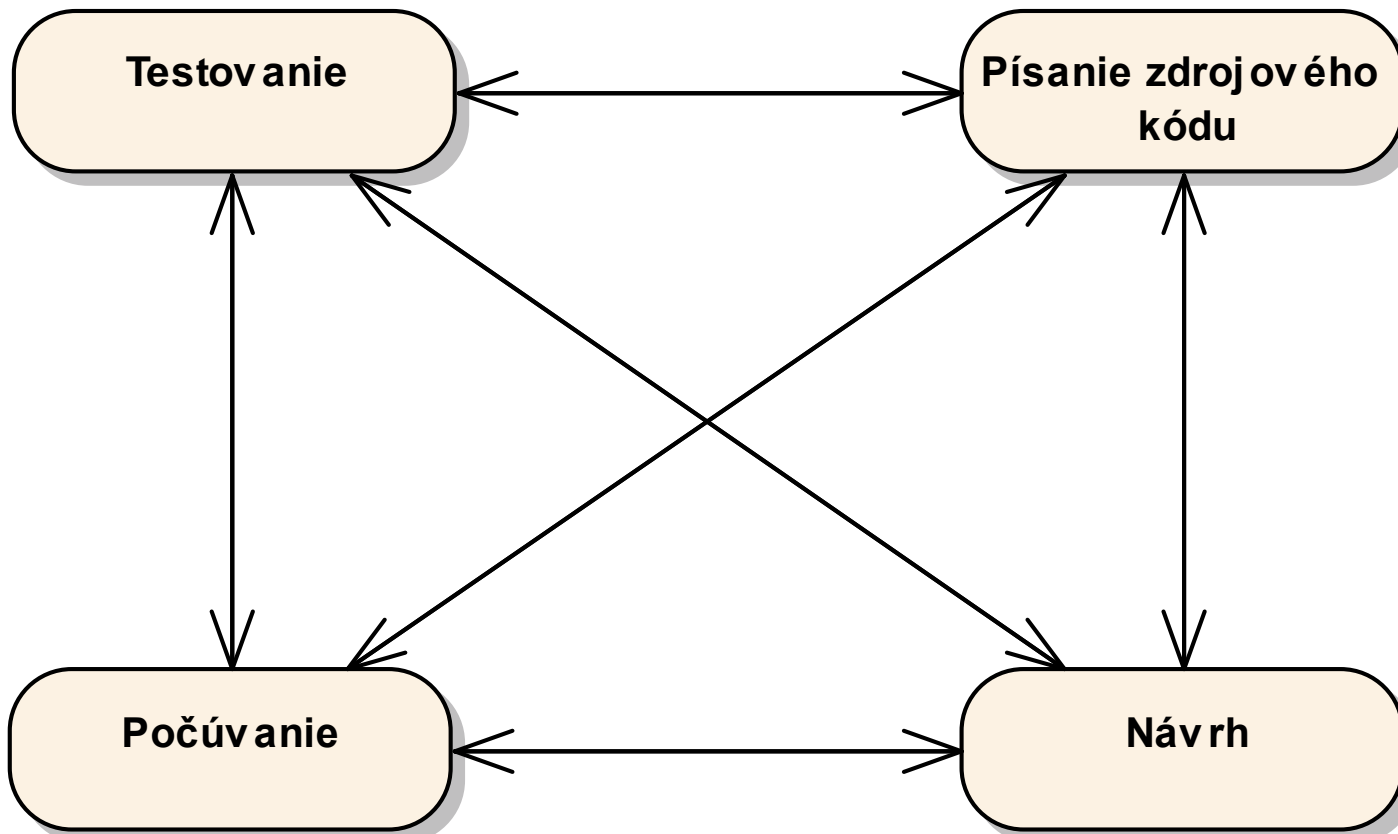
Odvaha

Rešpekt

Základné postupy



act XP Činnosti



Výhody

- **Práca v súlade s inštinktmí**
- Iteratívna
- Inkrementálna
- Priamy postup k cieľu
- Bez formalít
- Podpora v IDE, CASE

Nevýhody

- Detaily jednoduché, zložité vykonať
- Ťažké zavedenie
- Všetci členovia tímu
- Nie vhodné pre každého človeka

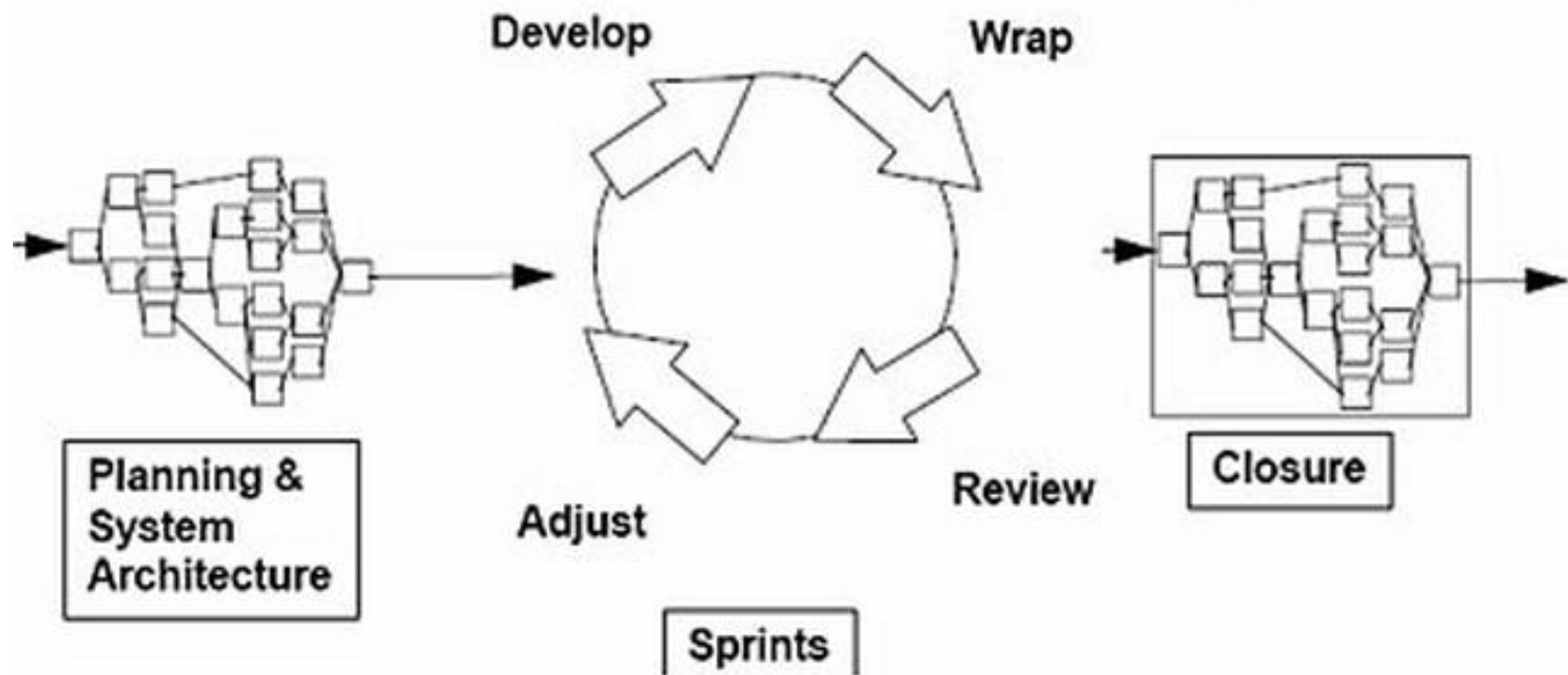
Cieľom tímu je „dotlačenie lopty“ na požadovanú pozíciu

Dopredu nevieme úplne presne, čo bude pri vývoji nutné robiť. Zavádza teda každodenné stretnutia, ktoré prinesú slabé, zato časté záblesky svetla

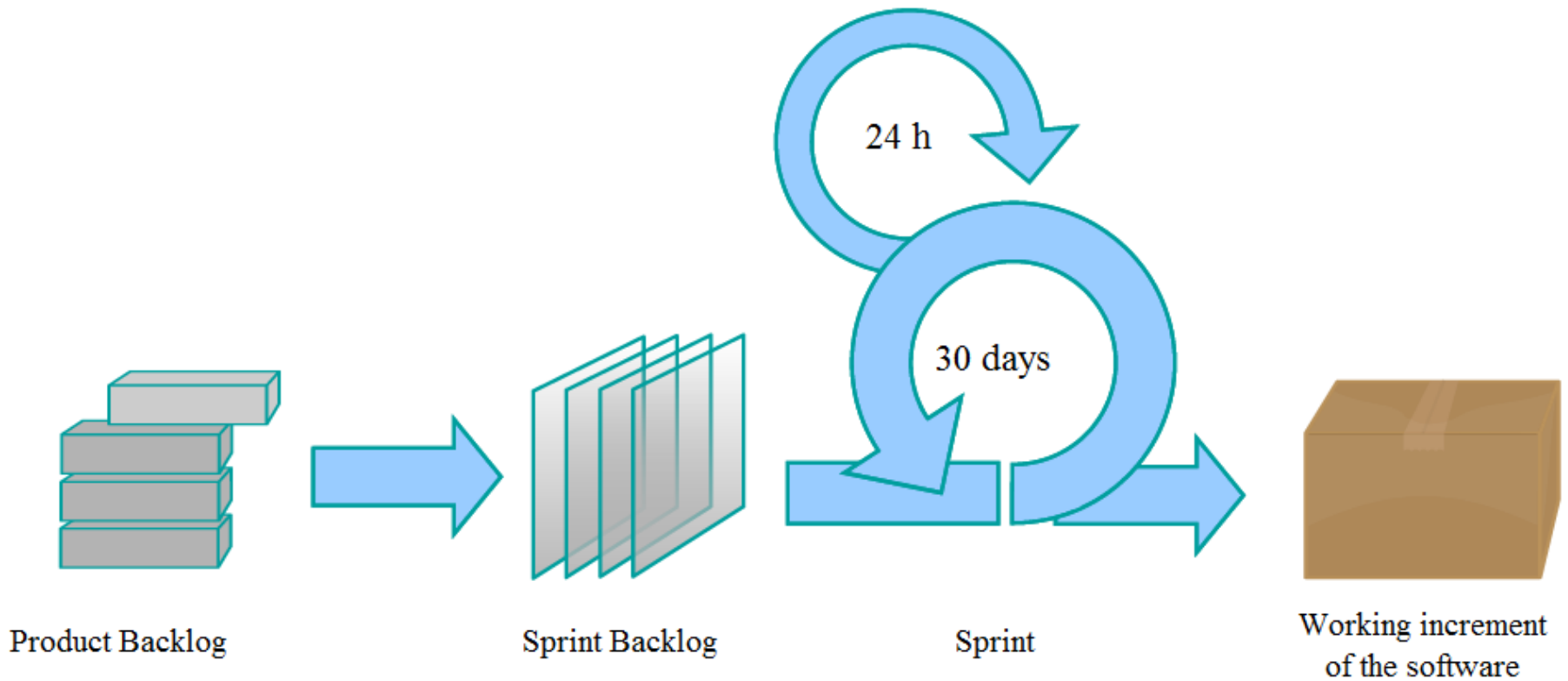
Klíčové pojmy

- Flexibilné predmety dodania
- Flexibilný harmonogram
- Malé tímy
- Časté revízie
- Spolupráca
- Backlog
- Šprint
- Riziko
- Scrum meeting

SCRUM fázy



Fáza Vývoj



Výhody

- Reakcie na zmeny
- Sloboda voľby riešenia
- OOP
- Prepracovaný spôsob odhadu pracnosti

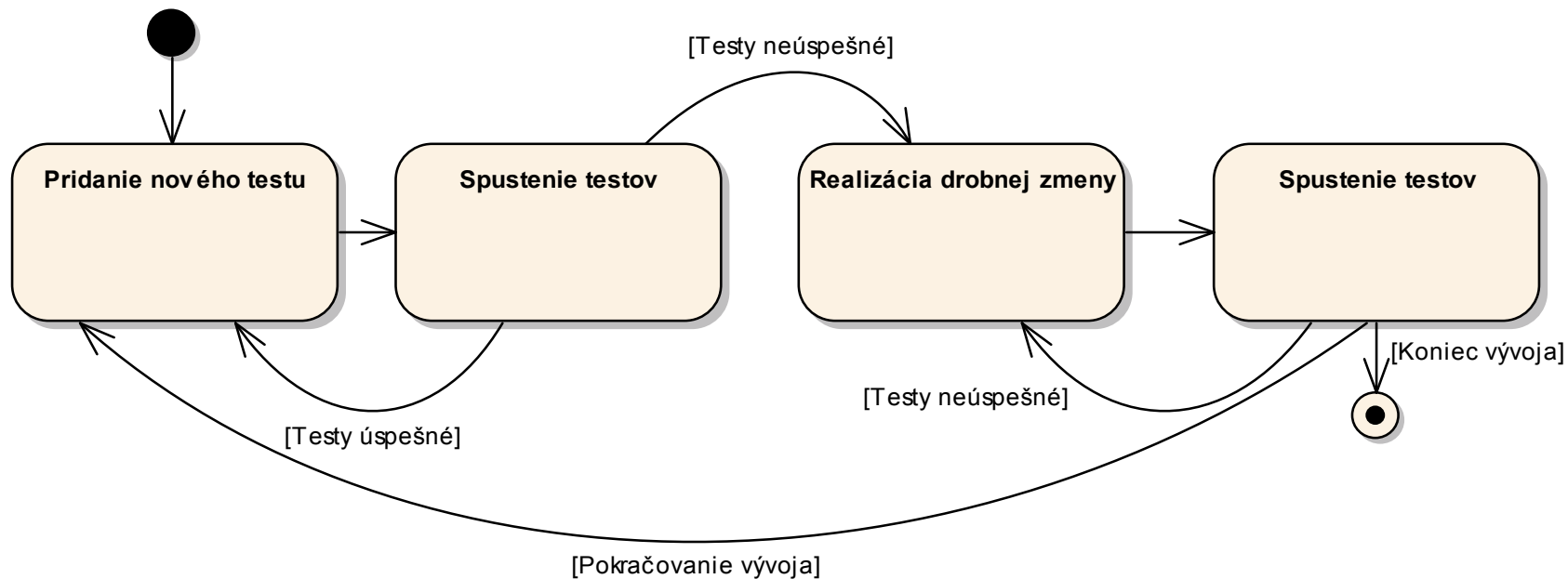
Nevýhody

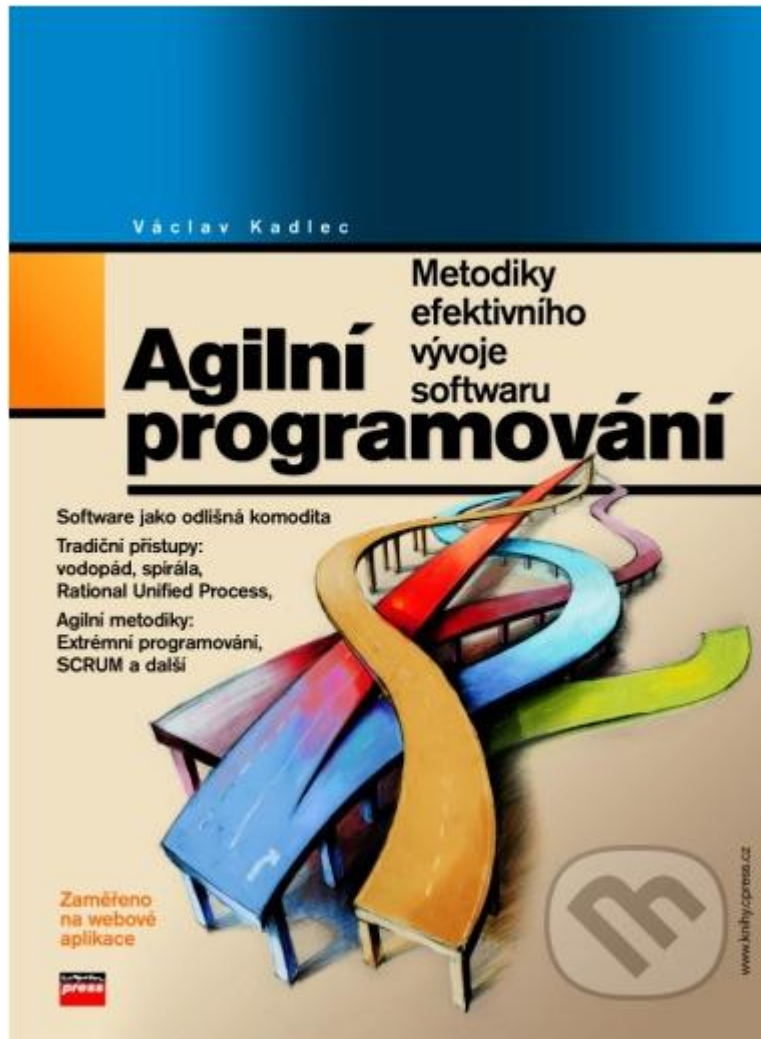
- Skôr súhrn vzorov
- Ťažké zavedenie
- Všetci členovia tímu
- Nie vhodné pre každého človeka

**Testovací kód musí byť dokončený
ešte pred začiatkom písania
testovaného kódu**

TDD kroky

act TDD





- Agilní programování
Metodiky efektivního vývoje softwaru
- Václav Kadlec