

# Řízení projektu

A také důsledky (ne)kvality dat

Předmět mnoha kurzů a knih

Zde hlavní fakta a prostředky možné  
podpory řízení projektu

# Definice projektu podle ISO/CSN 1006

- Projekt je jedinečný proces sestávající s více koordinovaných a řízených činností a termíny zahájení a ukončení prováděný pro dosažení předem stanoveného cíle, který vyhovuje specifikovaným požadavkům, včetně omezení danými časem, náklady a zdroji.
- Projekt je síť činností mající jeden začátek a jeden konec, alokované zdroje a směřující k vytvoření určitých produktů nebo dosažení určitých změn. S tím je obvykle spojeno riziko

# Fáze projektu

- Vize, proč, cíle (co chceme dosáhnout)
- Plán, kdy, co, s kým, za kolik, správa rizik
- Vlastní vytváření produktu, může se opakovat podle (rámcové) smlouvy
  - Provedení, vytvoření výstupů
  - Převzetí resp. akceptace zákazníkem
  - Vyhodnocení, užívání, podpora

# Řízení SW projektů pro IS

- V podstatě stejné jako u projektů jiných technických oborů
- Rozdíly
  - Velmi malá opakovatelnost, málo použitelných vzorových řešení (best practices, case studies), rychlé změny oboru
  - Značný vliv důsledků technologie, např. volby SOA. agilita
  - Nizká cena vytváření nikoliv však nutně cena užívání kopií
  - Značná svoboda volby způsobů řešení (to se mění)
  - Nutnost spolupráce s uživateli při vývoji

*Nutná kombinace manažerských a informatických znalostí (někdy project leader\* project manažer, jindy má vedoucí projektu zástupce, který se stará o administrativu), často jsou nutné i iterdisciplinární znalosti a týmová práce zahrnující i odběratele*

# Řízení SW projektů

- Role odborného vedoucího projektu vyžaduje značnou úroveň odborných znalostí (závislost organizace práce a způsobů řešení na architektuře systému), většinou je nutné rozsáhlé zapojení vedoucího (manažera) do řešení odborných problémů.
- Odborný talent by se neměl starat o každodenní chod prací. Pro tuto činnost nemívá dostatečné dovednosti. Proto bývá vhodné, aby se o rutinní záležitosti staral specialista na administrativu.
- Je pravděpodobné, že se řízení SW projektů stane v budoucnu podobnější řízení projektů v jiných technických oblastech (současná praxe se podobá vztahu mistra a tovaryšů).

# Mnohé rysy IT jsou překvapivě podobné jiným průmyslovým oborům

- Kódování je řemeslo, velký význam talentu
  - Automatizuje se, scrum je vlastně obdoba manufaktury
  - Velké rozdíly v rychlosti kódování a v kvalitě kódu,
- Dobří kódéři podceňují analytiku a manažéry a platí to i naopak, škodí to obojím
  - Kecálisti kontra nádeníci
  - Problém využívání špičkových talentů

# Řízení SW projektů

1. Plánování, včetně zabezpečení kvality, síť aktivit
2. Vytvoření a řízení a údržba řešitelského týmu, podpora profesního růstu členů týmu a kvality nástrojů
3. Dekompozice problému a zajištění subdodávek
4. Každodenní řízení (operativa projektu)
5. Spolupráce (někdy i boj) s nadřízenými a se „sponzory“ a zadavateli (stakeholders) projektu, organizace a podpora spolupráce týmu s lidmi mimo tým

# Infrastruktura projektu

- Nástroje a prostředky (SW, HW)
  - Vývojové prostředky
  - Propojení v týmu uvnitř i navenek
  - Dostupnost dat, prostředků jako jsou tiskárny, skenery, kreslicí zařízení atd.
- IS projektu, spousta komerčních produktů, viz Computer World 5/2012, často stačí i jednoduché vlastní systémy
  - Nic se nesmí ztratit, vše potřebné má být dostupné
  - Tlak na disciplinu
  - Nutné pro kontrolu průběhu



# IS projektu, nejdůležitější části

- **Domovská stránka**, měla by obsahovat stručný název, dlouhý název a logo projektu, blog projektu
- **Log** všech komunikací (hlavně e-mailu)
- **Dokumenty** (fulltextová databáze), každý dokument s id tvůrce a id vlastníka, především
  - Všechny verze vývojových dokumentů, především vize-cíle, specifikace požadavků, důležitá data návrhu a samotné programy včetně testových dat
  - Deník projektu se záznamy jednání, zápisy, FAQ, výsledky testování a zkušenosti z provozu
  - Texty smluv, záznamy jednání
- Někdy je třeba doplnit i id oponentury apod.
- Mnohé prostředky jsou open source, často se používá MS Project. Mnohé je vhodné pro velké projekty

# Řízení projektu při agilním vývoji

- Důvody IS projektu, opakování
  - Vše je po ruce,
  - Podpora řízení a kontrole
  - Nic důležitého se neztratí
  - Na nic se nezapomene
  - Vytváření business intelligence, růst týmu
- Platí i pro agilní formy vývoje (i tam se dá leccos zapomenout, některé požadavky se zapomenou, není vždy zaručeno, že se zadavateli nebudou spory, není zaručena vystopovatelnost a udržitelnost, je žádoucí uplatnit vyšší varianty CMM).

# Plán zajištění kvality, ISO 900003, ISO250xx, ISO 20000

1. Účel
2. Seznam dokumentů, na něž se plán odvolává.
3. Popis organizace týmu a rozdělení odpovědnosti.
4. Seznam úkolů pro zajištění kvality ve vazbě na etapy životního cyklu, mj. pravidla provedení kontrol, oponentur a auditů.
5. Seznam dokumentů, které musí být vypracovány..
6. Popis metod, praktik a konvencí, např. normy na kódování.
7. Prováděné inspekce, revize a audity.

# Plán zajištění kvality, ISO 900003

8. Řízení konfigurace, tj. metody a prostředky kontroly toho, zda jsou spojovány správné moduly a jejich verze, řízení a kontrola změn .
9. Metody evidence a způsob řešení zjištěných problémů a závad.
10. Použité softwarové prostředky a použité metodologie.
11. Metody kontroly kódu, tvar knihoven a normy jejich použití.

# Plán zajištění kvality, ISO 900003

12. Způsob ochrany médií a záznamy na nich: zálohování, ochrana před neautorizovanými zásahy, uchovávání verzí atd.
13. Pravidla kontroly subdodávek.
14. Pravidla údržby dokumentů nutných pro zajištění kvality.
15. Audity.

Modernizace přístupu v ISO250xx

# Seznam úkolů, bod 4

1. Inspekce požadavků na software.
2. Inspekce předběžného návrhu, ověření technické proveditelnosti (feasibility study).
3. Inspekce návrhu, ověření, zda návrh odpovídá požadavkům.
4. Oponentura způsobu testování, jeho adekvátnosti a úplnosti metod.
5. Kontrola dodržení funkcí před předáním, ověření, zda funkce již realizovaného softwaru odpovídají specifikacím.

# Seznam úkolů, bod 4

7. Fyzická kontrola úplnosti dodávky.
8. Průběžné kontroly. Obvykle se prověřují:
  - a) programy proti specifikacím,
  - b) správnost rozhraní,
  - c) implementační rozhodnutí -- zda zajišťuje správnost funkcí,
  - d) testy -- zda prověřují správnost všech funkcí.

# ISO 250xx, 270xx

- ISO 250xx: SW Products Quality Requirements and Evaluation,
  - několik subnorem
  - Náhrada ISIO 9000 9126
- ISO 270xx Software security management
- ISO 20000 Management and processes



# Řízení konfigurace

- Zajistit montáž v požadovaném čase ze správných a prověřených částí, obecně průmyslový problém
- Některé OS a CASE nástroje mají podpůrné prostředky (Make, SCCS)
- Plán řízení konfigurace má podobnou strukturu jako plán zajišťování kvality, s některými odchylkami, které souvisí s algoritmy zjišťování správnosti konfigurace a s pravidly pro provádění změn.
  - konvence pro tvoření jmen a čísel verzí,
  - pravidla práce s médii, zásady provádění změn,
  - doporučení zásad práce a struktury dohlížecího výboru atd.

# Plán řízení konfigurace

- odpovědnosti řešitelů a pravidla vedení projektu,
- vazby na ostatní dokumenty, především na plán zajištění kvality,
- termíny inspekcí a kontrol vázaných na vytváření konfigurace,
- způsob sledování změn rozhraní -- specifikace rozhraní, postup přijímání změn, údržba dokumentů o rozhraní, ověření rozhraní "za běhu" systému,
- použití organizačních postupů: zařazení realizovaného softwaru do vyššího celku, pravidla pro rozsah testů před zahrnutím části do celku atd.,
- metody správy konfigurace: stavba knihoven, práva přístupu, zásady ochrany, jištění, historie změn, vzpamatování po výpadku atd.,
- použití softwarových nástrojů a technik

# Varianty týmové práce

- Pevný tým
  - Úkoly se hledají k týmu
  - Víceméně nutnost u malých firem
- Najímaný tým
  - Vedení pevné pro daný projekt, ostatní se z pracovníku firmy „najímají“ podle potřeby (nejde u malých firem)
  - Vhodné pro spíše rutinní práce
  - Pro kódování a zčásti i testování se používá hosting a modifikace extrémního programování

# Problémy týmové práce

- Late team increase makes project late
- Velký rozptyl výkonnosti (1:20, jeden za 30, poněvadž tým o 20 vyžaduje řadu organizačních a podpůrných činností)
- Specifické požadavky na různé role
- Náplň rolí silně závisí na typu a velikosti projektu
  - Velký/malý, nekritický/kritický, nový/rutina, ...
- Použít function points, případně COCOMO
  - odhady jsou málo spolehlivé, zvláště v případě SME, nboť není dostatek dat pro kvalitní odhady

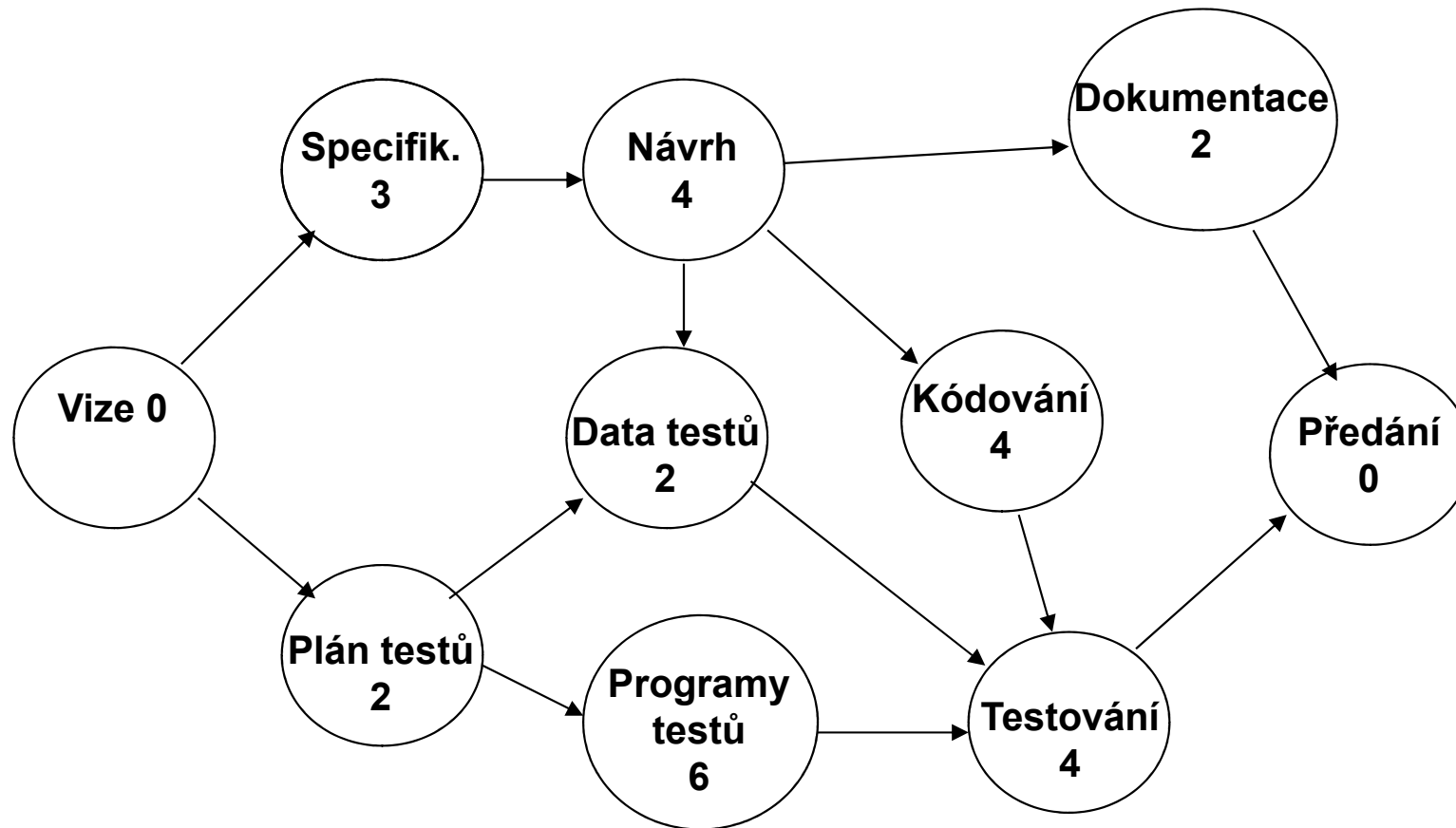
# Zbytečná administrativa

- U velkých a složitých systémů jejichž selhání může způsobit velké škody je nutné papírovat a oponovat
- U menších projektů se vyplatí udržovat ty dokumenty, které jsou celkem užitečné a s nimiž je co nejmenší práce
- Při vhodných nástrojích není generace dokumentů náročná a uživatel může být velký
- Volba optimálního rozsahu dokumentace je důležitou povinností vedoucího týmu

# Varianty vývoje obvykle pro monolity

- **Agilní forma** je vhodná pro nekritické úlohy, malý projekt (tisíce řádků), vstřícní uživatelé, kvalitní a znalí řešitelé schopní používat moderní SW nástroje
- **Rapid delivery:** Dodávka do několika málo měsíců, do pěti řešitelů. I ekonomické ztráty, střední projekty, kvalitní řešitelé
- **Express.** Větší projekty, i průměrní řešitelé, nejde o životy: 1-2 roky, 5-10 řešitelů
- **Iterace, inkrementy** v ostatních případech

# Sítový graf projektu

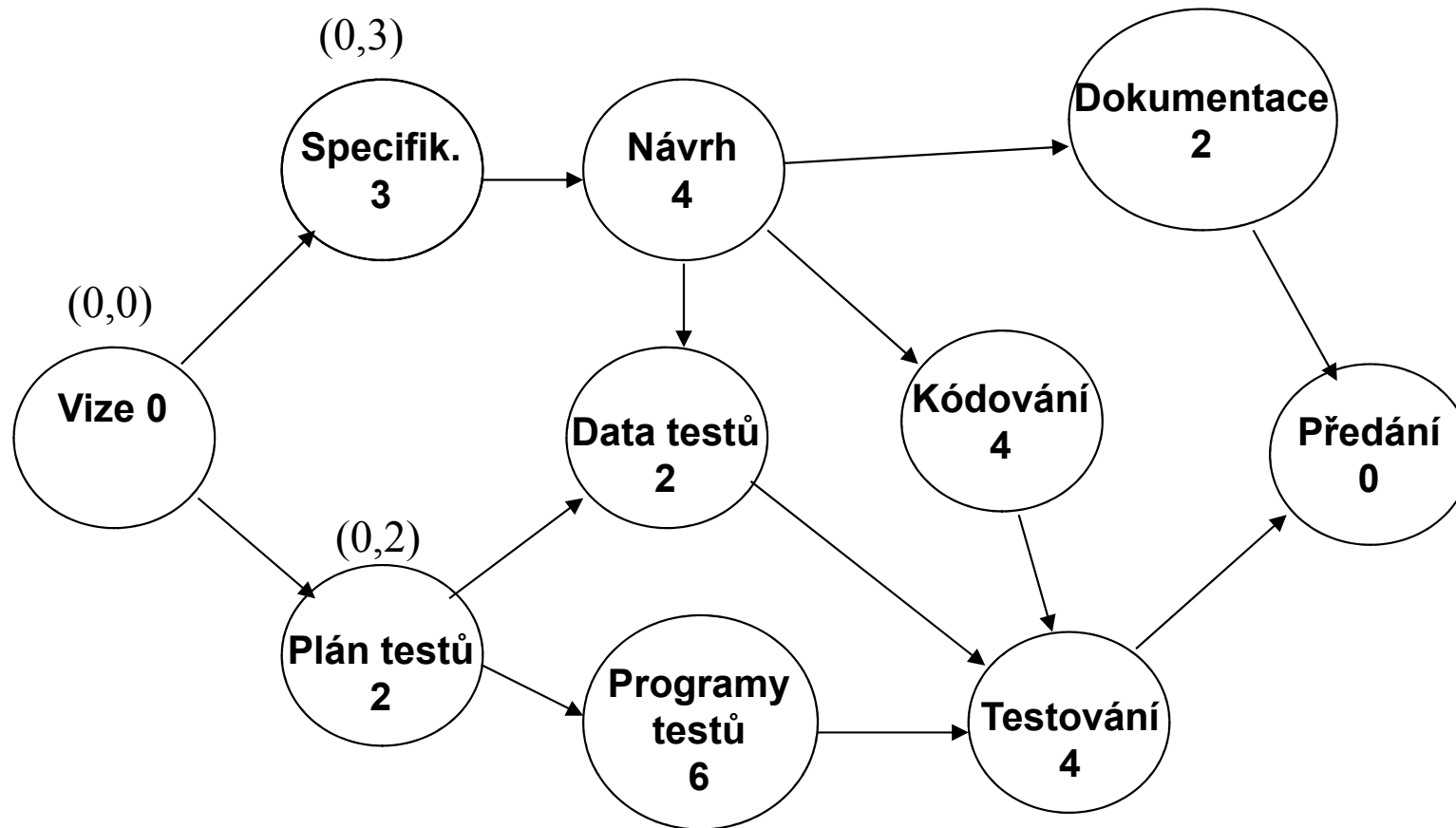


# Nejdříve možný začátek

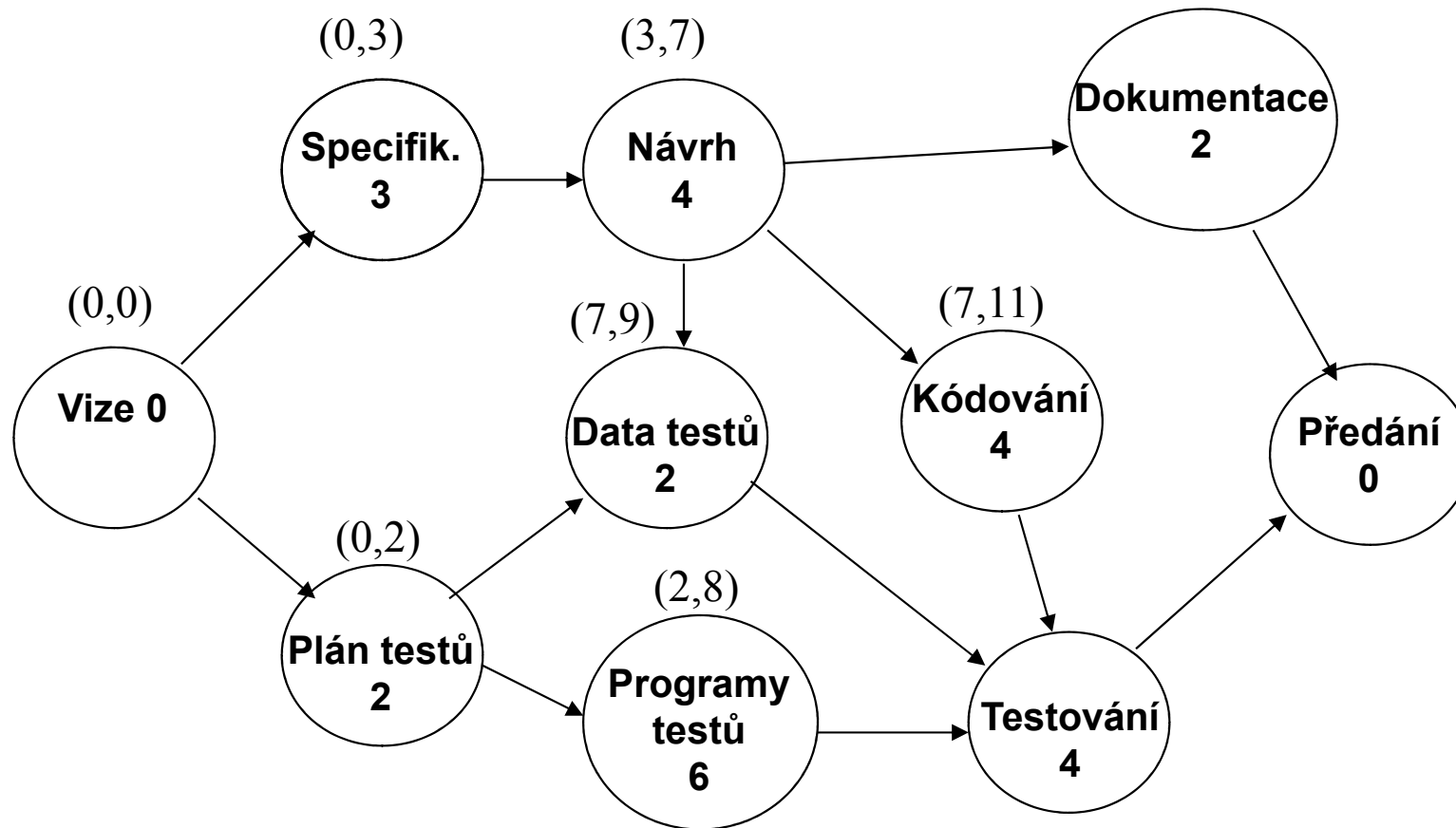
- Maximum nejdříve možných konců předchůdců
- Předpoklad, jeden zdroj, jedna nora, acykličnost grafu (dá se oslabit/obejít), souvislost grafu
- Počítá se tedy od zdroje
- Konec = Začátek + Doba řešení



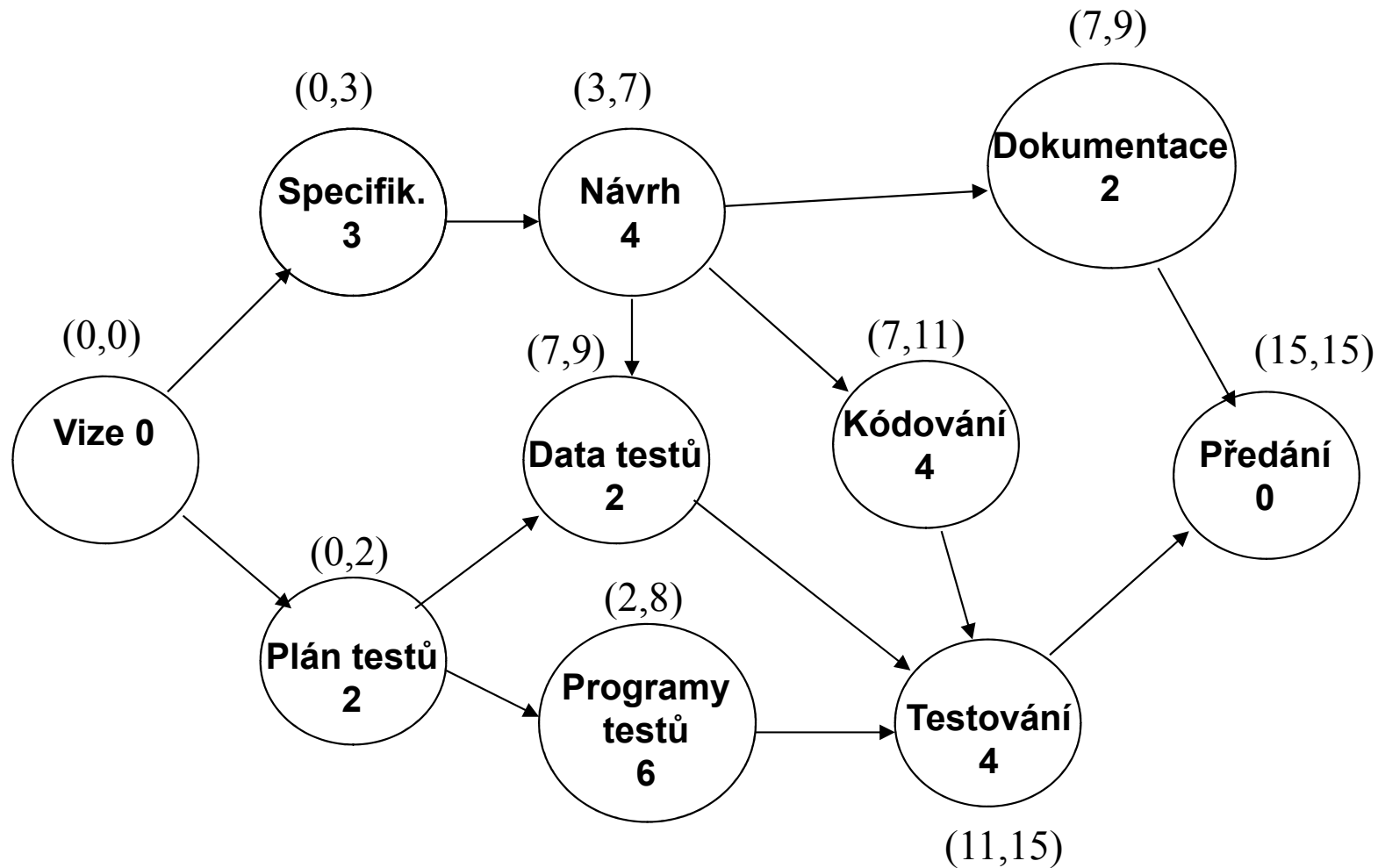
# Výpočet nejdříve možných zahájení a ukončení prací



# Výpočet nejdříve možných zahájení a ukončení prací



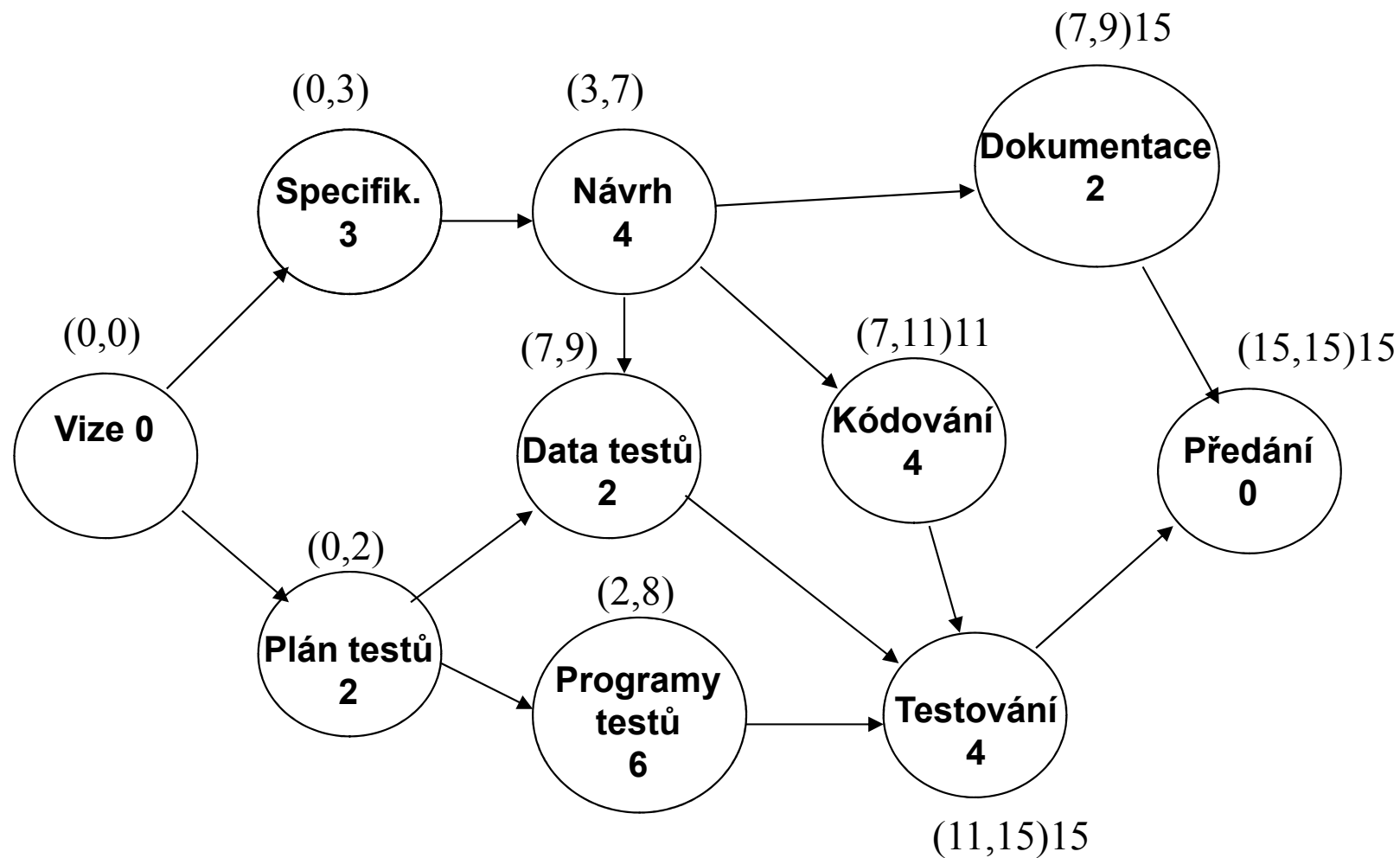
# Výpočet nejdříve možných zahájení a ukončení prací



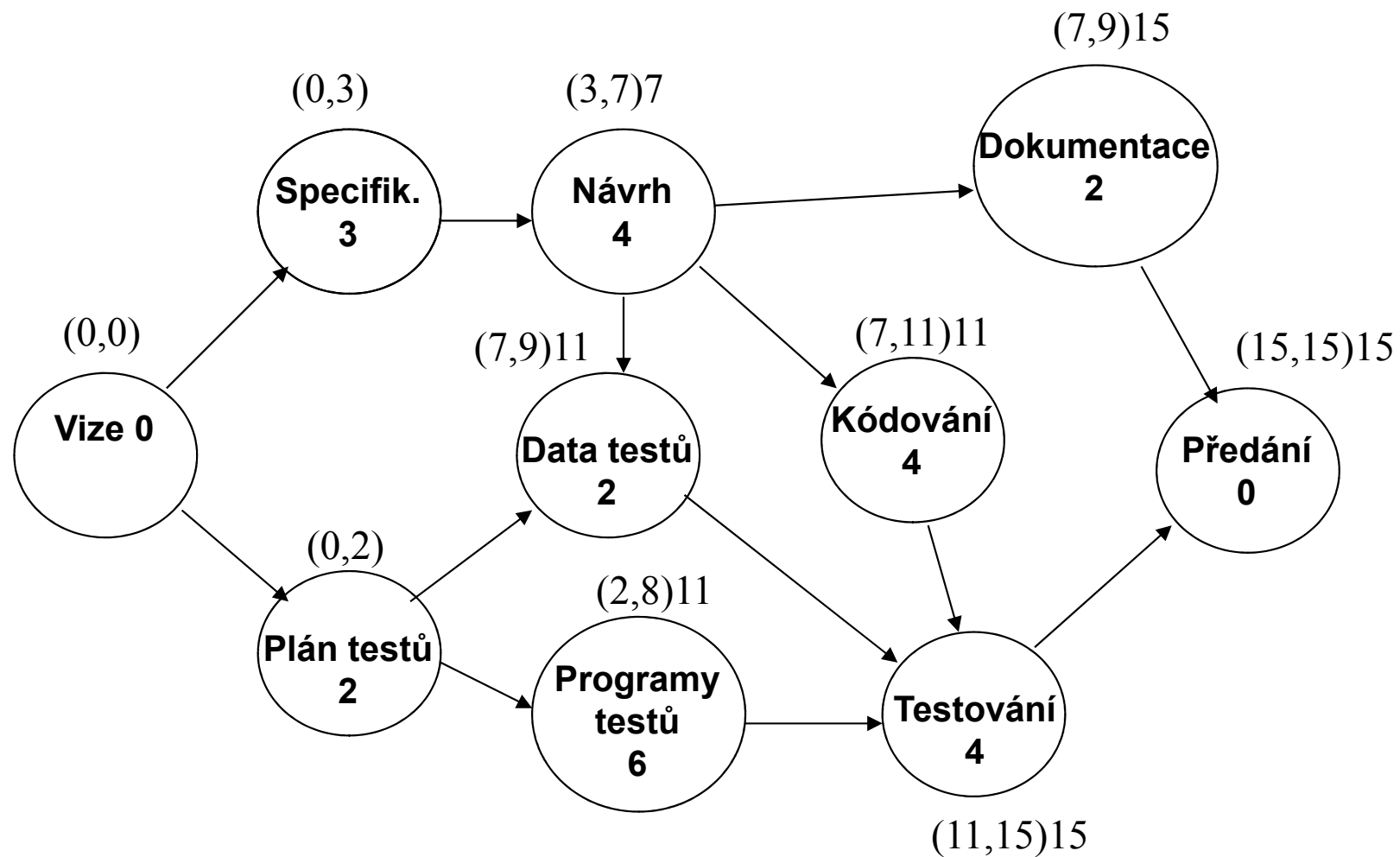
# Nejpozději možné ukončení

- Minimum nejpozději možných zahájení následníků
- Počítá se od nory

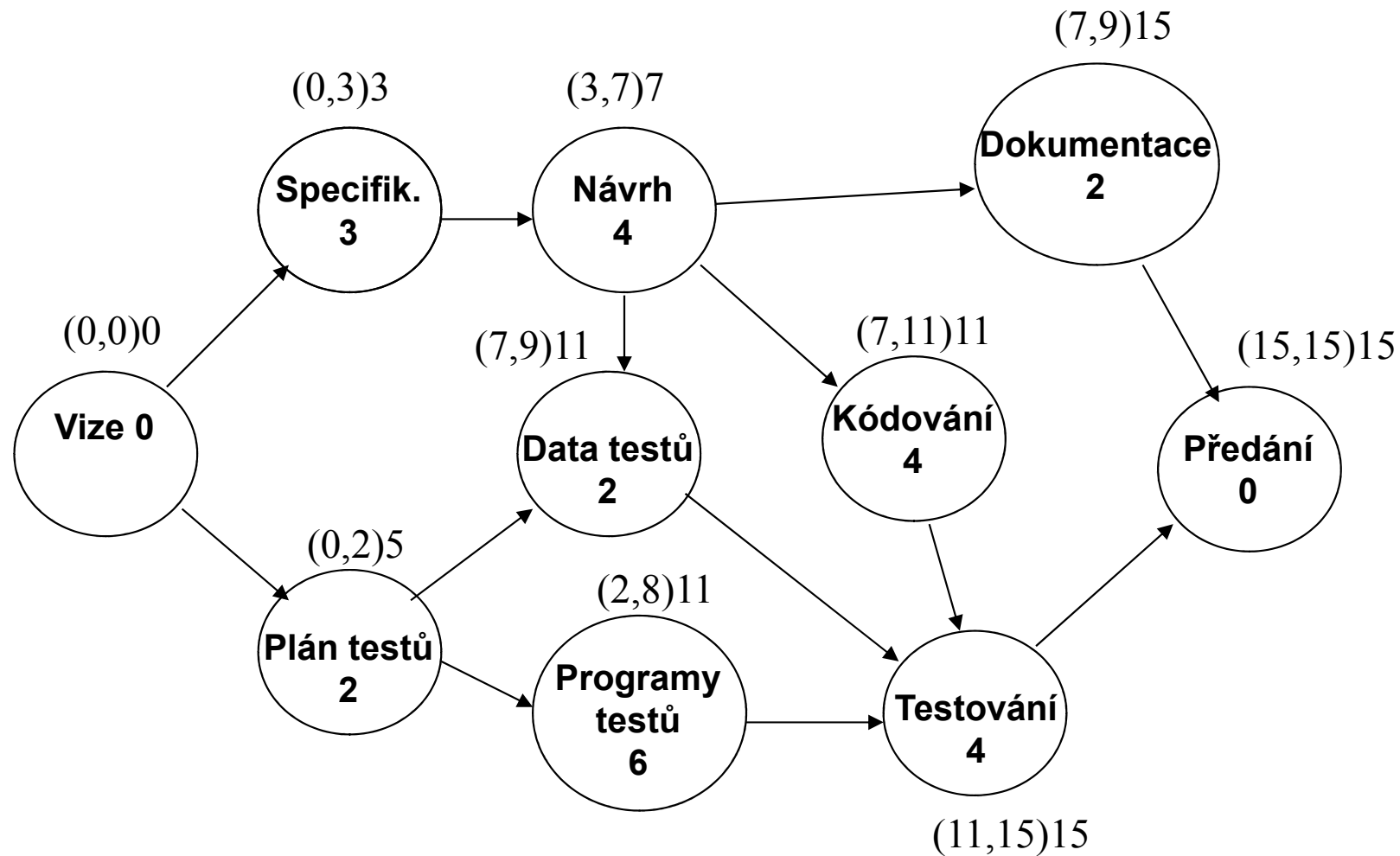
# Výpočet nejpozději možných zahájení a ukončení prací



# Výpočet nejpozději možných zahájení a ukončení prací



# Výpočet nejpozději možných zahájení a ukončení prací

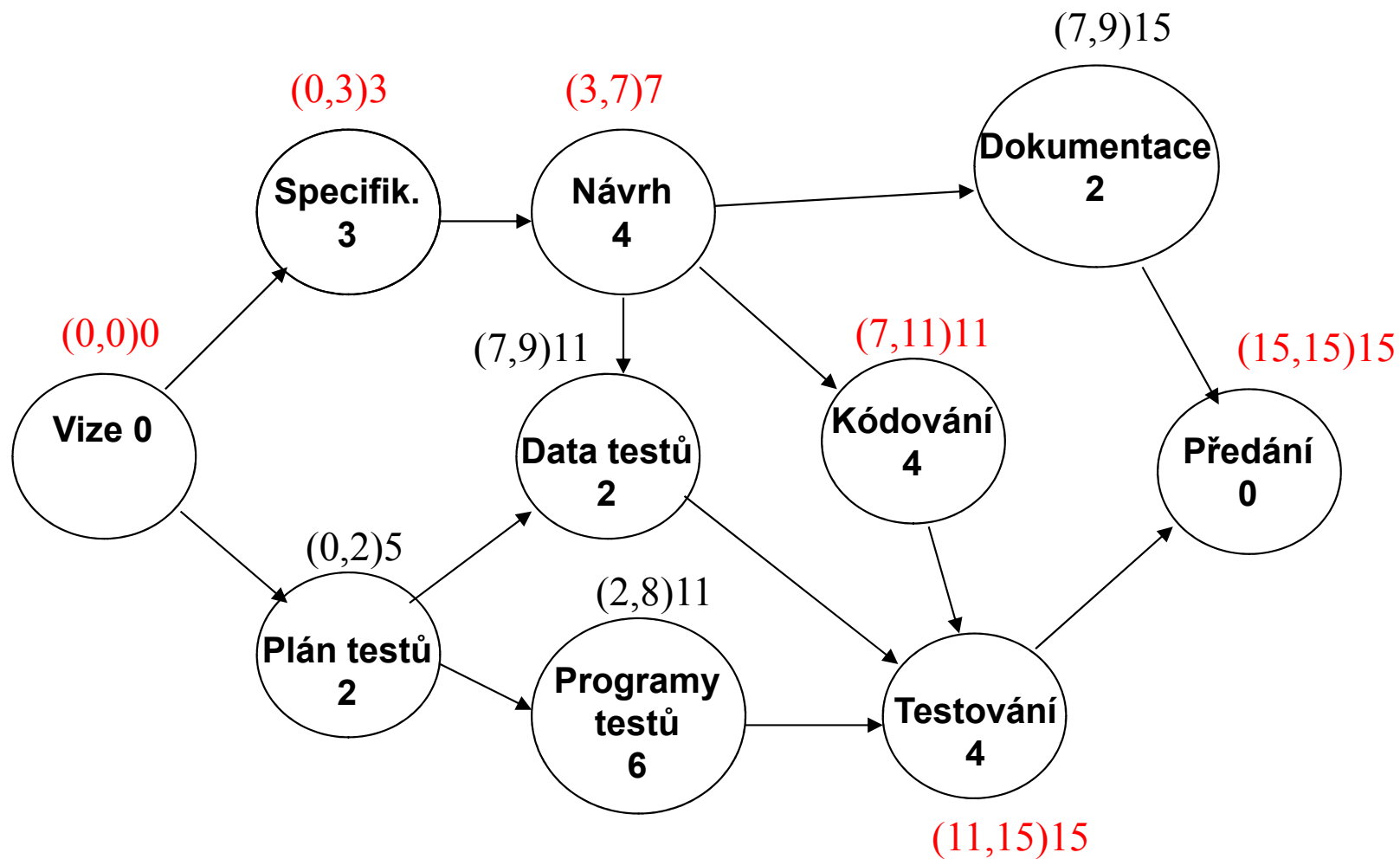


# Kritická cesta

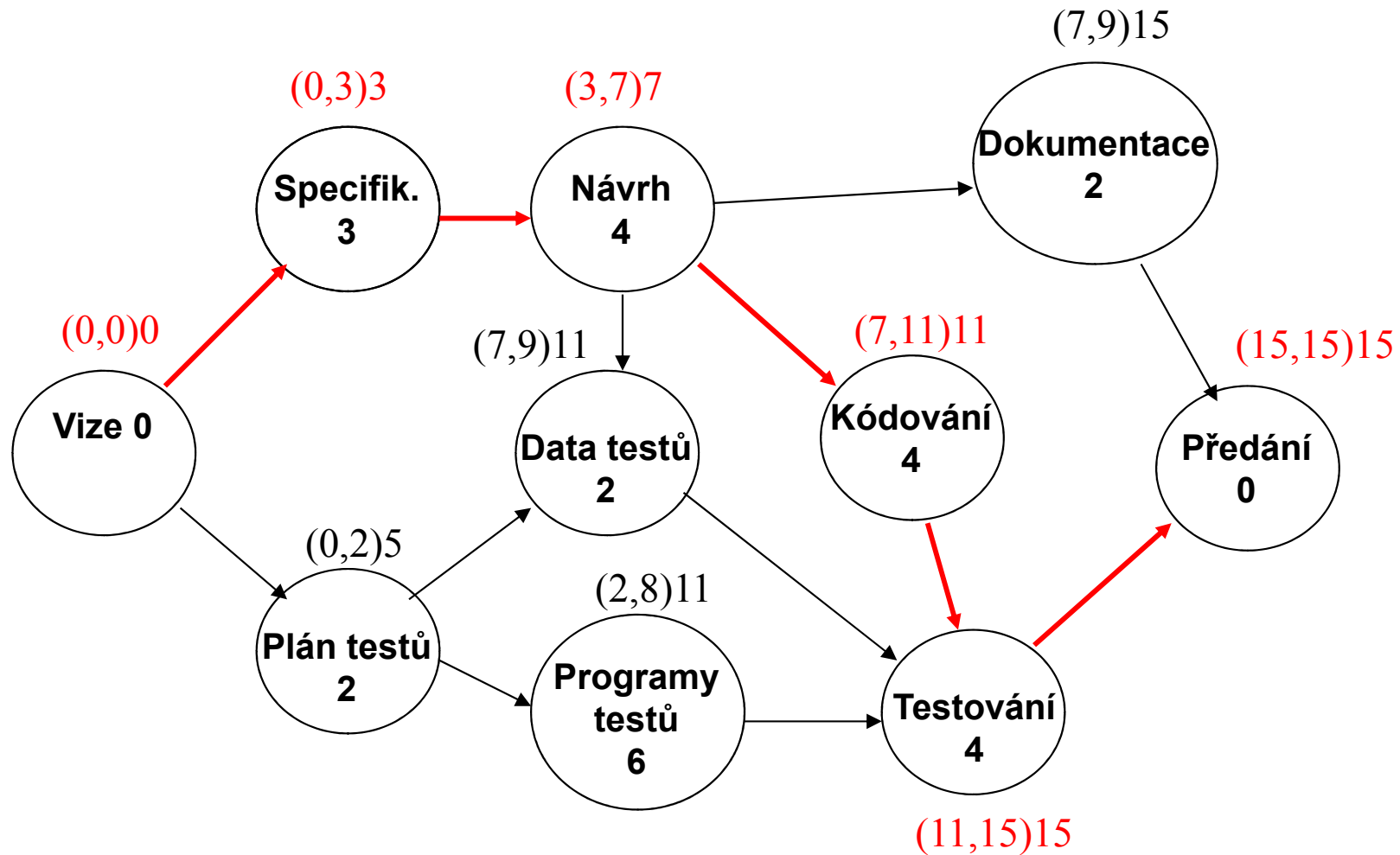
- Cesta od zdroje k noře mající tu vlastnost, že prodloužení řešení libovolné činnosti na kritické cestě zvětší dobu řešení projektu
- Kriterium – nejpozději možné ukončení je rovno nejdříve možném ukončení



# Výpočet kritické cesty



# Kritická cesta



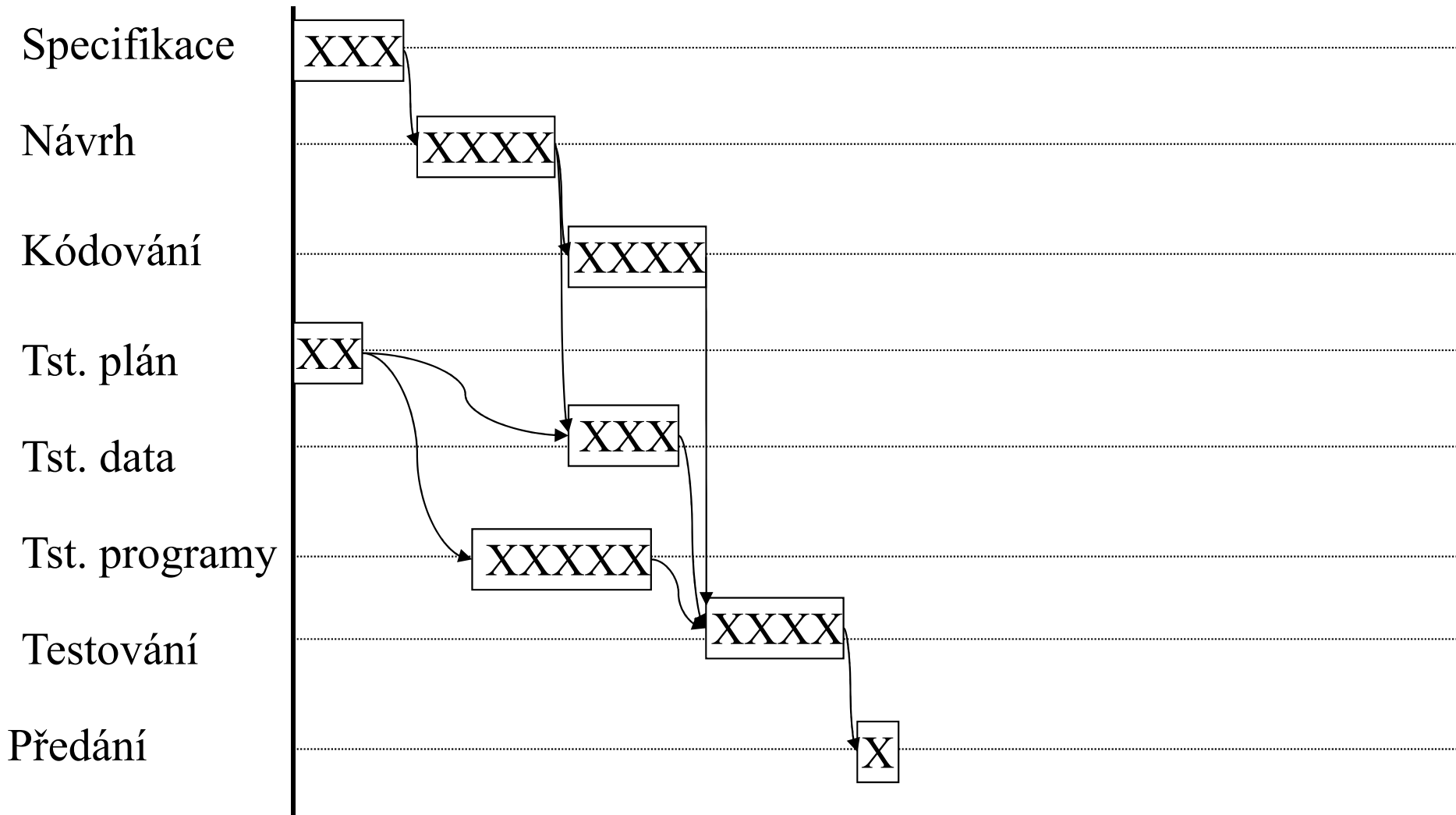
# Kritická cesta je Goldrattovo úzké místo projektu

- Zkrácením činností mimo kritickou cestu se nezkrátí doba řešení, kritická cesta je Goldrattovo úzké místo
- Manažerské pravidlo – sledovat kritickou cestu
- Kritická cesta může být i při malých změnách dob řešení nestabilní

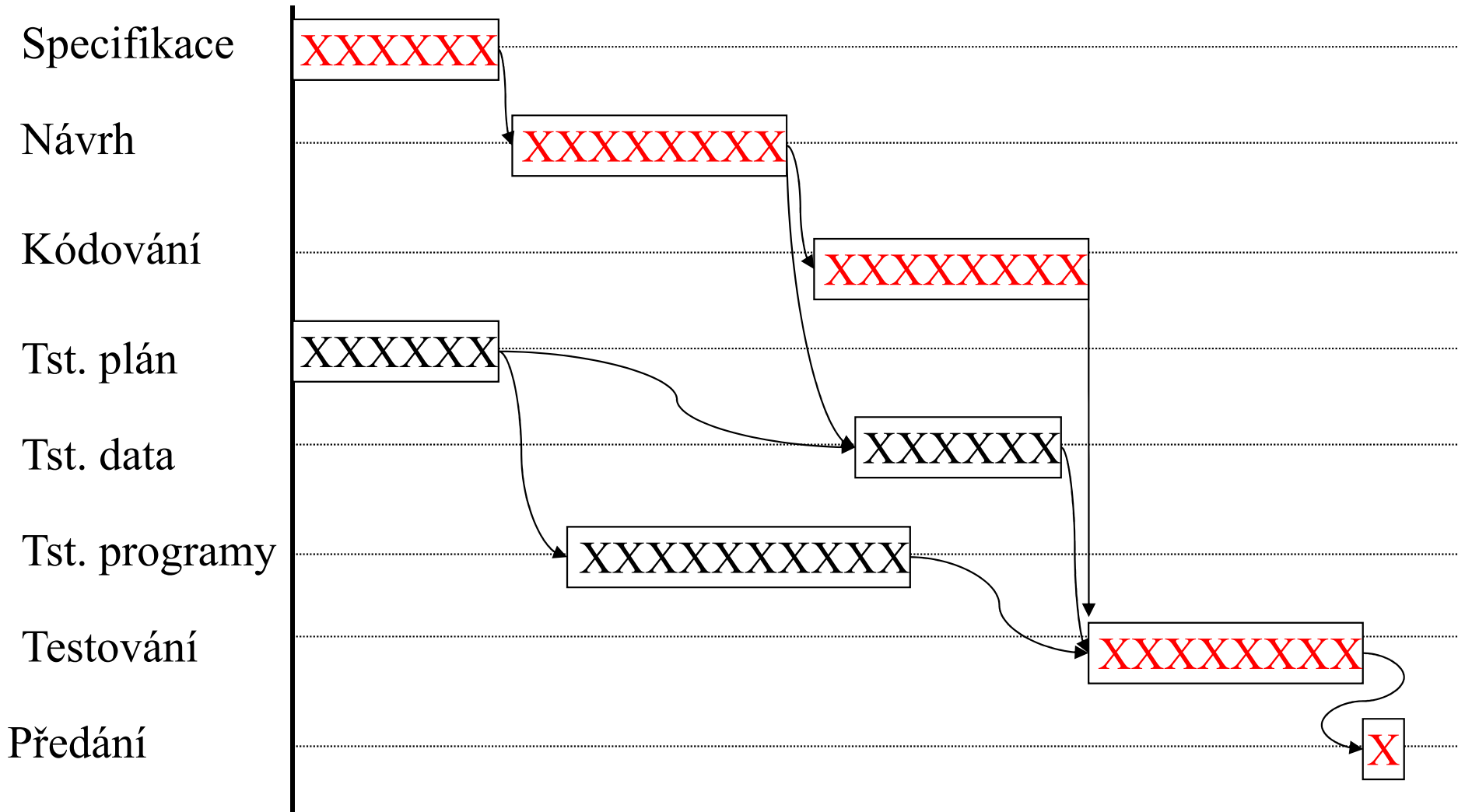
# Ganttovy grafy, grafické řešení kritické cesty

- Uzly grafu jsou úsečky v délce úměrné době umístěné na časové ose pro každou činnost tak, aby ukazoval kdy bude činnost vykonávána
- Časové závislosti a následnosti jsou vyjádřeny šipkami jako v síťovém grafu
- Kritická cesta se dá vypočítat i šoupáním úseček
- Výhodou je, že je okamžitě vidět souběžnost činností a že je to velmi názorné. Forma presentace problému je tady zcela zásadní

# Ganttovy grafy, lze najít kritickou cestu



# Ganttovy grafy, jiné měřítko, viz grafiku v MSPProject



# Další použití Ganttových grafů

- Ganttovy grafy jsou široce používány při plánování výroby, ale také při tvorbě rozvrhů, např. ve škole, jako v MS Project.
- Z Ganttova grafu projektu jsou odvozovány Ganttovy grafy využití jednotlivých zdrojů, např. rozvrhy práce jednotlivých pracovišť nebo pracovníků

# PERT grafy

- Doby práce jednotlivých aktivit nebývají známy a navíc bývají ovlivňovány různými neočekávanými událostmi. Musí se proto odhadovat.
- Metoda PERT navrhuje, by se spolu s odhadem očekávané doby řešení (jak jsme si to ukazovali při hledání kritické cesty), prováděl ještě optimistický odhad doby řešení, když to půjde velmi dobře a pesimistické doby řešení, kdy se nebude dařit. Doporučuje se tedy udávat dolní i horní hranici konfidenčního intervalu



# PERT grafy, problémy

- Doby práce jednotlivých aktivit nebývají známy a navíc bývají ovlivňovány různými neočekávanými událostmi. Musí se proto odhadovat. To bývá obtížné, ale ještě obtížnější je odhad hranic konfidenčního intervalu, hlavně té dolní,
- V reálných situacích mají pracovníci tendenci jako pravděpodobnou dobu řešení udávat horní hranici konfidenčního intervalu, neboť bývají postihováni za nedodržení termínů. Vedení má naopak tendenci prosazovat jako optimální dolní odhady.
- Jedno z řešení je metoda kritického řetězce

# Problémy kritické cesty

- Metoda kritické cesty se dá zobecnit tak, aby byl zohledněn problém soutěže a společné zdroje
- V praxi se osvědčuje při plánování činností, které se opakují a kde lze doby řešení dobře odhadnout, dokonce změřit
- V případech, kdy jsou odhady nejisté, jak je pravidlem při vývoji SW, nebývají výsledky dobré
- MS Project je založen na metodě kritické cesty. MS Project využívá rozhraní založené na Ganttových grafech.

# Problémy kritické cesty, boj o měkké normy

- V SW bývá obvykle málo spolehlivých dat potřebných pro dobrý odhad (malá opakovatelnost, nové typy požadavků, změny technologie) – je nutno se spolehnout na odhady řešitelů jednotlivých úkolů.
- Pokud někdo nesplní termín, má potíže
- *Důsledek: Odhady se licitují a volí tak, aby byla malá pravděpodobnost, že se termín nesplní (tj. volí se horní hranice konfidenčního intervalu)*

# Problémy kritické cesty, boj proti zpevňování norem

- V dobách řešení jsou velké rezervy. Přesto se i pak se termín obvykle prošvihne. Proč?
- Pokud někdo skončí dříve, práci neodevzdá z obavy, že příště by nevylicitoval měkký termín, dali by mu tvrdší termín a on by čelili riziku, že ho nesplní.

# Problémy kritické cesty, syndrom „líného“ studenta

- V dobách řešení jsou velké rezervy, proč i pak se termín projektu obvykle prošvihne?
- Pokud má někdo rezervu nezačne včas pracovat (dělá na něčem jiném). Pracovat začne až když má dojem, že už je třeba začít, že by to už při dalším čekání nemusel stihnout. Pak ale často (s pravděpodobností v jednotkách procent, ve špatně vedených týmech i v desítkách procent) nestihne svůj termín. Skluz už se obvykle nedohoní v důsledku boje proti zpevňování norem.

# Problém není v lenosti

- V době, kdy mám takzvaně čas mohu dělat něco jiného (učit se na jinou zkoušku) v podniku by vznikl zbytečně prostoj
- O lenost může také jít a to je ta nejhorší varianta, neboť poflakování zhoršuje pracovní návyky

# Problémy kritické cesty, efekt multitaskingu a vyšší moci

- V dobách řešení jsou velké rezervy, proč i pak se termín projektu obvykle prošvihne?
- Pokud se neudělají opatření, vznikne často situace, kdy nelze na určitém úkolu začít pracovat, protože se dělá na jiném úkolu, jindy nelze začít pracovat, protože předchůdce neskončil včas, což nemusí být jen lenošením – efektem líného studenta, také mohou být potíže s kapacitou uzlů (drahé stroje, specialisté), které pracují pro více projektů
- *Výsledek: Nutně dojde ke skluzům a je malá naděje, že se skluzy podaří dohnat, protože se pracovníci bojí zpevňování norem*

# Řešení, kritický řetězec

- Kritický řetěz funguje jen, když jsou řešitelé ochotni odhalit rezervy a neskryvat, že jsou hotovi dříve, než se plánovalo.
- Odměny za dodržení termínu (za zkrácení není další bonus, vedlo by to opět k licitování)
- Termín se odvozuje z očekávané doby řešení
- Navíc má každá činnost odhad doby, když jdou věci špatně (horní hranice konfidenčního intervalu)
- Termín pro celý projekt se určuje jako odhad horní hranice konfidenčního intervalu jeho řešení



# Řešení, kritický řetězec

- Každý začne pracovat hned, jak je jeho předchůdce hotov. To ale znamená, že je nutné nějak vyloučit efekt multitaskingu, To se řeší tak, že se dané činnosti postupně stále přesněji oznamuje, kdy bude třeba začít pracovat na daném projektu (činnost je tedy spravována jako autonomní služba)
- Práce se odevzdává v okamžiku, kdy je hotova. Její začátek se ale postupně zpřesňuje
- *Důsledek: Dosti často se daří, aby práce trvala přibližně tak dlouho, jako kdyby byla zcela nezávislá*

# Pravidla hry podrobněji

1. Především změníme způsob plánování prací. Nebudeme stanovovat, kdy se přesně na jednotlivých etapách začne pracovat a kdy práce skončí. Místo toho se stanoví, jak dlouho bude asi řešení etapy trvat (např. odhad střední doby práce nebo mediánu, často se volí polovina odhadu  $H$  horní hranice konfidenčního intervalu), a kdy se asi na ní bude moci začít pracovat.

# Pravidla hry podrobněji

2. Práce na etapě se zahájí co nejdříve od okamžiku, kdy je to možné práci zahájit. Aby tomu bylo možné vyhovět, je postupně zpřesňován odhad okamžiku, kdy bude možné začít na etapě pracovat. K tomu je nutné mít průběžné informace o stavu řešení předcházejících etap. Zkušenost ukazuje, že tento postup skutečně umožňuje, aby řešitelé zorganizovali práci tak, aby mohli začít na projektu pracovat hned, jak je to možné.

# Pravidla hry podrobněji

3. Od okamžiku zahájení prací se pracuje pouze na úkolech spojených s řešením etapy a pracuje se s maximální intenzitou (to vylučuje efekt líného studenta a multitaskingu).
4. Řešitelé dostatečně často předem hlásí, kdy asi budou hotovi a práci odevzdávají hned, jak jsou hotovi (to je nutné pro bod 1).

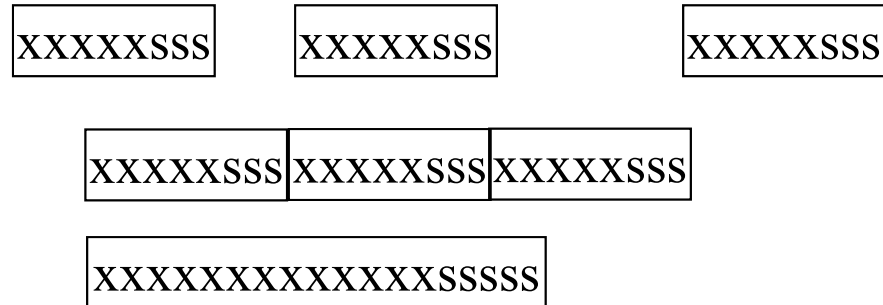
# Pravidla hry podrobněji

- Tomu lze vyhovět jen tehdy, jsou-li řešitelé ochotni pracovat naplno a odhalovat své rezervy. Musí být proto zainteresováni na úspěchu projektu a musí mít také jistotu, že se proti nim nepoužije procedura zpevnování norem.
- Vedení projektu musí naopak chápat, že jsou termíny kruté a že se často nesplní. Z nesplnění termínu by neměly být zpravidla vyvozovány žádné postihy. Všichni by měli mít prospěch ze zkrácení doby řešení a z prémie za včasné dokončení celého projektu, nejen jejich úkolu. Podstatnou roli tedy hrají psychologie a sociální aspekty fungování týmu. To je u IS standardní situace.
- Je dobré provádět analýzu dat, kdy a jak je kdo hotov s dlouhodobým cílem odhalit lenochy, netrestat ale hned kritická místa a postupy

# Potřebné znalosti

- $M$  střední hodnota (zjednodušeně průměr)
- $D$  rozptyl,  $\sigma$  směrodatná odchylka  $\sigma = \sqrt{D}$
- Rozptyl součtu nezávislých náhodných veličin je součet rozptylů (lze dokázat)
- Takže  $\sigma_{\text{součtu}} = \sqrt{\sum \sigma_i^2}$
- V běžných situacích
- $P(M-3\sigma < X < M+3\sigma) > 0.95$

# Kritický řetězec – doba řešení je součtem nezávislých n.v.



- Doba řešení  $T$  klasické metody kritické cesty bývá
- $T \geq \sum (t_i + 3 \sigma_i)$
- Doba řešení pro kritický řetězec bude většinou

$$\bullet T \approx \sum (t_i + 3 \sqrt{\sum \sigma_i^2})$$

# PřípojnÉ buffery

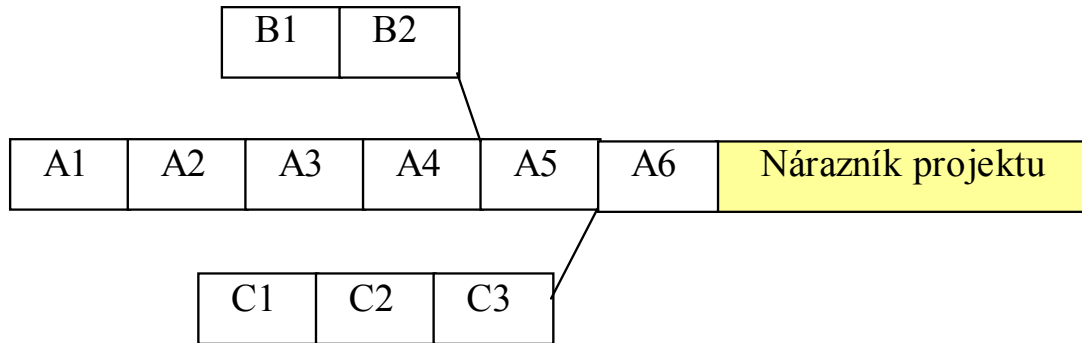
A1	$\sigma$	A2	$\sigma$	A3	$\sigma$	A4	$\sigma$	A5	$\sigma$
----	----------	----	----------	----	----------	----	----------	----	----------

A1	A2	A3	A4	A5	Nárazník projektu
----	----	----	----	----	-------------------

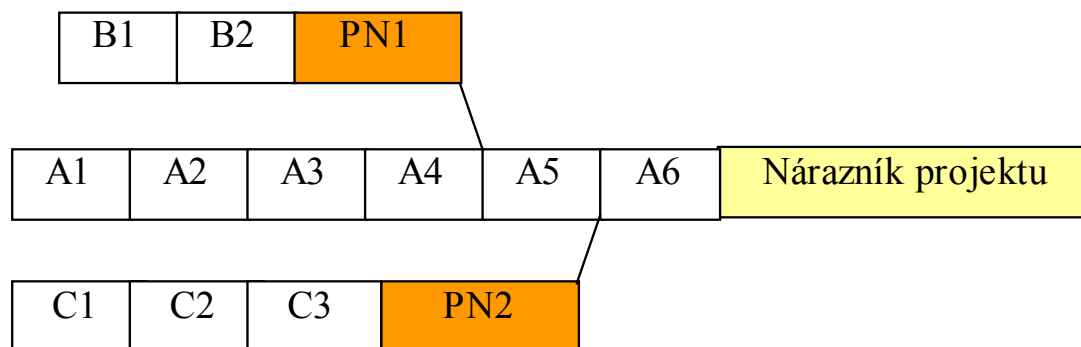
$$\underline{\text{Délka nárazníku}} \approx (3\sqrt{\sum \sigma_i^2})$$



# PřípojnÉ buffery



Obr. KC2. Kritická cesta s nárazníkem projektu. A1, .. , A5 je kritická cesta



Obr. KC3. Projekt s přípojnými nárazníky PN1 a PN2

# Práce s přípojnými buffery

- Předpokládejme, že návaznost činností v projektu tvoří více lineárních úseků, které se postupně spojují. V tom případě najdeme kritickou cestu. Připojíme za ní nárazník projektu jehož velikost je dána rezervami činností na kritické cestě. Činnosti mimo kritickou cestu zobrazíme jako větve stromu (obr. KC2). Aby činnosti mimo kritickou cestu a projekt se dal lépe řídit je žádoucí doplnit nárazníky délky vypočtené výše uvedeným způsobem pro každou postranní větev. Tyto nárazníky nazveme přípojně nárazníky
- Při řízení projektu se sledují pro každou větev výše uvedeným způsobem přípojně nárazníky a nárazník projektu. Pokud přípojný nárazník nestačí (je vyčerpán) zkrátí se i nárazník projektu.

# Soutěž o zdroj X

- Klasická metoda kritické cesty nedostatečně zvažuje případ, kdy je nějaká činnost X prováděna na více větvích, nebo dokonce ve více projektech (příkladem mohou být kontrolní nebo dokumentační činnosti). Řešení tohoto problému je poměrně komplikované, dobře pracuje následující přiblížení.

# Soutěž o zdroj X

- a) Činnosti X se považují za činnosti na kritické cestě a proto přispívají standardním způsobem ke zvětšení projektového nárazníku (tato cesta se nazývá kritický řetěz).
- b) Před každou činností X se vytváří přípojný nárazník.
- c) Pro činnost X je možné vytvořit frontu prací obsluhovanou v závislosti na termínech navazujících etap v projektu.
- d) Pokud je X úzké místo celé firmy (do značné míry určují výkon firmy) dostávají přednost požadavky těch projektů, pro které má zlomek  $(\text{výnos projektu})/(\text{doba vytížení X})$  maximální hodnotu.
- Komplikovanější řešení používá i plánování kritického zdroje (podle teorie omezení bývá jen jeden).

# Hodnocení

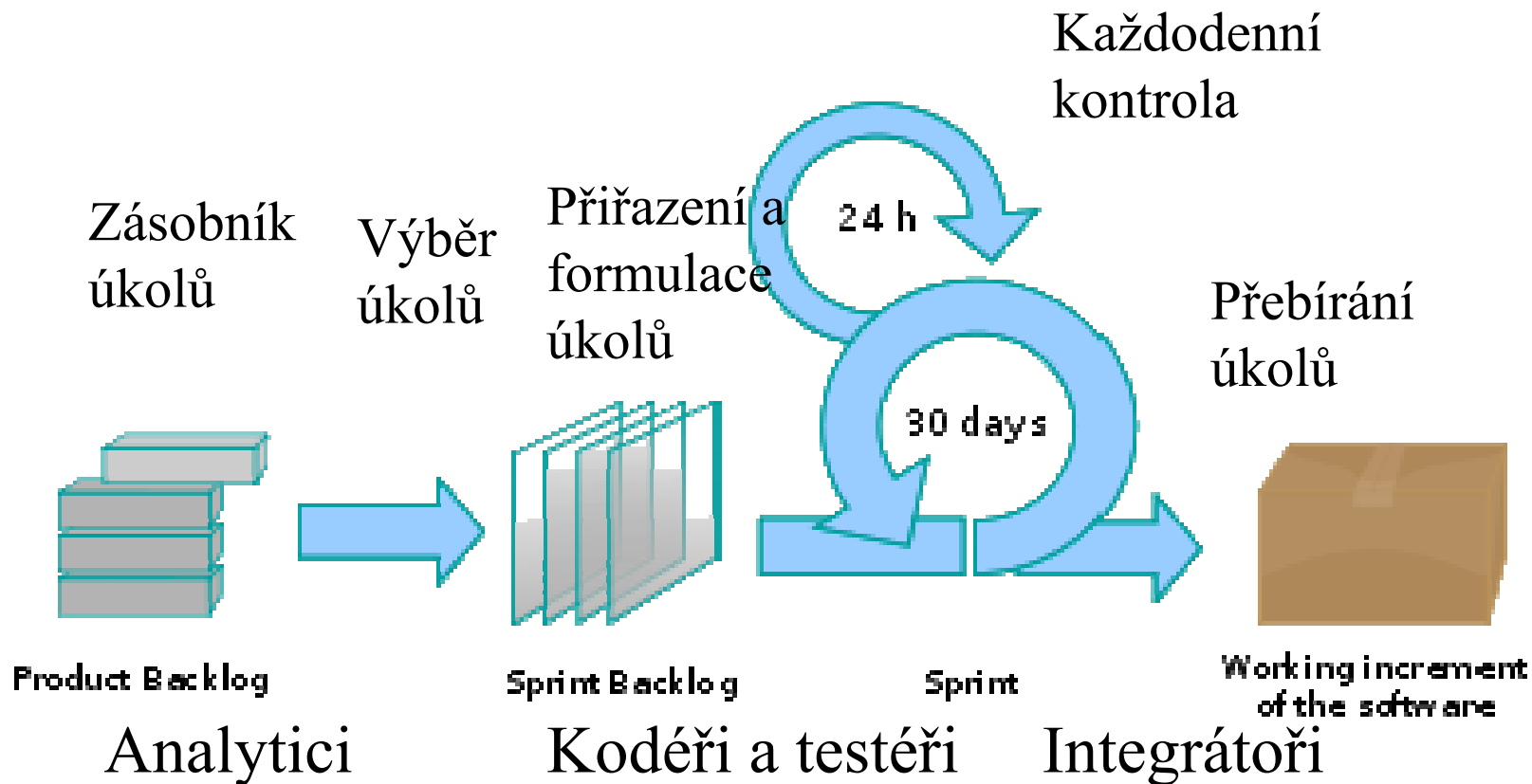
- Mnohé nedořešeno (obecné acyklické grafy sítí činností)
- Osvědčuje se podle dostupných zpráv
- Chtělo by to asi lepší statistické zpracování obecnějších případů
- **Vyžaduje společný zájem všech zúčastněných**
- **Pravidla spolupráce (odhalování rezerv) je možné a rozumné využívat neformálně**

# Hodnocení

- Krásný příklad, jak řešení závisí na kvalitě dat a také na tom, že i pak je řešení závislé na na dobrých vztazích v podniku, jeho kultuře a morálce.
- Považují-li lidi za onuce, nemohu čekat dobré výsledky
- Psychologický kapitál může být zatraceně významný
- Dobrý vztah k lidem není věc dobročinnosti ale chladného kalkulu
- Kritický řetěz je zaměřen na operativu nikoliv dlouhodobé úkoly

Řešení problémů líný student a  
zpevnování norem kontrolou a  
přechodem od řemeslné k  
manufakturní práci

Dekompozice, přechod k  
velkovýrobním postupům



Je důležité, jak se volí témata a formuluje jejich cíl  
 Míra spoluúčasti uživatelů a programátorů!!!!!!



# Agilní scrum zlepší pravidlo 80/20

- Správně vybraný úkol sprintu umožní podstatně zlepšit analýzu a tedy i kvalitu následných sprintů, to je věc kterou Pareto neuvažoval, pravidlo 80/10 zvažuje jen řazení již pevně daných úkolů
- V SW asi lze dosáhnout vztahu 90/10, ale za podmínky správné formulace vizí.

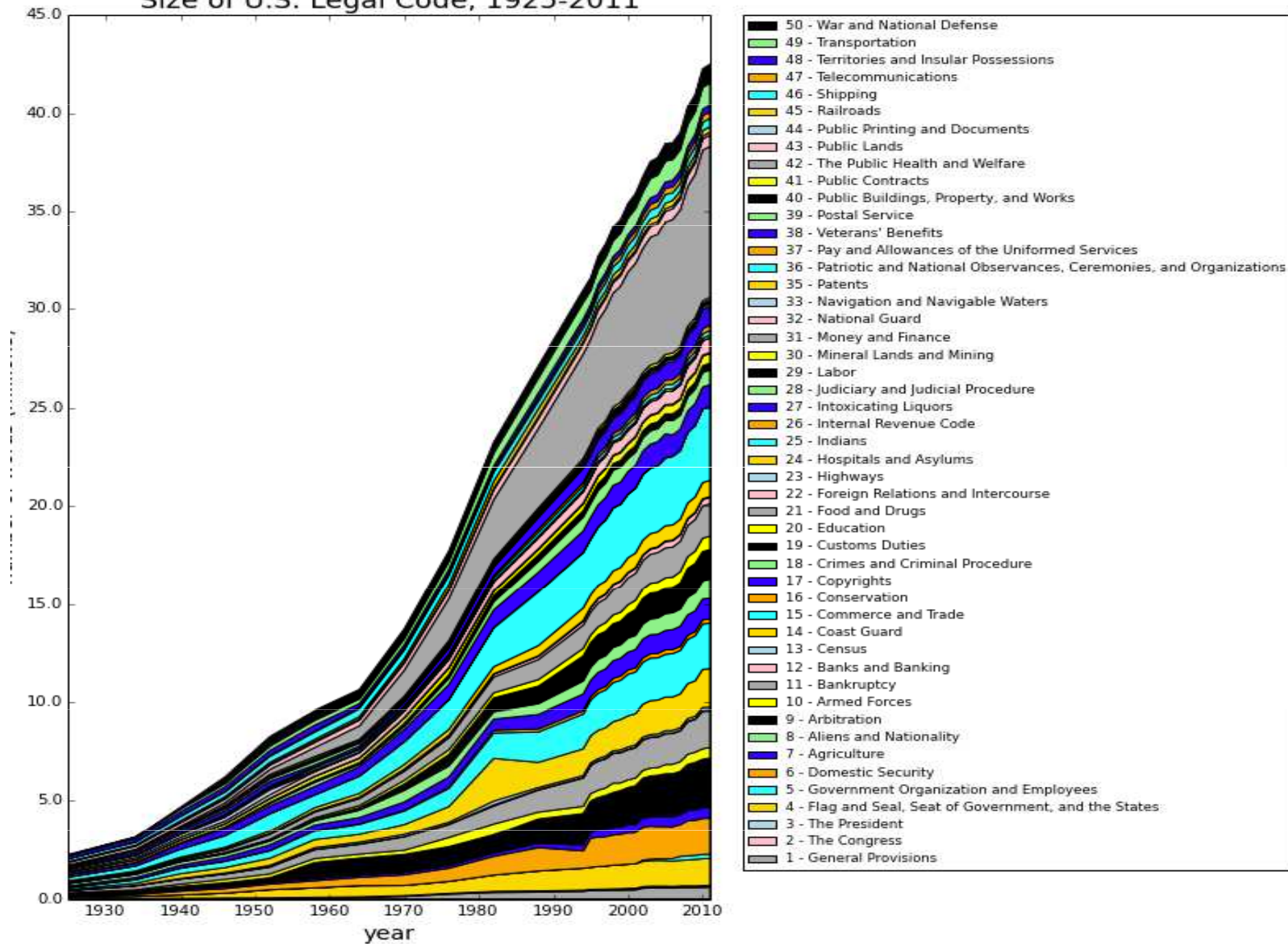
Růst rozsahu a složitosti norem  
má obdobu textech zákonů

- *Law is Code: A Software Engineering Approach to Analyzing the United States Code*
- William Li,\* Pablo Azar,\* David Larochele,\* Phil Hill,\* and Andrew W. Lo\*
- This Revision: September 21, 2014

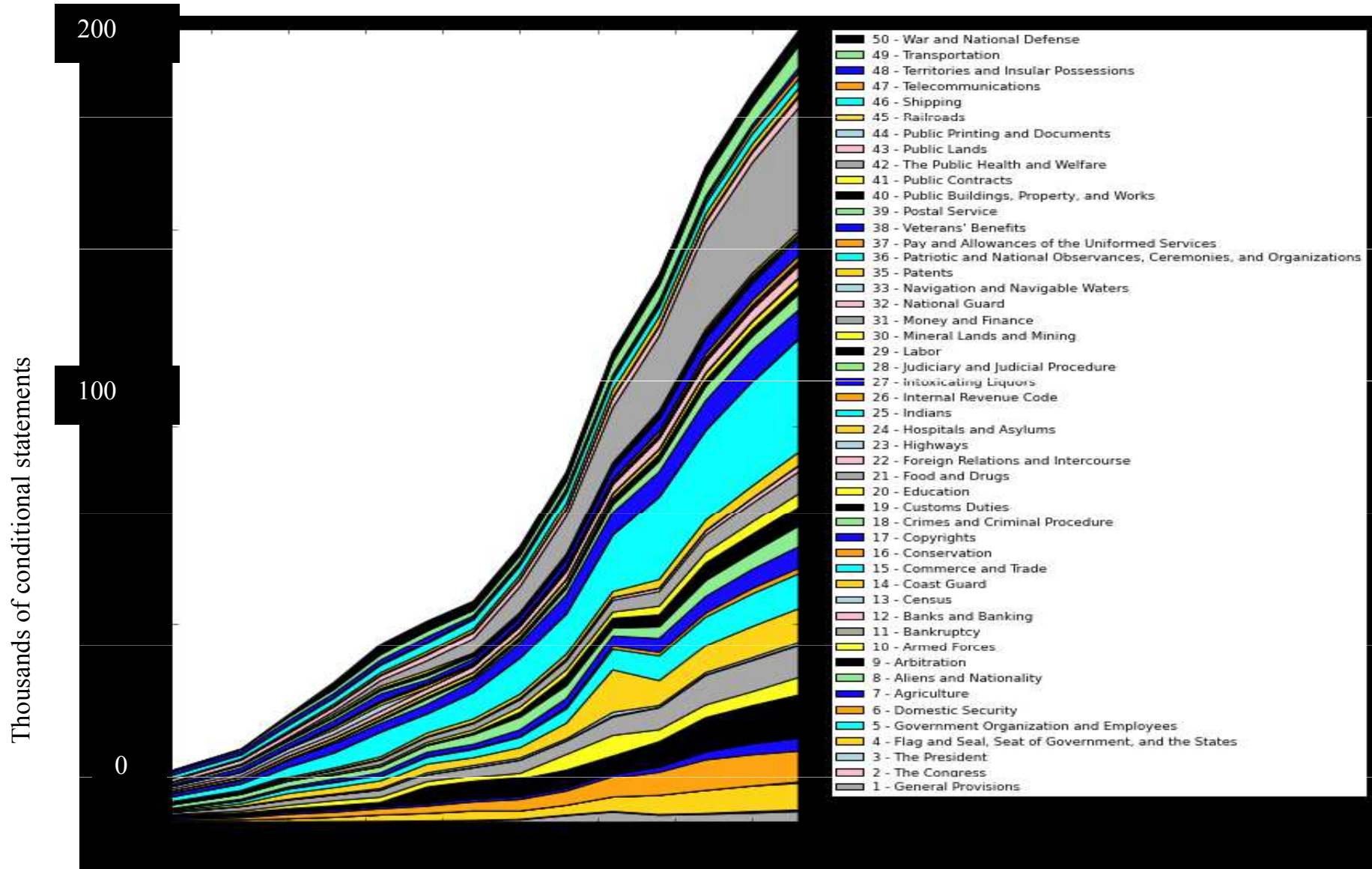
Výstup projektu a iniciativy SSRN

# Možný průšvih, viz SSHD projekt

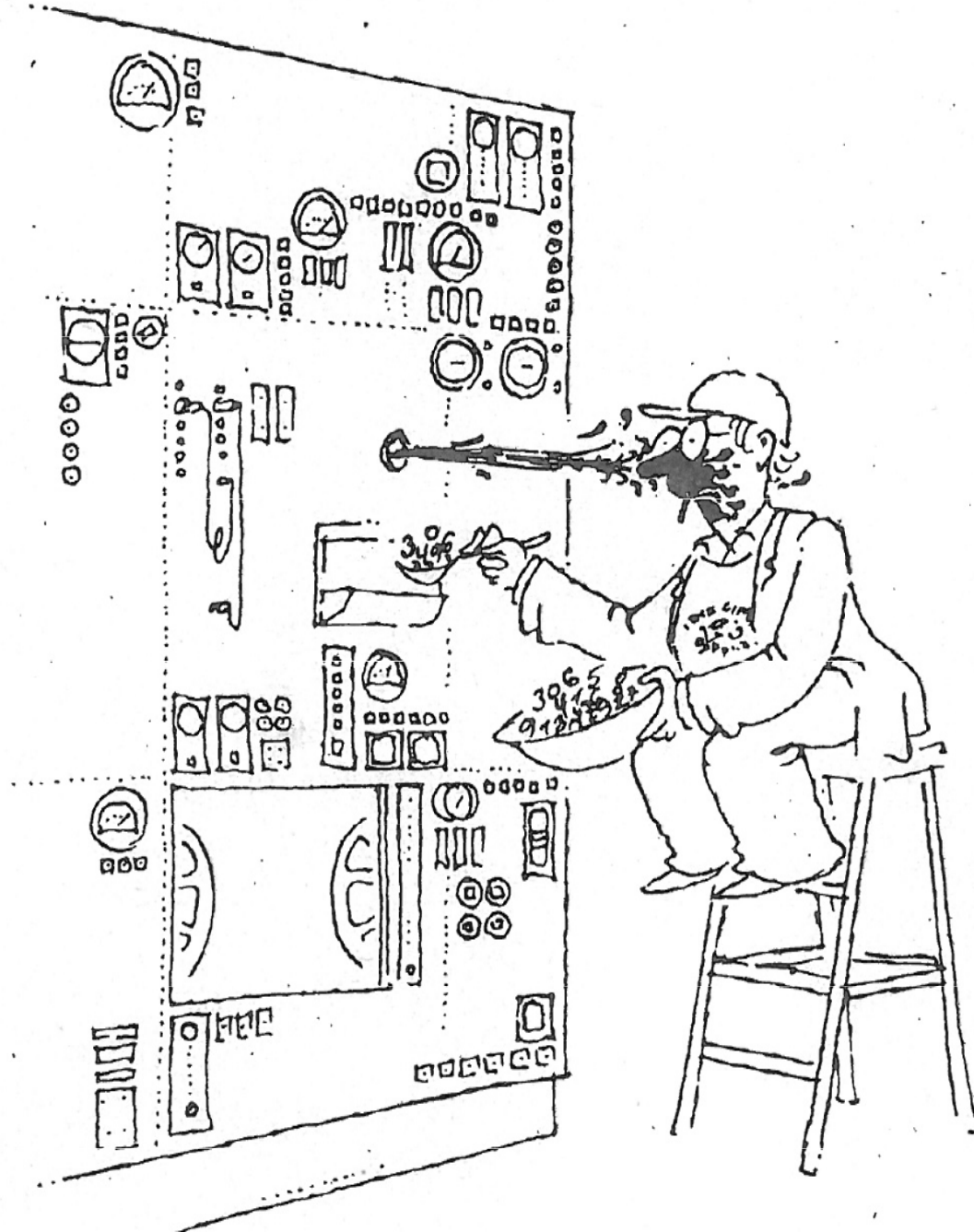
Size of U.S. Legal Code, 1925-2011



# Metrika McCabe pro legislativu USA

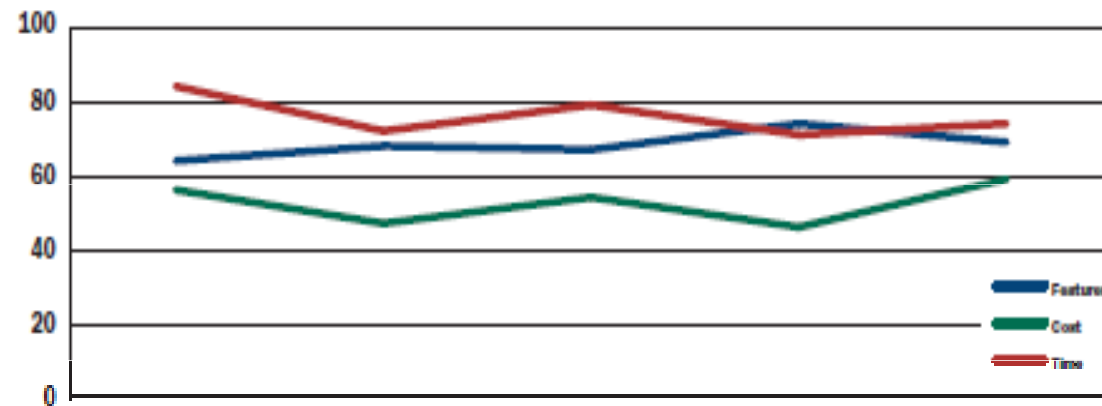


Pokrok ve formulaci cílů a obecně  
analýze není uspokojivý  
**STÁLE JE AKTUÁLNÍ**  
**NÁSLEDUJÍCÍ KARIKATURA**



## OVERRUNS AND FEATURES

Time and cost overruns, plus percentage of features delivered from CHAOS research for the years 2004 to 2012.



	2004	2006	2008	2010	2012
<b>TIME</b>	84%	72%	79%	71%	74%
<b>COST</b>	56%	47%	54%	46%	59%
<b>FEATURES</b>	64%	68%	67%	74%	69%



2013

## FACTORS OF SUCCESS FOR SMALL PROJECTS

Factors of Success	Points
Executive management support	20
User involvement	15
Optimization	15
Skilled resources	13
Project management expertise	12
Agile process	10
Clear business objectives	6
Emotional maturity	5
Execution	3
Tools and infrastructure	1

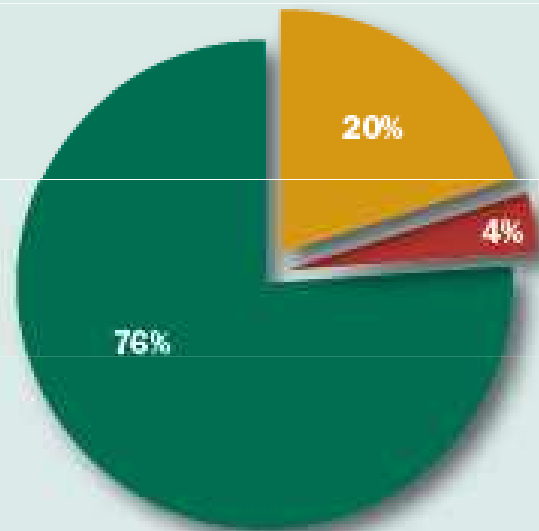
2013

2013

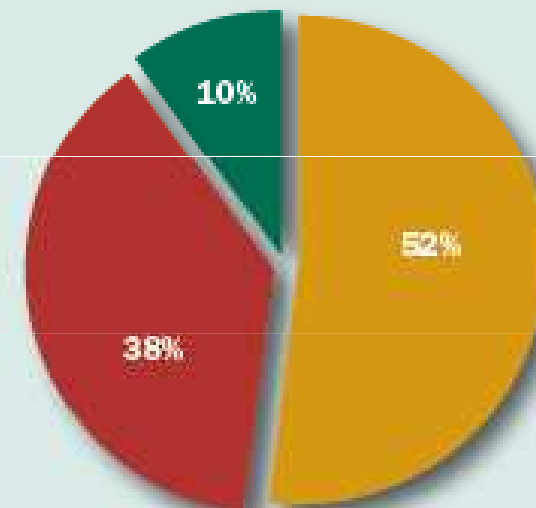
## CHAOS RESOLUTION BY LARGE AND SMALL PROJECTS

Project resolution for the calendar year 2012 in the new CHAOS database. Small projects are defined as projects with less than \$1 million in labor content and large projects are considered projects with more than \$10 million in labor content.

Small Projects



Large Projects

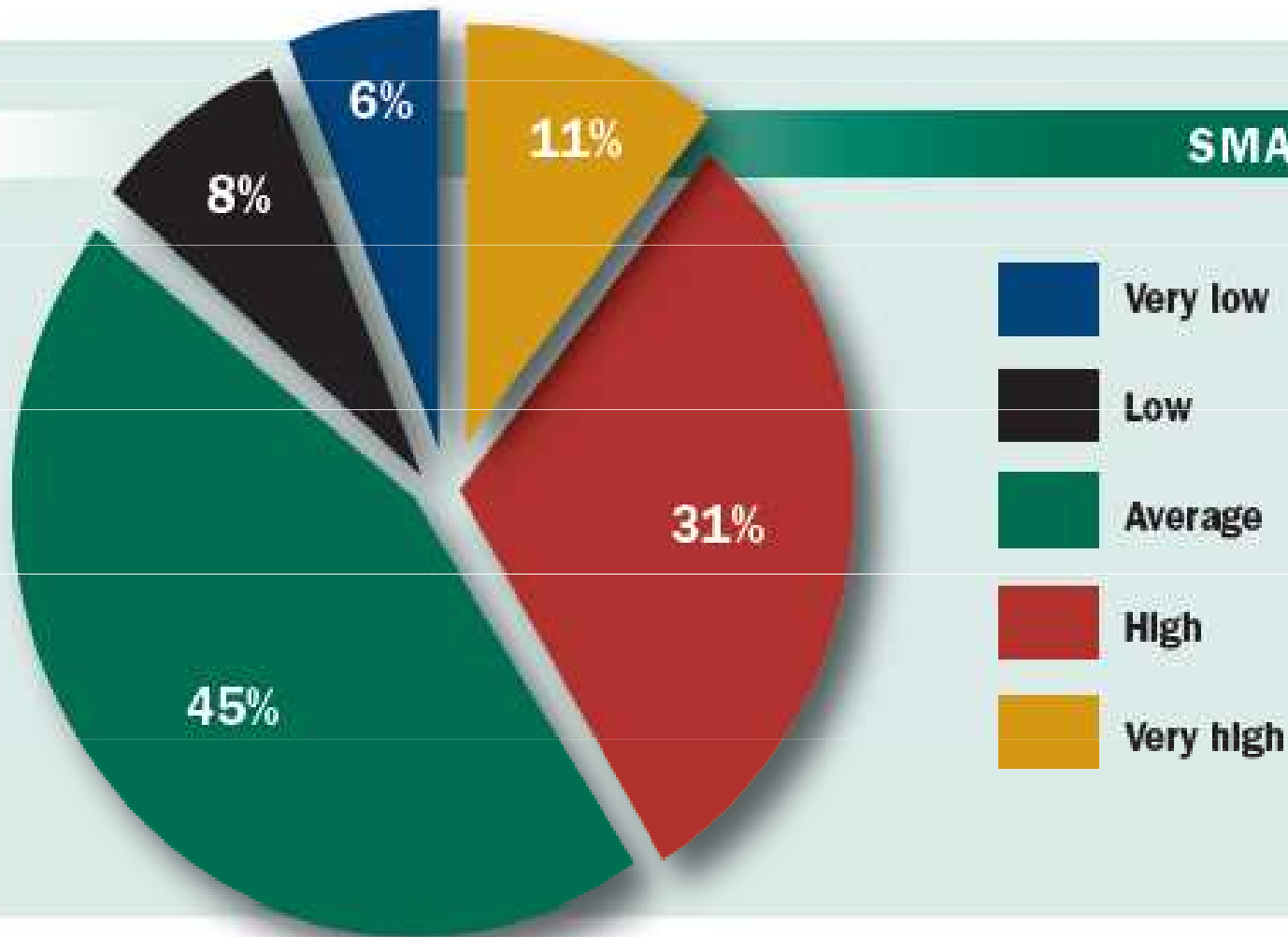


## SKILL LEVEL OF THE EXECUTIVE SPONSOR

	Highly Skilled	Skilled	Moderately Skilled	Poorly Skilled
In understanding what Inspires the project team	11%	62%	18%	9%
In celebrating large and small accomplishments	9%	47%	27%	16%
In recognizing team member contribution – big and small	8%	59%	17%	16%
In rewarding someone of outstanding effort	12%	59%	8%	21%
In giving a celebration party	6%	44%	27%	22%

Respondents were asked to rate their IT management's skill in influencing and educating the project executive sponsors on the listed skills.

## SMALL PROJECT VALUE

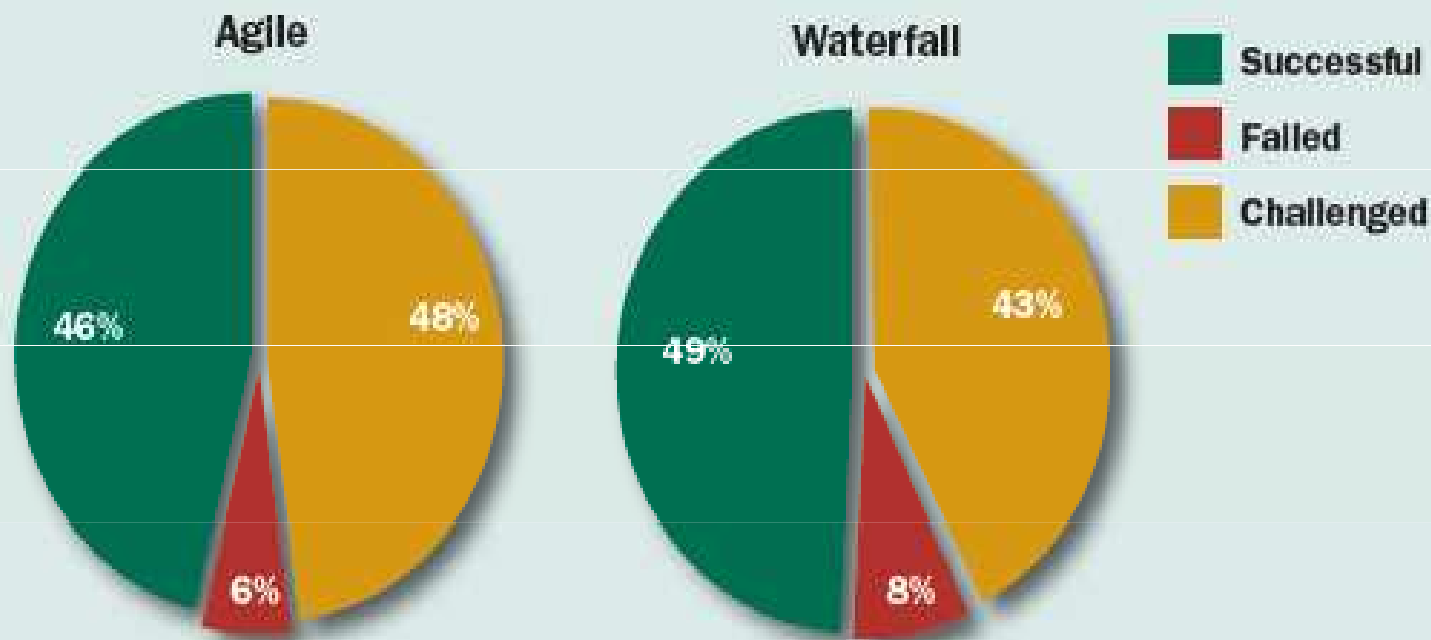


Reported  
value of small  
projects for  
calendar year  
2012.

# Lidé se často rozhodují neracionálně

- Podvědomé obavy
- Lenost
- Chybějící znalosti
- Podceňování složitosti úkolu (GAČR)
- Snaha se někomu pomstít (boj proti ponorkám za války, raději bombardovat a to ještě ne rafinérie, v míru jsou podobné případy)

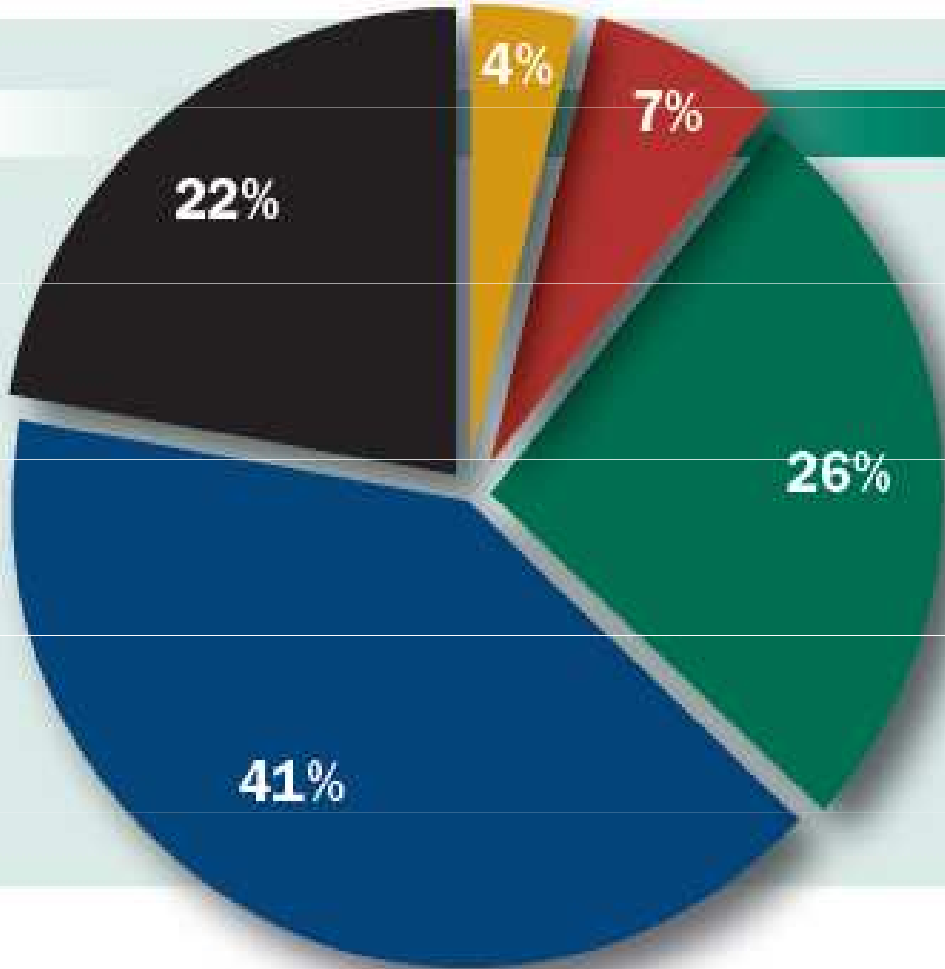
## AGILE V. WATERFALL SMALL PROJECTS



The charts show success rates for small software development projects using modern languages, methods, and tools, from 2003 to 2012.

U malých projektů agilita moc nepřináší  
Hypotéza – nelze je rozumně dekomponovat

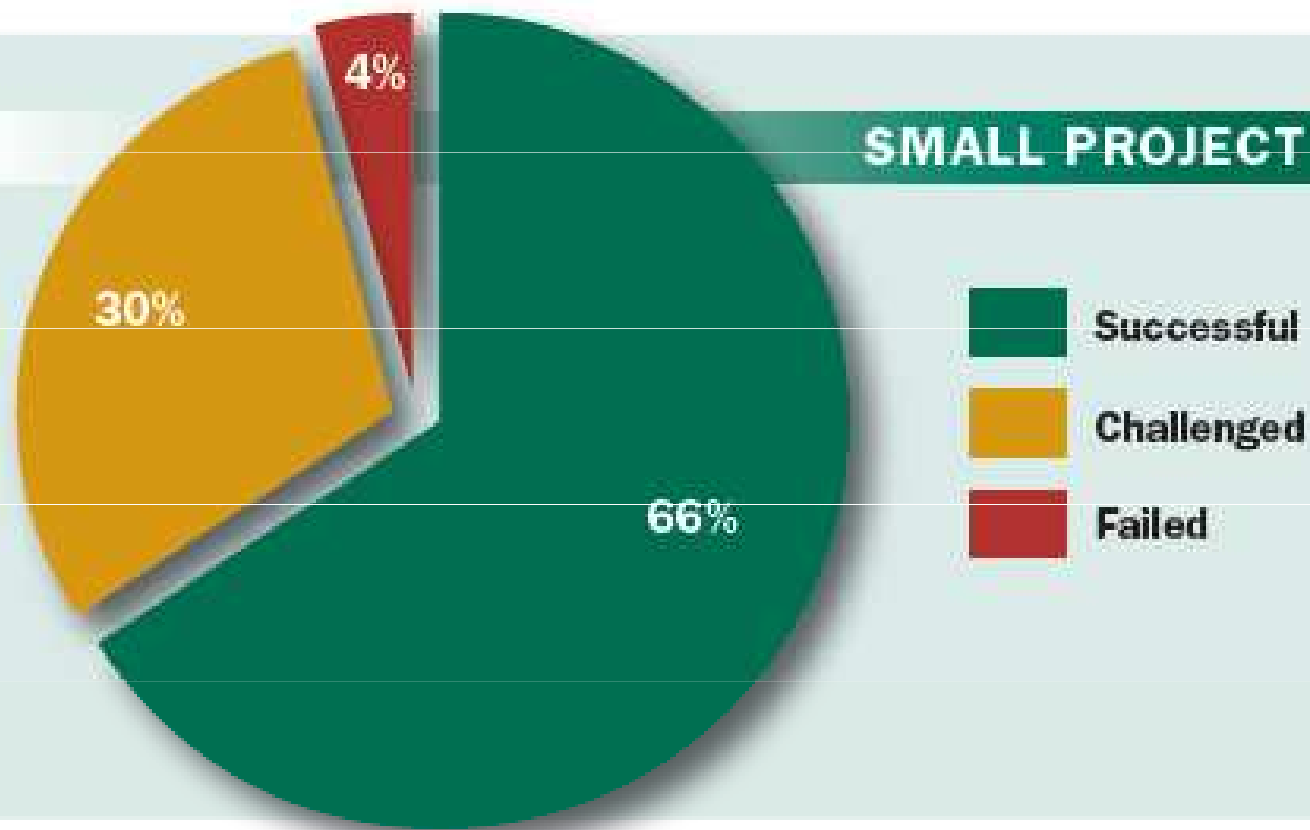
### SMALL PROJECT GOAL



-  Distant
-  Vague
-  Loose
-  Close
-  Precise

The chart shows reported goals of small, successful projects for calendar year 2012.

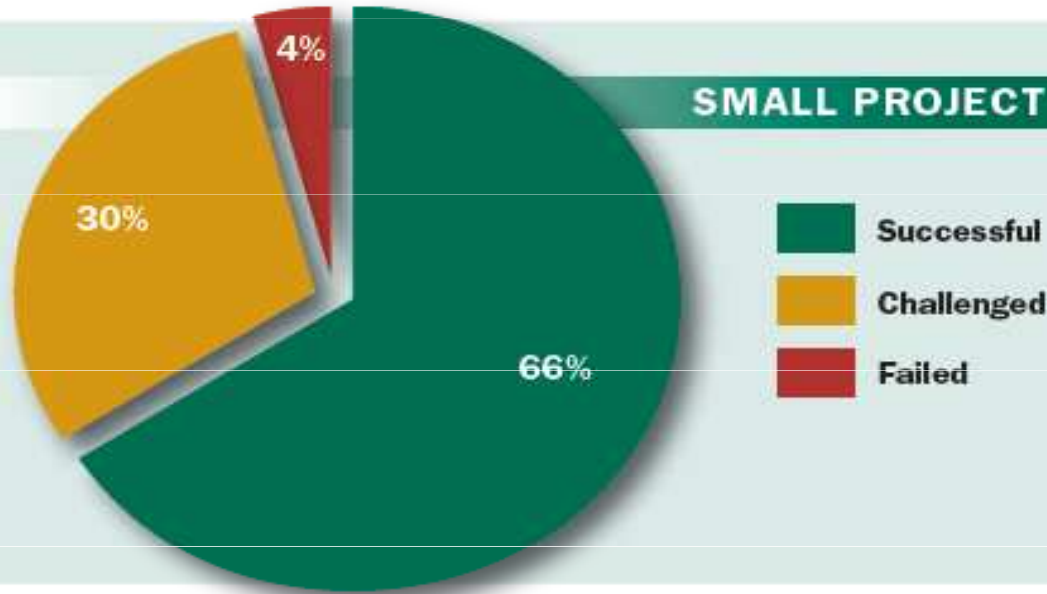
## SMALL PROJECT RESOLUTION IN BANKING



The chart shows small project resolution in the banking industry, from 2003 to 2012. A small project is a project with less than \$1 million in labor cost.



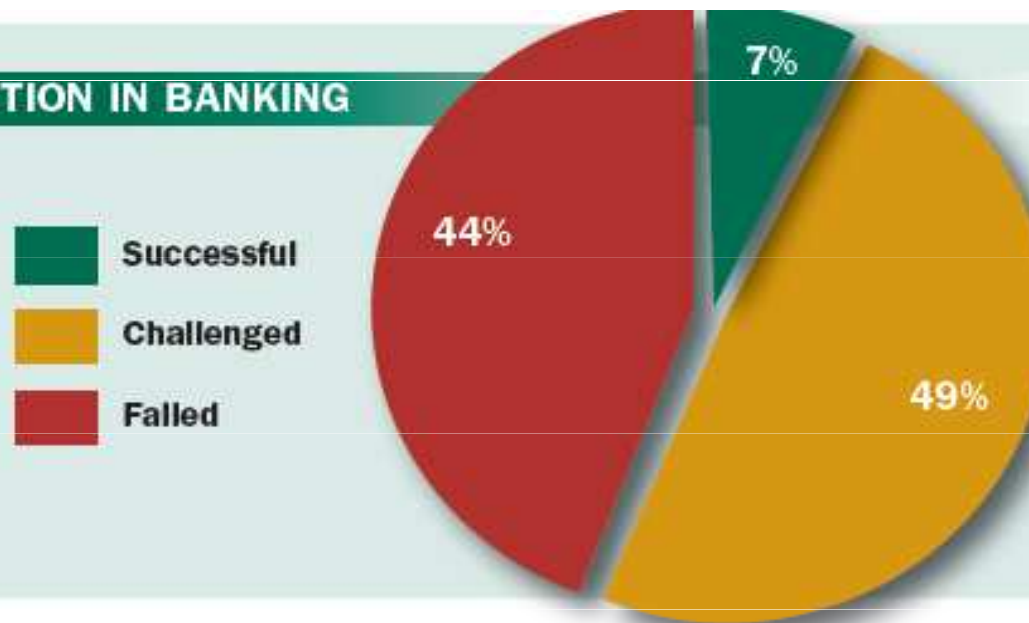
### SMALL PROJECT RESOLUTION IN BANKING



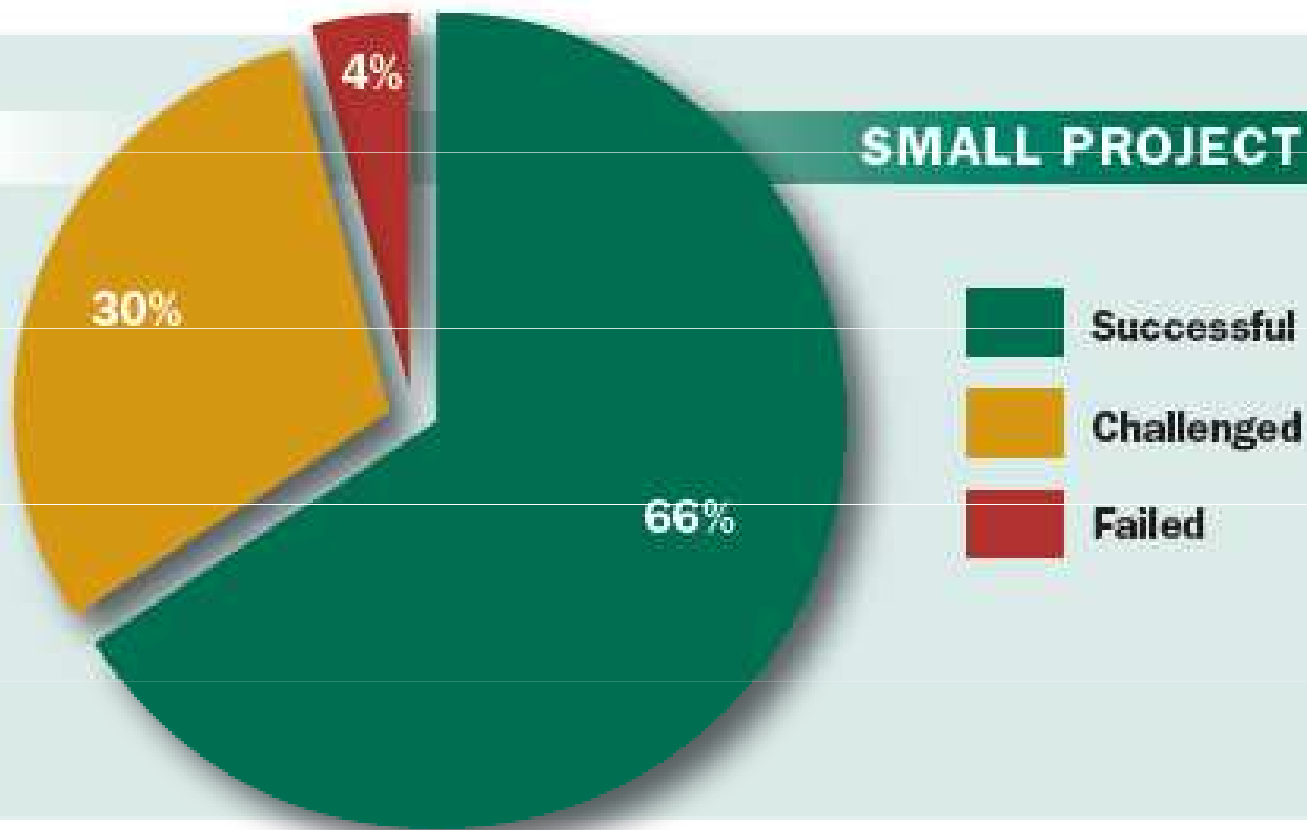
The chart shows small project resolution in the banking industry, from 2003 to 2012. A small project is a project with less than \$1 million in labor cost.

### LARGE PROJECT RESOLUTION IN BANKING

The chart shows large project resolution in the banking industry, from 2003 to 2012. A large project is a project with more than \$10 million in labor costs.

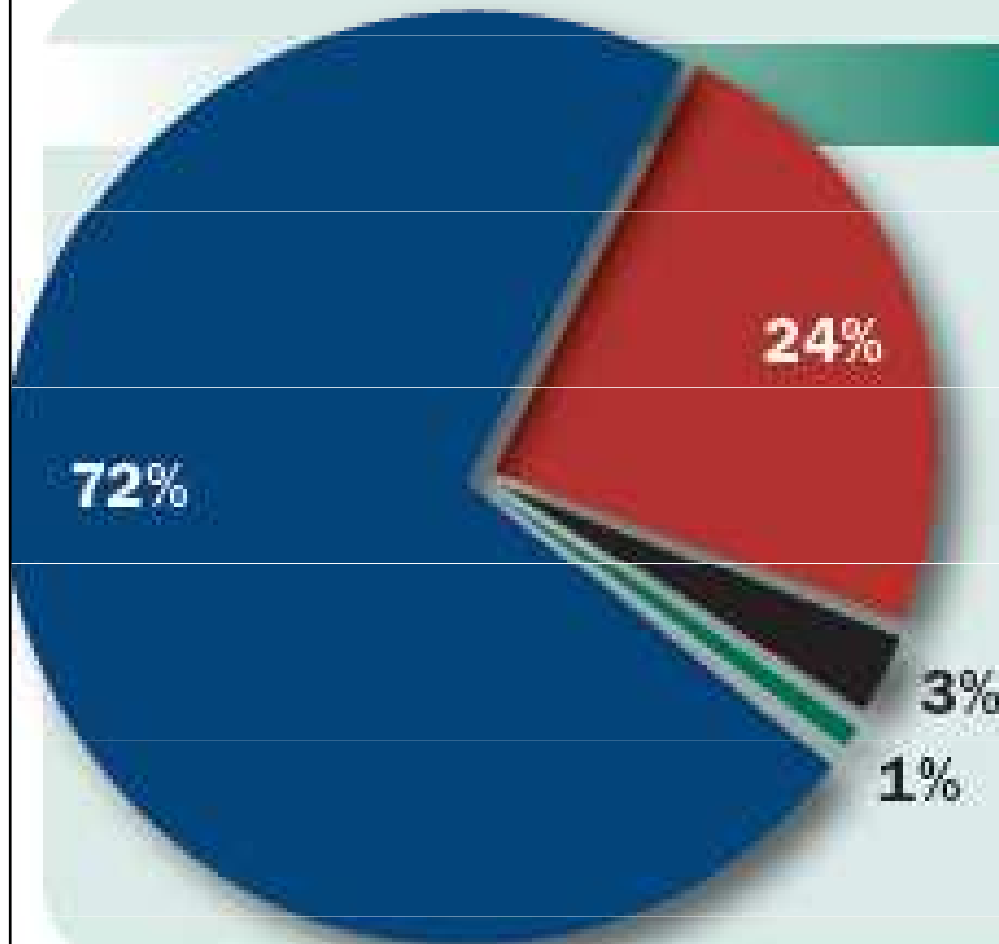


## SMALL PROJECT RESOLUTION IN BANKING



The chart shows small project resolution in the banking industry, from 2003 to 2012. A small project is a project with less than \$1 million in labor cost.

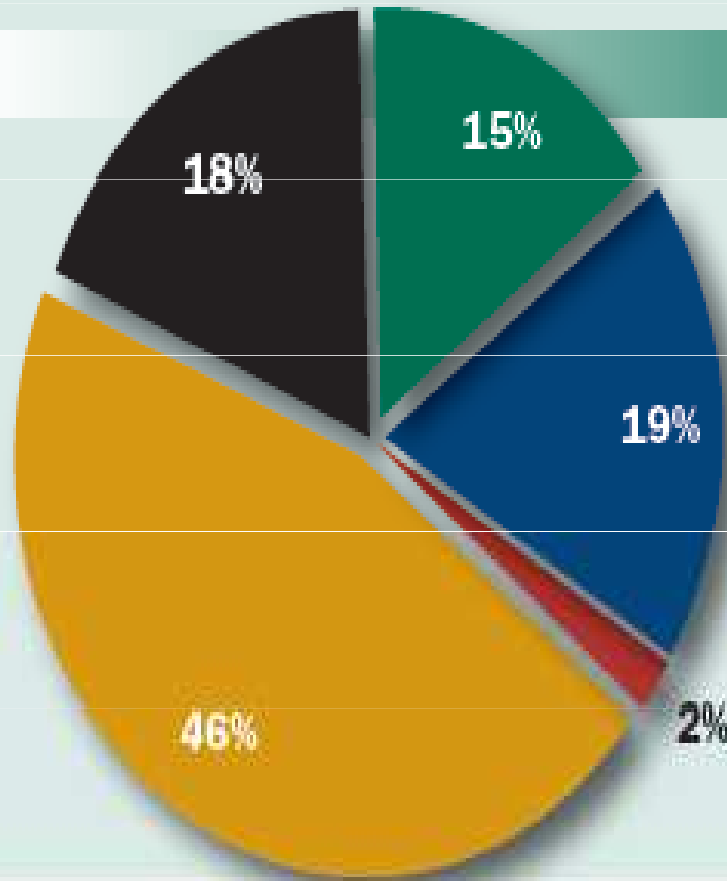
## VALUE ACCURACY



-  Do not know
-  Not accurate
-  Somewhat accurate
-  Accurate

IT executives were asked how accurate the value evaluation is.

## POST-PROJECT VALUE EVALUATION



- Never perform a post-project value evaluation
- Rarely perform a post-project value evaluation
- More than three years in use
- One to three years in use
- After the projected payback period

IT executives were asked how long after a project has been completed, implemented, and is being used to gain value, do they generally evaluate the stated value to see if the project met its value goals.

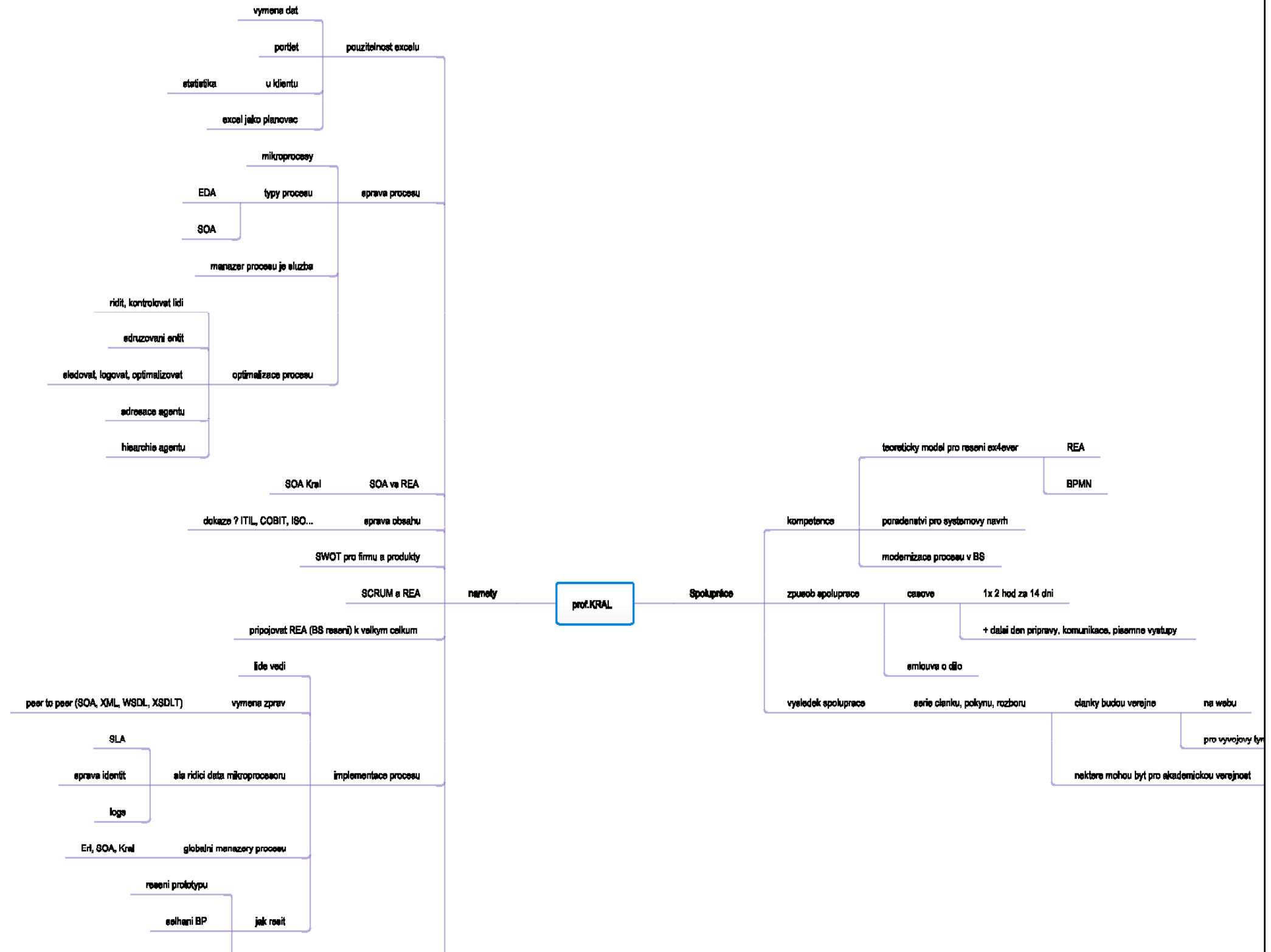
# Mindmap (use cases)

Sledování příčin a následků

Sledování problémů a opatření

Mnoho i freeware řešení

[www.Mind42.com](http://www.Mind42.com)



pouzitelnost excelu

mikroprocesy

typy procesu

sprava procesu

manazer procesu je sluzba

optimalizace procesu

SOA Kral

SOA vs REA

dokazy ? ITIL, COBIT, ISO...

sprava obsahu

SWOT pro firmu a produkty

SCRUM a REA

pripojovat REA (BS reseni) k velkym celkum

lide vedi

vymena zprav

implementace procesu

globalni manazery procesu

reseni prototypu

jak resit

teoreticky model pro reseni ex4ever

REA

BPMN

paradenstvi pro systemovy navrh

modernizace procesu v BS

způsob spoluprace

časove

1x 2 hod za 14 dni

+ dalei den pripravy, komunikace, pisemne vystupy

emlouva o dilo

vysedek spoluprace

serie clenku, pokymu, rozboru

clanky budou verejne

na webu

pro vyvojovy tym

nektere mohou byt pro akademickou verejnost

prof.KRAL

namety

Spoluprace