

PA105/2017

Management a ergonomie informačních systémů
v malých a středních organizacích
(small to medium enterprises – SME)

Malé a střední podniky

Small to medium enterprises (SME)

- Z hlediska podnikové kultury jsou malí asi do dvaceti ažtááků.
- Střední do dvou set ažtááků
- Hranice závisí na tom, co podnik dělá a na kvalitě jeho zaměstnanců.

SME

- Mnoho (více než 1/3) lidí pracuje v nějakém menším podniku
- SME – podniková kultura, klíčové inovace a omezení pro podrobné rozpracování procesů
- Mnohé technologie vyvinuté v SME jsou postupně používány mimo SME a jsou předobrazem budoucnosti ITC
- Z malých občas vyrostou velmi velcí
 - Facebook, Google, HP
 - Simplity
- SME a moderní architektury

Malí a velcí a IS

- Malý dodavatel SW a malý odběratel
 - Typická oblast uplatnění malých SW firem
 - Comodity, open source, vlastní vývoj, dobré nápady i při rizicích
- Malý dodavatel – velký odběratel
 - Subdodávky pomocí „přílepků“
 - Opencard, žádanky, přístup k analýza dat (Simplify)
 - Z velkého systému se pomocí změny architektury stane malý
 - asi Mýto, jistě Opencard, účast na inkrementálním vývoji

Velcí dodavatelé SW a malí uživatelé

- Nejčastěji poskytovatelé SW komodit
 - datové báze, řízení, standardní operační systémy a normy
- Problém s rozdílností podnikových kultur
 - SAP, Oracle
- Problém s nákladností řešení, konfekce za vysokou cenu
 - SAP

Hlavní témata

- SW procesy, užitečná pravidla jak a zda zahájit projekt
- SW architektury a dokumentová orientace – nové příležitosti pro SME
 - Proč to nejde ve státní správě
- Kvalita dat a kvalita sw systémů
- Manažerské aspekty řízení rizik, časté prohřešky
- Týmová spolupráce a základy personalistiky
- Efekty investic do nepřímých přínosů (do ekosystému), ergonomie SW a psychologický kapitál kontrola hygieny
- Řízení pozdních etap životního cyklu
- Použitelnost, uživatelské rozhraní a management,
- SW metriky odhady a plánování
- SW normy a jejich výhody a nebezpečí

Zvláštnosti IT, hlavně IS

- Prudký vývoj
 - Za pět let pokrývají současné znalosti jen 50 procent potřebných znalostí
 - Naučit se získávat nové znalosti
 - Moorův zákon (vzhledem k paralelizaci vlastně stále platí)
 - Zdesateronásobení výkonu (běžné sítě) procesorů za pět let
 - Za 50 let 10^{10} zvýšení kapacit, místo 320 let jedna sekunda
 - Nové možnosti a nové požadavky
 - Zábava, sociální sítě
 - Správa, vedení podniků, přístup k informcím
 - Ochrana i hrozby
 - Každých deset let nové paradigma
 - , DFD, Objekty, Komponenty, SOA, otevřené systémy, REA
- Manažerské aspekty jsou stabilní, manažeři hlavní hrozba
 - Procento krachujících projektů se dlouhodobě nad 25

Tempo zastarávání

- V technologii vývoje SW je 50% používaných znalostí mladší pěti let
 - neplatí to pro principy vývoje
- V analýze je tempo změn mírnější
 - Lidé se mění pomalu
 - Kamkoliv pohlédneš, všude jsou lidé stejní pitomci
 - Nápis na sumerské tabulce z Uru (hlavní město sumerské říše ve druhém tisíciletí před Kristem, podle Bible rodiště Abraháma)
 - Nemá smysl se učit všechny systémy, klíčové ale ano, učit se zvládat nové, umět to s lidmi

Co se relativně málo změnilo

- Lidé
 - Postoje
 - Cíle
 - Předsudky
- Potřeba a dovednost spolupráce
- Nedostatečné chápání ICT jako technologického oboru
- Manažerské nedostatky

Zkušenosti přednášejícího

- 1959 Statistik se stává programátorem ve assembleru, výpočty pro fyzikální chemii, pitomí politici – nepustíme ani jeden transistor do výroby počítačů, když máme dost relé
- 60-70 Přejít na Fortran, Algol (pokús o kompilátor), formální jazyky, některé mé články z té doby se dodnes citují, mainframy
- 70-80 on line vstupy, minipočítače v přímém řízení systému dílny, první aplikace principu výměny zpráv mezi SW komponentami
- 80 účast na vývoji českých PC, spolupráce se Siemensem na řízení rozsáhlého výrobního systému,
- 90 Soukromý podnik, počítačové sítě a IS , nová varianta analýzy shora, dekompozice IS, počátek www Autonomní komponenty a SOA
- 2000 SOA, patterns, účast na projektech malých firem, konfederativní SOA, práce v týmu , dokumenty řízené systémy, role osob v SW systémech

Hlavní poznatky

- Krize IT se vrací asi po deseti létech, může přijít znovu
- Po krizi se vždy se vrátil zájem o informatiku, ale s podstatně jinými úkoly-paradigmaty
- O české informatiky je stále zájem, především o kódéry a návrháře
- Nově globalizace vývoje SW, hlavně v kódování
- Mnoho lidí funguje jako servis běhu SW systémů
- Vývoj od začátku není častý, hlavně v menších firmách
- Účast na vývoji velkých firem v jejich pobočkách
- Mnoho lidí programuje podporu e-governmentu

Podceňované vlastnosti IT

Efekty IT, především IS se obtížně měří, často jsou jinde než se čekalo (uvedeme příklady)

Jsou často dlouhodobé a ty jsou jiné než krátkodobé

Někdy není ochota je uplatnit (př. školství)

Praxe ochrany dat neumožňuje data správně využívat aniž zajišťuje dostatečnou ochranu dat

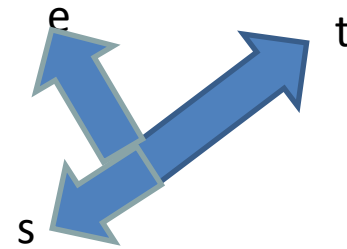
Kvalita IT je tedy věcí kvalifikovaného používání, znalostí, zkušeností a někdy i (politické) vůle a boje s předsudky

Budeme využívat zkušenosti z (starších i současných) praktických projektů pro malé a střední firmy a nedávné výsledky výzkumu metod SW inženýrství v oblasti SOA metod integrace velkých komponent do velkých systémů s využitím dokumentové orientace

Obtížnost aspektů IT, hlavně IS

- Klasické IS, např. správa skladu (operativa)

- Ekonomický užitek e je dosti zřejmý
- Jak technicky udělat t také,
- Sociální aspekt s málo ovlivňuje řešení

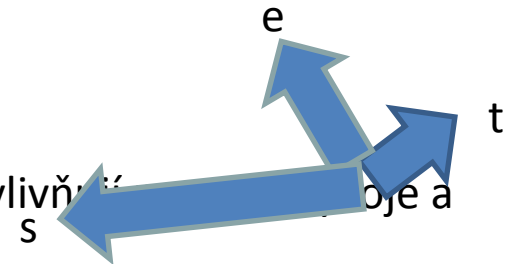


- Informační technologie dnes

- Ekonomický užitek se dostatečně nezvažuje
- Jak technicky udělat: řešení ad hoc, předražené, architektury
- Sociální aspekt téměř vždy přítomen, mnohdy zcela zásadní

- Je zdrojem obtížnosti a problémů
 - Nejasnost cílů
 - Postranní zájmy (viz evaluaci škol), korupce
 - Potřeba globálních propojených měnících se systémů
 - Existence sociálních a zdravotních rizik

- Ekonomické a sociální aspekty zásadním způsobem ovlivňují řešení a musí se zvažovat při specifikacích požadavků



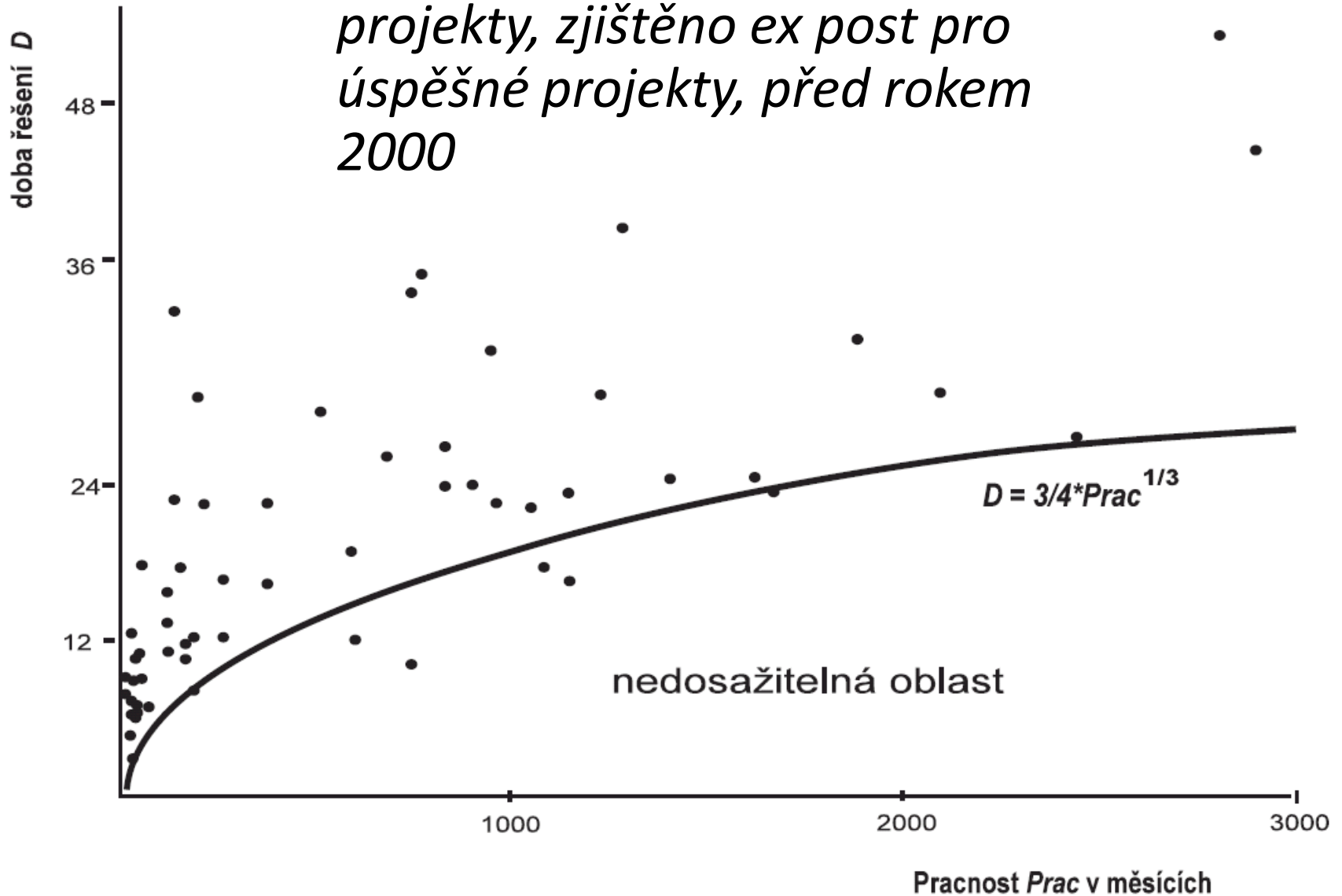
Úzká místa informatiky

- Zaměříme se na úzká místa vývoje IS , bez jejich vyřešení nedosáhneme úspěchu
 - specifikace a spolupráce s uživateli, společenské souvislosti
 - Úkoly při řízení projektů, problém sledování úkolů pomocí smart sheetu
 - SW procesy především s aplikacemi na SOA
 - SW normy a SW metriky
- Něco se dá jen obtížně na škole nacvičit, lze jen seznámit (MBA lze studovat až po získání praktických zkušeností)
 - Spolupráce s uživateli
 - Tvorba rozsáhlých systémů
 - Management, zvyky ve firmách
 - Dovednosti a zkušenosti,
 - Pokus o otestování v PV024

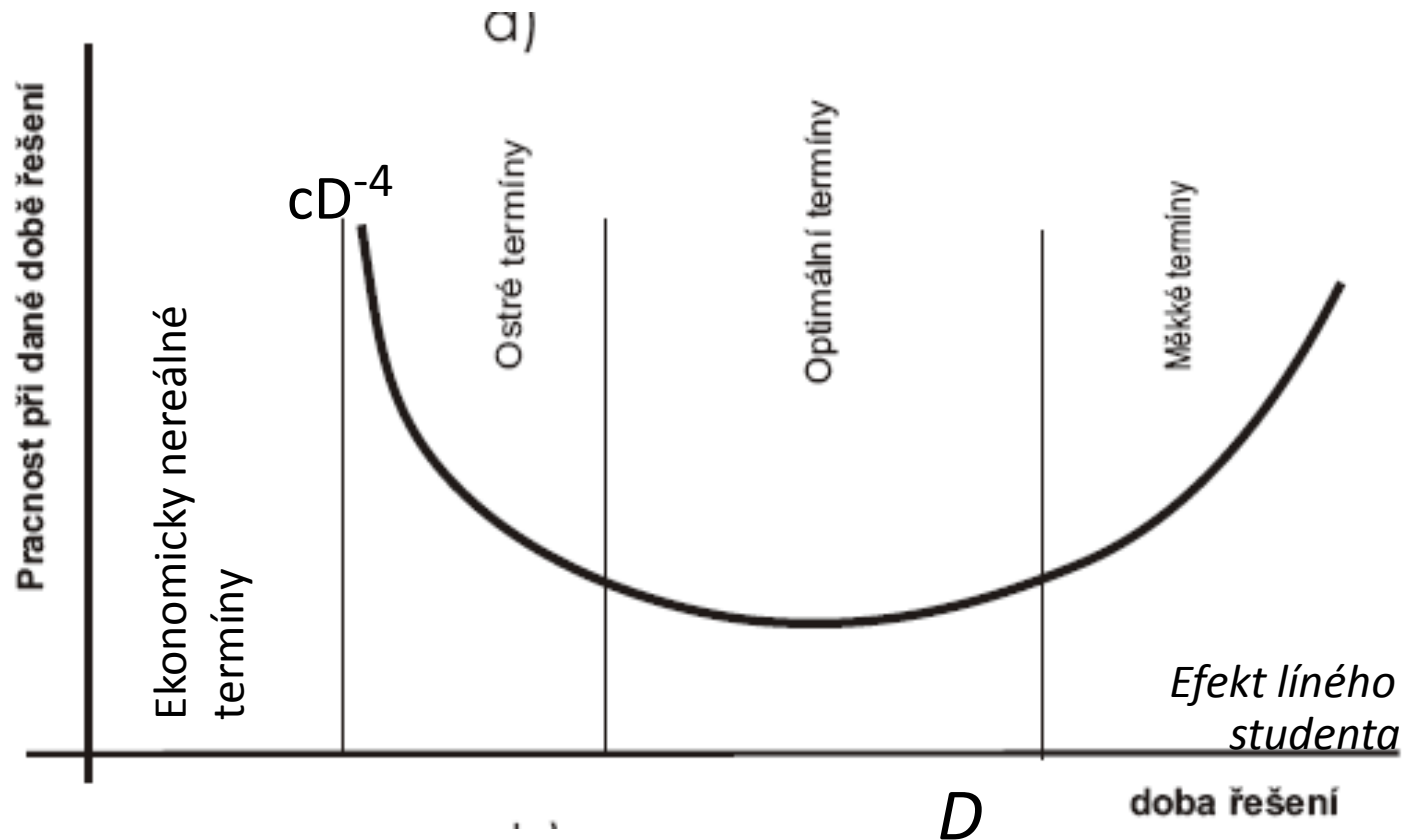
Tempo zastarávání znalostí v IT,

- V technologiích vývoje SW se inovuje 50% znalostí za 3-5 let
 - Systém může zastarat dříve, než se dokončí jeho vývoj
 - Doba vývoje závisí na velikosti systému a nedá se libovolně zkracovat ⇒ dilema stálého zastarávání
 - Budeme hledat řešení v servisních systémech složených z AUTONOMNÍCH ČÁSTÍ, služeb
- Každých asi deset let dochází ke změně paradigmatu (struktury, objekty, cloudy, služby...)
 - Zaměříme se na věci, které mají šanci přežít
 - Inkrementálnost vývoje, zapojení uživatelů, otevřenost

*Některé starší výsledky pro SW
projekty, zjištěno ex post pro
úspěšné projekty, před rokem
2000*



Pracnost projektu při zkracování termínů, nedosažitelné oblasti



Co je typické pro malé (SW) firmy

- Rychlé změny
 - Diktuje je trh, malá tržní síla firmy – nemůže si dupnout
 - Úkoly i metodiky se rychle mění v důsledku požadavků uživatelů a změn vývojových ekosystémů
 - Menší firmy jsou pružnější obecně
- Nejsou zdroje:
 - Nelze nasadit mnoho specialistů, nejsou k dispozici
- V malých a středních SW firmách se snáze uplatní superprogramátoři a nové nápady

Co je typické pro malé SW firmy

- Často nelze použít hotová komplexní řešení
 - Hotová řešení často dělají velcí pro velké, to není nic pro svět malých firem, mnohdy platí i pro normy, mnohé se musí použít znovu
 - Speciální požadavky: ne velké náklady ani dlouhá doba řešení
 - Kupodivu leccos vhodné pro malé firmy se uplatní i v e-governmentu a tam, kde IS silně ovlivňují přímo uživatele (dokumentově orientovaná rozhraní), Viz používání papírových dokumentů

Omezení pro malé výrobce SW

- SME mají omezené zdroje (investice, čas, zkušenosti, ...) a tedy nemohou zpravidla vyvíjet či modernizovat velké a dokonce ani středně velké systémy metodou velkého třesku (*uvidíme, že to pro velmi velké systémy platí i pro firmy s prakticky neomezenými zdroji*)
- Musí se vyrovnat s tím, že jejich zákazníci potřebují ne zcela malé systémy a vyžadují, aby bylo možné některé části svého a systému outsourcovat
- Nové systémy musí být integrovatelné s existujícími systémy
- Musí být snadné je udržovat a při instalaci přizpůsobovat specifickým podmínkám svého zákazníka

Důsledky omezených zdrojů v malých organizacích obecně

- Méně lidí: role musí mít širší škálu úkolů, člověk ale zvládne jen omezený počet pravidel \Rightarrow podnikové procesy nemohou být detailně specifikovány
- Malá uživatelská firma nemá prostředky na nákup komplikovaného systému, a ani nevytvoří podmínky pro jeho používání (zaučení, změny podnikové kultury), totéž platí pro vývojová prostředí SW firem
- Velké IS vytváří velcí pro velké, nejsou tedy vhodné z kulturních důvodů pro malé, malí je nejsou schopni vzhledem k výše uvedeným omezením vyvinout naráz

Řešení omezených zdrojů v SME

- Vyrábět co nejjednodušší systémy,
 - princip maximální možné lenosti
 - K čemu a proč je to třeba
 - Stálé podceňování pracovních úprav
- Přebírat
- Používat otevřený SW a adaptovat ho (ERP5, AGdampiere,..) nebo používat hotová řešení (systémy třetích stran nebo legacy)

Řešení omezených zdrojů v SME

- Nepoužívat Big Bang, nedělat monolitická řešení, používat architektury s autonomními komponentami
 - Moderní servisně orientované architektury (hrubozrnné SOA), dokumentová orientace
 - integrace systémů, přílepky cloud...
 - Inkrementální vývoj a údržba
 - Integrace s existujícími systémy (doplňky),
 - Změny skryté v komponentách
 - Údržba splývá s vývojem

Nadměrná složitost řešení škodí i velkým

- Složitá řešení se ne vždy vyplatí,
- I když velcí výrobci SW na to někdy mají, koncoví uživatelé zpravidla ne
- Normy resp. dokumentace o tisících stránek, která se často mění
- Mění se přístupy, inovace velmi často už během pěti let

Filosofie řešení malých se stává důležitá i pro velké firmy

- Systémy se stávají natolik komplikované, že je ani velcí SW výrobci nezvládají pokud nepoužijí triky malých SW firem. Jinak jsou jejich systémy
 - Příliš drahé
 - Permanentně zastaralé
 - Obtížně udržovatelné
 - Nespolehlivé

IT je technický obor

*Ne zcela si uvědomujeme důsledky toho, že SW má zpravidla charakter nástroje, vývojové prostředky (programovací jazyky) mají daném SW prostředí charakter nástrojů pro výrobu nástrojů, proto mluvíme o **inženýringu***

Důvody, proč se vyvíjejí nové jazyky a jiné zanikají jsou velmi podobné tomu, proč se vyvíjejí nové stroje a nástroje pro ně, viz. spousty kleští u kováře

musí se s ním dobře pracovat při výrobě určitých artefaktů pokud má kovář potřebný talent znalosti a dovednosti, o artefakty musí být zájem

Kovář má objednávky a měl by mít výkresy

Neakceptuje se, že tvorba SW je
inženýrský obor
SW entity jsou průmyslové výrobky

Shody jejich vlastností s vlastnostmi
klasických pokročilých výrobků jsou
významné

Instrukce se neošoupou?

Omyl: Instrukce se neošoupou, software tedy také ne

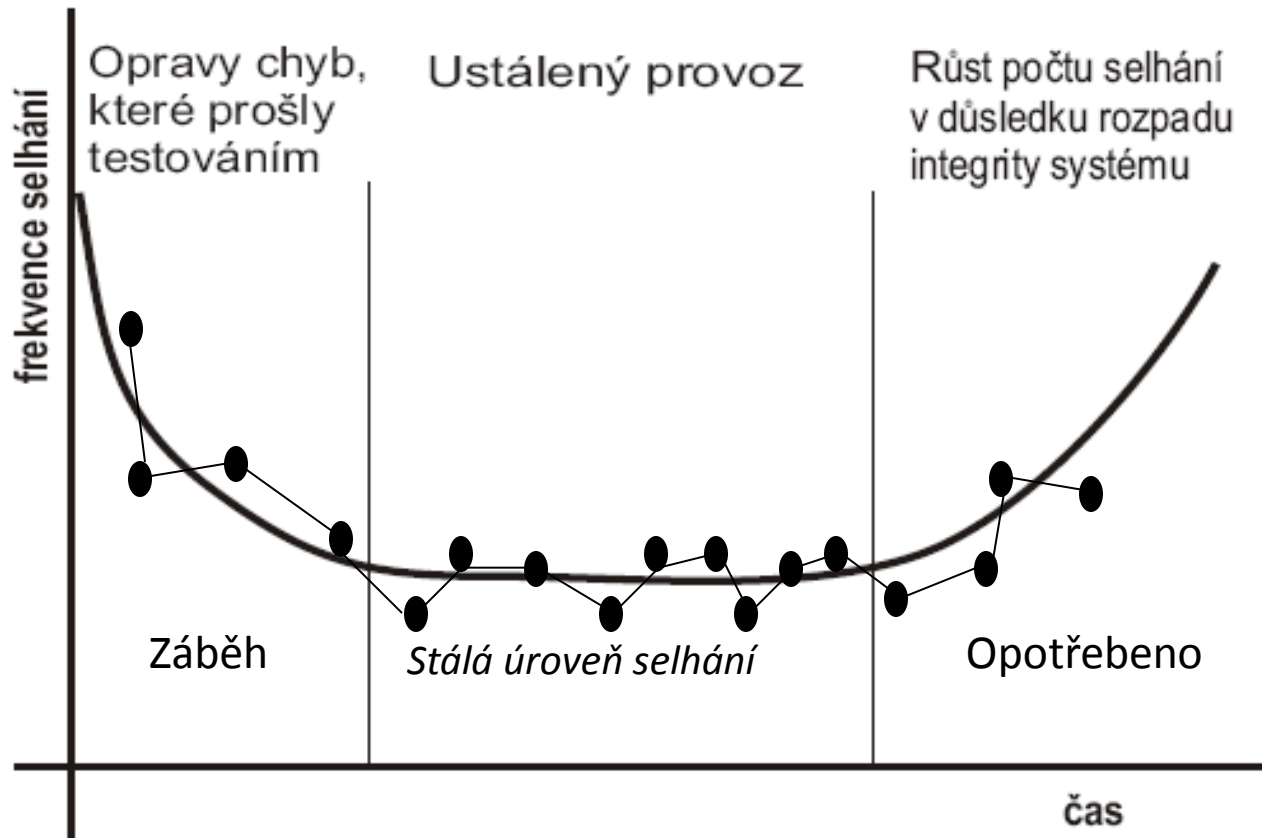
Skutečnost: Softwarové systémy se chovají jako složité systémy:
frekvence selhání sleduje křivku známou z teorie spolehlivosti
záběh, provoz, opotřebení

Provoz vyžaduje stále další úpravy (vylepšení, adaptace na nové platformy), to platí pro informační systémy a kupodivu stále více pro vědecké systémy

Výzva. Po čase se musí systém zahodit nebo přepsat, na to často stačí jen velké podniky

- pokud systém lze někdy přepisovat po částech (inkrementálně), je to dostupné i menším podnikům
- pak ale musí mít SW vhodnou architekturu, dnes obvykle servisní

Vanová křivka. I SW se opotřebí platí zvláště pro IS a human oriented SW



Nedělat monolity

- Dekomponovat do autonomních částí tak, aby se daly snadno integrovat.
- Klíčové je dělat to inkrementálně, např. metodami agilního vývoje.
- Pokud zvolím správné rozhraní lze snadno dělat prototypy, zmnožovat komponenty, přebírat hotové, zaměňovat,

Chce to xml, inteligentní síť, uděláme příklady.

Další průšvihy

- Vede to na metodiku velkého třesku
 - Strašně drahé
 - Vlastně nový vývoj
 - Velký spěch, nedosažitelné termíny, překročení nákladů
 - Nemusí se vůbec povést
 - Obtíže se specifikacemi
 - Obtížné pro uživatele
 - jistý čas používá staré i nové
 - Nevyužívají se léty nashromážděné znalosti a dovednosti uživatelů....

Existují metodiky jak se tomu všemu vyhnout

- Modulární systémy různé úrovně zrnitosti
- A získat další výhody jako je
 - Permanentní modernizace
 - Integrace s jinými systémy
 - Přílepky
 - Zapojení lidí
 - Záložní řešení, snadná paralelita
- Je to ale spojeno s manažerskými rozhodnutími

Proč jsou SME stále úspěšné

- SW systémy jsou stále komplexnější i díky globalizaci a stále větší funkcionalitě a zapojování do společenských procesů
- Velcí hráči zdánlivě stále více ovládají svět SW
- SME však stále obsazují podstatnou část scény.
- Velcí znovu používají existující malí také. Jak je to možné

Situace SME

- SME nemohou dělat podrobné modelování
 - Není čas, nejsou lidi, nejsou peníze
- Lze řešit zcela nové věci
- Snáze se využijí špičkoví pracanti
 - Ti často udělají 20krát více a to o lépe než běžní pracanti
- Nemohou dělat velký třesk, části
- Důsledek je moderní architektura řešení

Kouzelný proutek

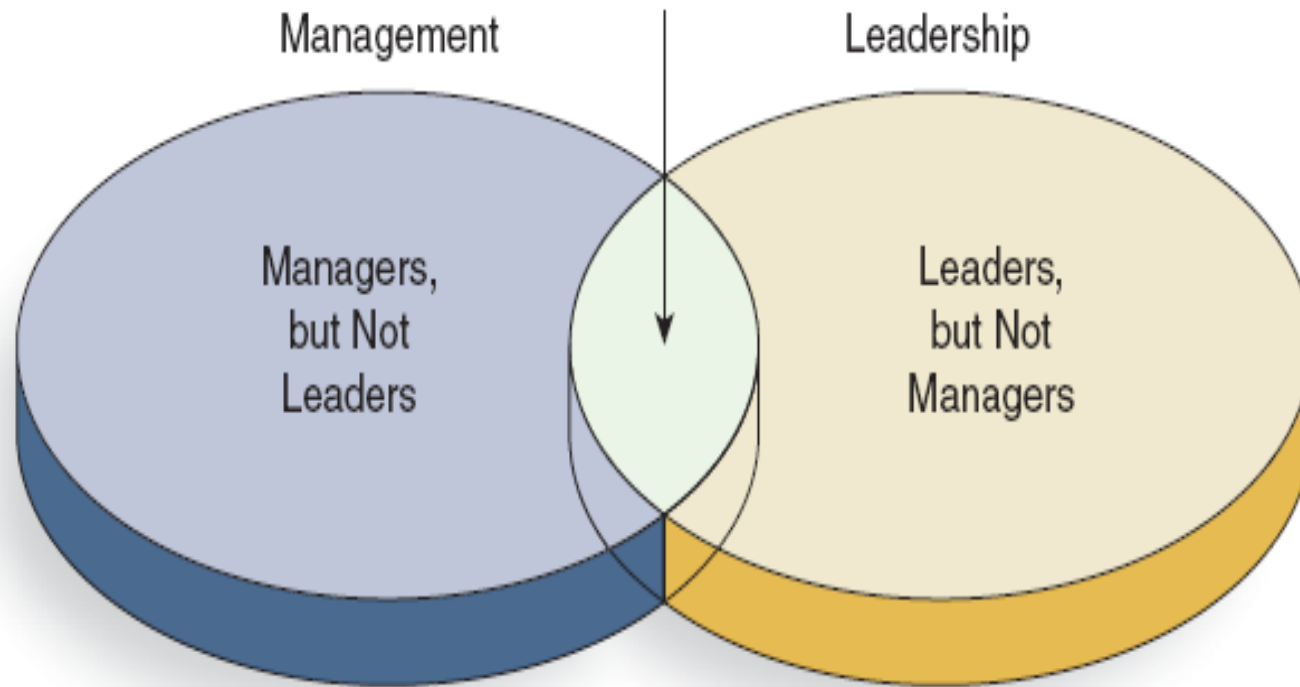
- Dekompozice do autonomních modulů a jejich pořízení nebo vývoj a integrace
- Chytrá rozhraní na vyšších úrovních (xml, digitalizované lidem srozumitelné dokumenty vhodně zapouzdřené)
- Záměnnost modul
- Organizační (architekturní) aktivity
- Příklad: Košilky, prototypy s flexibilním uživatelským rozhraním

Příklady architekturních služeb a chytrého SOA

Pozorování

- Roste podíl činnosti, které jsou manažerského resp. obchodního charakteru
- Vedle provozních záležitostí roste význam personalistiky
- Je nutné chápat i omezení zdravotní. To se týká vlastností SW i prostředí
 - Střídání činností
 - Nemoci psychické i fyzické (RSI,)

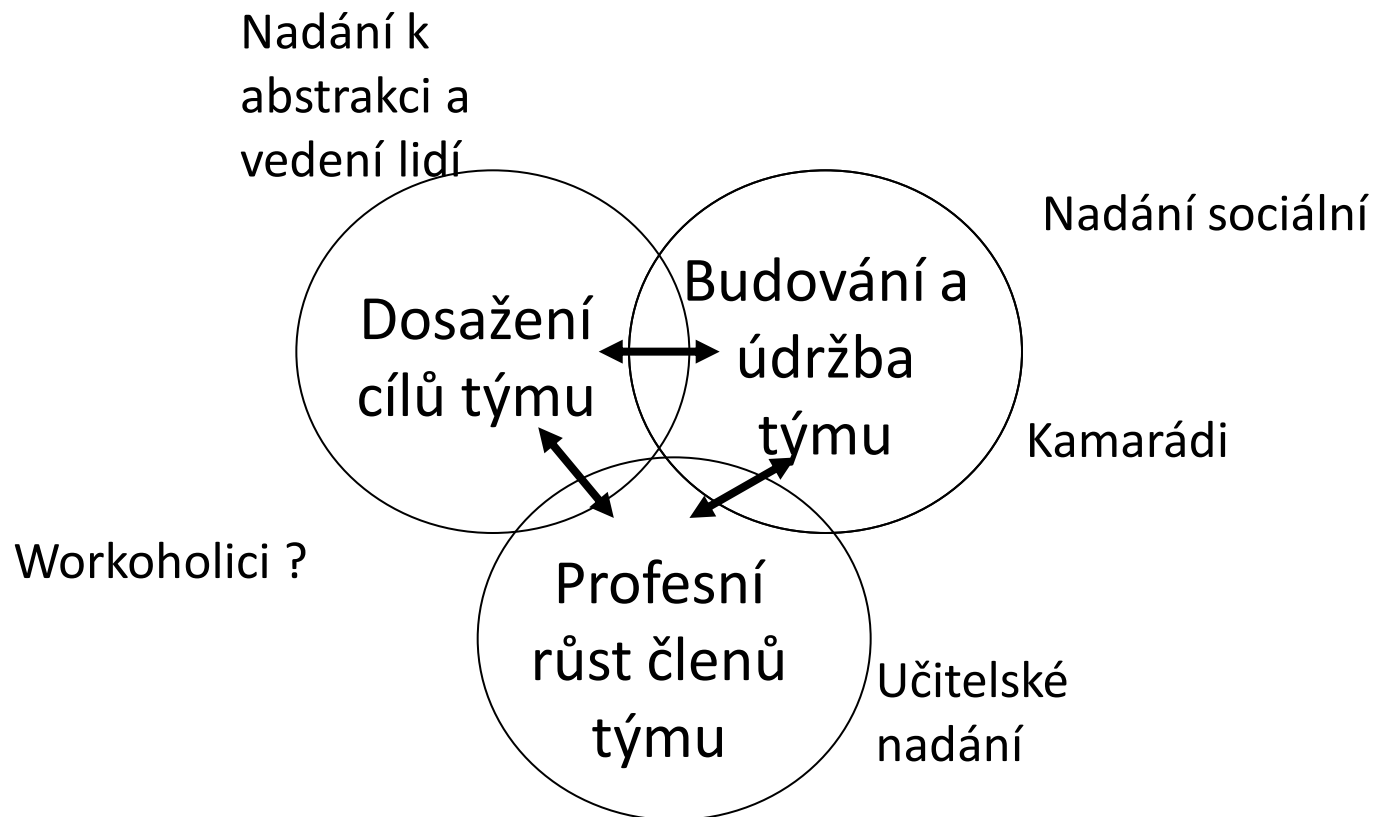
The Skills to Be
Both a Manager and a
Leader



Dovede zorganizovat a
zařít, zabezpečovat
provoz, dohlížet,...

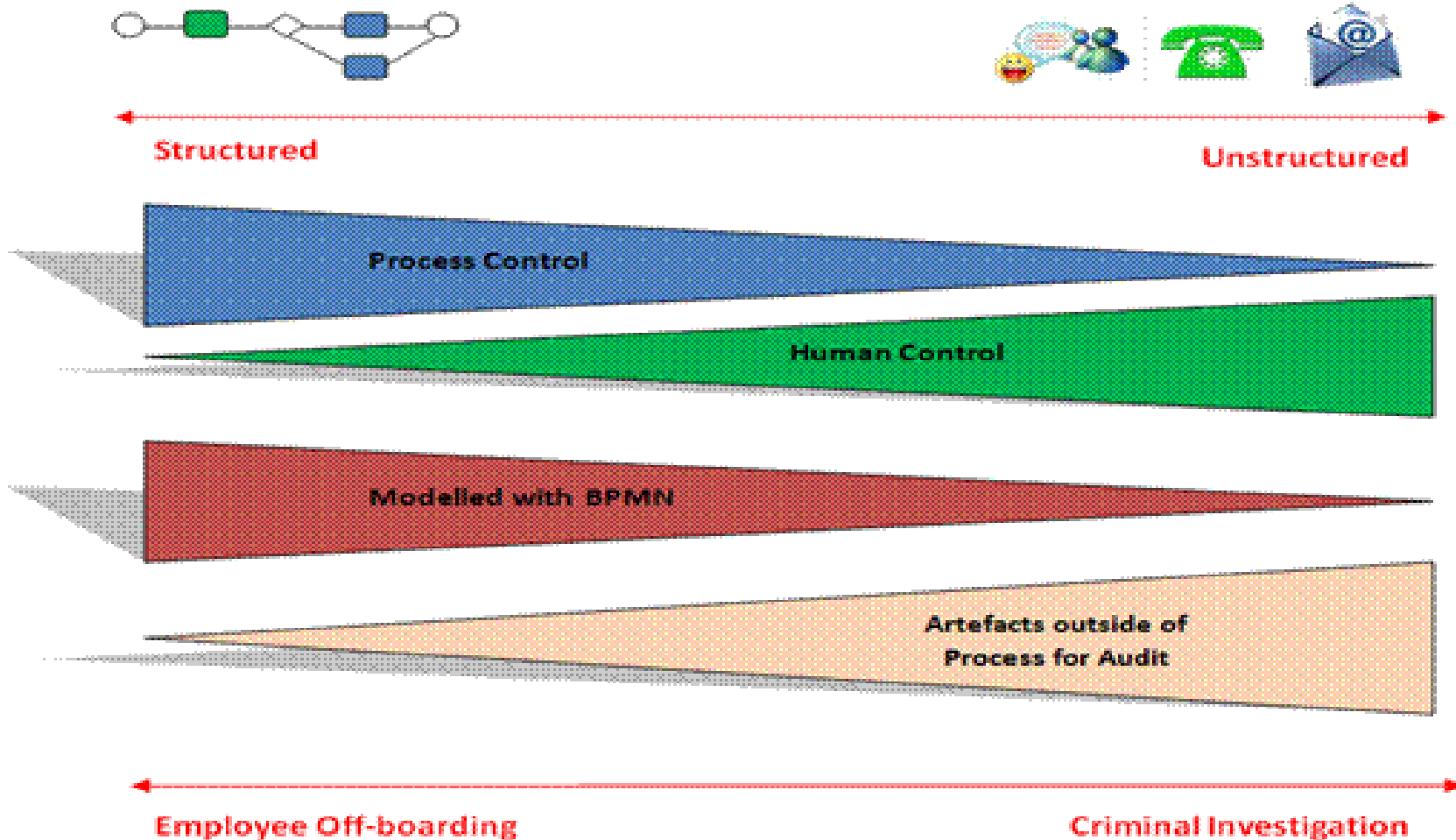
Dovede přesvědčit a najít
řešení a řídit po stránce
odborné

Tři aspekty činnosti týmu



V SME je nutné sladit cíle týmu s cíli podniku, podnik má poskytovat i jinou podporu (nábor lidí, školení, sociální AKCE)

Obrázek od Oracle k normě týkající se byznys procesů



Dříve převažovala tendence posunu vlevo dnes se to zčásti obrací
Platí to obecné, nejen pro bpmn