

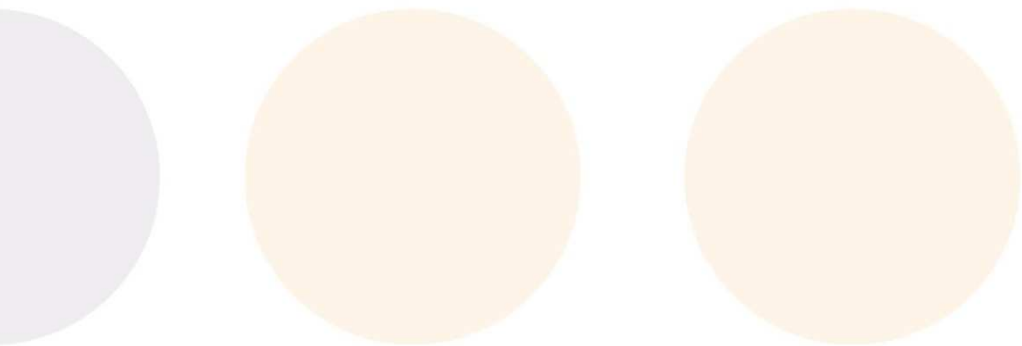


CM, Prostředí, Údržba



Agenda

- Docházka
- CM
- Vývojové prostředí
- Údržba
- Dotazy, přání, stížnosti





CM

- Configuration management != konfigurace systému.
- Configuration management ~ správa verzí
- Typická situace
 - V produkci je B3
 - V akceptaci je B4
 - Testuje se B5
 - Pracuje se na B6
 - Chyby z produkce je třeba opravit prioritně – i když se na nových verzích neprojevují.



CM – co verzovat

- Zdrojové kódy (ne generované)
- Data
 - Databázové dumpy (v rozumné míře)
- Knihovny
- Dokumenty
- Celé dodávky
 - Minimálně ta poslední – abyste byli schopni zjistit, co že jste to zákazníkovi vlastně poslali.



CM – nástroje

- CVS, SVN, ...
 - Základní nástroj
 - Branche
 - Nevhodné pro velké soubory
- Sdílený disk
 - Vhodný na velké soubory
 - Nevhodný na sdílení zdrojových kódů
- Bugzilla
 - Vhodná na dokumenty



CM - vývoj vs. údržba

- Zásadní rozdíl
- Během vývoje
 - Jede se nad trunkem
 - Člověk si vystačí s commit a update (a občas merge)
 - Feature větve (někdy)
- Údržba
 - Časté přecházení mezi větvemi
 - Mergeování změn z větví a mezi větvemi



Dodávky

- Když máme co, stačí to dodat :-)
 - Co? Komu? Jak?
 - A) Na servery nasazujeme dodávku my
 - B) Zákazník si nasazuje dodávku sám





Dodávky – A)

- Dodávku nasazujeme my
 - Jednodušší z pohledu chyb v dodávce
 - Případné chyby jednoduše opravíme a dodáme znovu
 - Pozor - některé věci jdou dělat jen jednou
 - je dobré na to skripty připravit

IF NOT EXISTS (SELECT 1 FROM syscolumns WHERE id =
OBJECT_ID('p_ink_predp_ppr') AND name = 'zaznam_id')

ALTER TABLE p_ink_predp_ppr ADD zaznam_id numeric(9) **null**
go



Dodávky – B)

- Zákazník si dodávku nasazuje sám
 - Nemáme práci se samotným nasazením
 - O to lepší musí být dodávka
 - Instalační postup musí být hodně dobře popsáný a připravený na různé varianty (nasazení na akceptační, ostrou)
 - Počet obrátek v akceptaci můžou měřit... a nějak vnímat :-)



Dodávky

- Velký release
 - Dodáváme celý systém
 - Dodávka má zpravidla pevně danou strukturu
 - Je potřebné si dopředu stanovit, jak co dodáváte
 - Přepsání konfigurace by mohl být slušný problém
 - Tedy jestli dodáváme všechno, nebo jen inkrementy něčeho a zbytek celý, ...
 - Po dodávce do akceptace následuje testování u zákazníka
 - Přijdou chyby (nedoufejte v opak).
 - Z toho plyne další dodávka
 - Lépe dodávat jen změněné instalační sety (psychologický efekt)
 - Obvykle máme dvě obrátky v akceptaci, jedna je výjimka, tři snad ještě nebyly :-)



Dodávky

- Malá dodávka (patch)
 - Používáme pro „malé“ změny mezi klasickými buildy
 - Dodáváme pouze změny
 - zdrojáky, zkompilované classy, data
 - vždy jen to, co se změnilo
 - v některých ohledech rychlejší
 - méně dokumentace
- Dopad dodávky na CVS (SVN)
 - Většinou to znamená vytvořit novou větev
 - Ale pokud čekám obrátky v akceptaci a jen opravuji, chvíli mi stačí main trunk (HEAD)
 - Někdy stačí tag



Vývojové prostředí

- Nejenom IDE
- Vývojové prostředí umožňuje vývojářům vyvíjet
 - Velmi neformálně řečeno
- Soubor HW a SW (včetně konfigurace a dat)
 - Může to být jeden notebook stejně jako tucet serverů.
 - Občas se přidává ještě dokumentace a personální zdroje.



Prostředí

- Může jich být hodně
 - Vývojové lokální
 - Vývojové centrální
 - Testovací - automatické testy
 - Předakceptační
 - Akceptační
 - Produkční #1
 - Produkční #2
- Minimum je vývojové a produkční.
 - Rozumné minimum je vývojové, testovací, akceptační a produkční





Co vše to zahrnuje

- Všechny typy CIs (configuration item)
- Instalace
- Zálohy
 - Nepodceňovat
- SW infrastruktura
 - Např. OS, DBMS, ...
- Podpůrné nástroje
 - Ant, CVS/SVN, Bugzilla ...



Vývojové prostředí

- Mělo by se podobat cílovému prostředí
 - Ne vždy je to možné => projektové riziko
- Nedegradovat ho
 - Například vyřazení autorizace, protože mě obtěžuje
 - Práce s maximálním oprávněním





Automatizace

- Noční život
 - Stáhne se aktuální verze zdrojových kódů
 - Přeloží se
 - Nasadí se (včetně například uložených procedur)
 - Spustí se testy
 - Případně se do sdíleného adresáře kompletně vygeneruje dodávka.
- O výsledku je někdo informován
 - Důležité hlavně, když se něco nepovede



Automatizace (2)

- Automatizace tvorby dodávky
 - Stojí čas (nezanedbatelné množství)
 - Zrychlení času tvorby dodávky
 - Možnost tvorby dodávky v rámci nočního buildu
 - Testuje se oproti verzi, která půjde k zákazníkovi
 - Odstranění lidských chyb
 - Obzvlášť nepříjemné, pokud jde dodávka přímo do produkce (například obsahuje opravu kritické chyby)
- CruiseControl + Ant – u nás základní nástroje pro automatizaci.



Zálohování

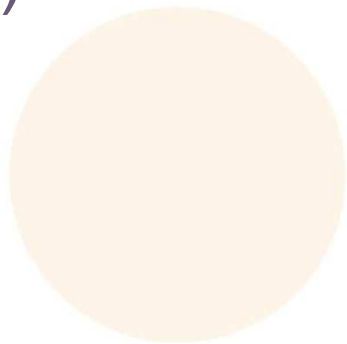
- Automatické
- Zálohuje se
 - Vývojová / Testovací /... databáze
 - Build skripty (včetně CC)
 - SVN/CVS
 - Bugzilla a ostatní podpůrný SW
 - Natavení systému (a ostatního SW)
- Práci si lze usnadnit použitím virtualizace
 - Zálohuje se celý virtuální server.





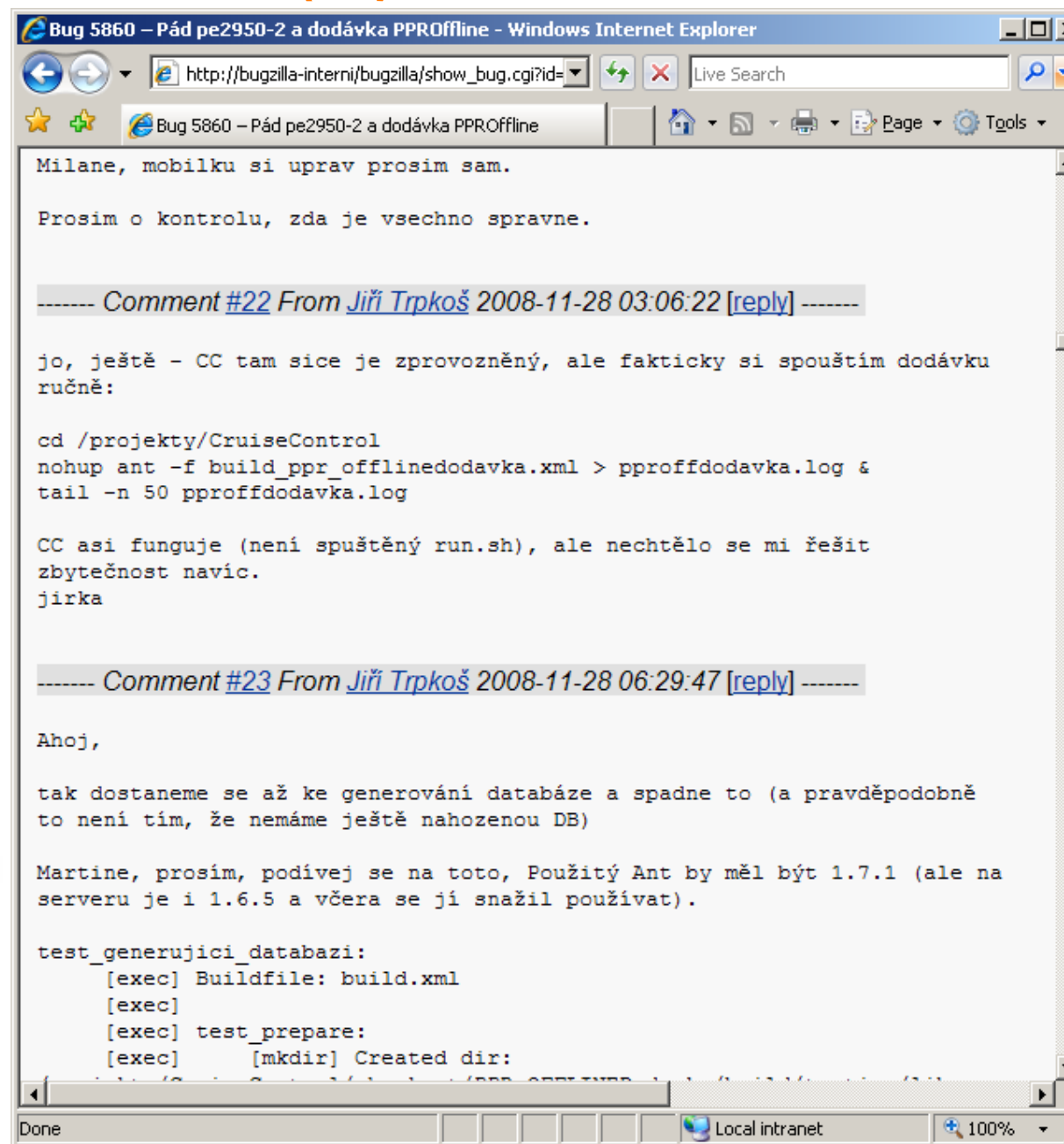
Příklad ze života

- V pátek dopoledne musela být dodávka.
- Ve čtvrtek odpoledne se zhroutil filesystém na hlavním serveru
 - Běžela na něm hlavní databáze (ze které se dodávka generovala)
 - Běžel na něm CruiseControl (který dodávku generoval)



Příklad ze života (2)

- Bylo třeba obnovit server ze záloh.
 - < 18 hodin
- Odpovídající HW nebyl k dispozici
 - Použití VMWare
- Někteří se moc nevyspali
 - Viz. obrázek





Příklad ze života (3)

- Obnovit plnou funkcionalitu se nepodařilo
- Nakonec jsme použili část starší dodávky (ze střeďeční noci)
- Část jsme sestavili ručně
 - S velikým rizikem, že se na něco zapomene
- Zachránilo nás to, že vývojové prostředí tvoří více fyzických a virtuálních serverů.



Příklad - poučení

- Zálohujte
 - Všechno a pořád
- Zkuste si občas ty zálohy obnovit
- Pokud to lze, rozdělte klíčové funkce mezi více serverů
 - Databáze, tvorba dodávky (CruiseControl), ...
- Virtualizace může pomoci
 - Zálohování celé VM



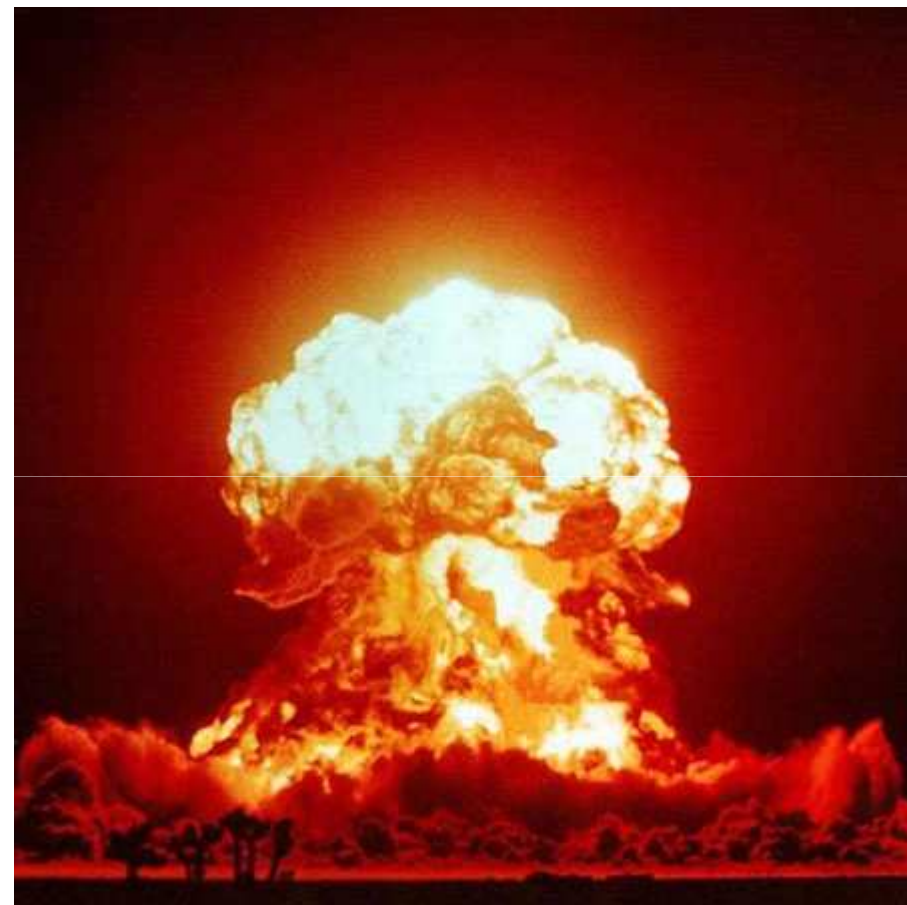
Důsledek

Přestanou-li existovat

- Sdílené disky
- CVS/ SVN
- Počítače lidí na projektu
- Vývojové prostředí

pak ze záloh je možno postavit
vývojové prostředí.

- Ekonomicky!





Údržba

- Fáze života SW po prvotním nasazení do ostrého provozu.
- Mýtus – údržba je zlo.
 - Musíte zachovávat zpětnou kompatibilitu
 - Není moc prostoru pro radikální architektonická rozhodnutí.
 - Systém často udržují lidé, kteří ho nevytvářeli
 - Neznají ho, nechápou ho.
- Většinou nejvýnosnější část projektu
 - Pokud se na ní dobře připravíme.



Příprava na údržbu

- Kvalitně systém navrhnout
 - Myslet dopředu.
 - Pokud to teď zákazník nechce, nebude to náhodou chtít za rok?
 - Nepřehánět to.
- Dobrá dokumentace
 - Kuchařky
- Kvalitní regresní testy
- Dobrá specifikace
 - Jinak se k údržbě nikdy nedostanete



Údržba

- Změnové řízení = miniwaterfall
 - Specifikace, design, programování, testování, ...
- Waterfall je zde vhodný
- Menší riziko podcenění pracnosti
 - Systém známe
- Platí pro ní stejné poučky jako pro vývoj z nuly.
 - Až na pár výjimek (viz waterfall)





Typické problémy

- Podcenění
- Opomenutí
- Chut' "vydělat" na parciální věci
- Závislost na konkrétních lidech
- Ne každý se tím dokáže nadchnout
- Plíživé ničení architektury, designu, prg. technik
- "Čemu nerozumím mažu,"
 - vs. „Bojím se cokoliv smazat, abych něco nerozbil“



Příklad ze života

- Zákazník hlásil chybu:
 - Po změně hesla v aplikaci dojde při komunikaci s centrem k zablokování uživatelského účtu.
- Problém byl způsoben následujícím řádkem:
 - `user.setPassword(password) //heslo pred zmenou`
- Řádek mi nedával smysl, tak jsem ho zakomentoval.
 - Čímž se problém odstranil



Příklad ze života

- Jeden kolega si všiml mé úpravy a nebyl z ní příliš nadšen.
 - Řádek smysl měl – řešil předchozí chybu.
- Společně jsme vymysleli řešení
 - Přesunuli jsme řádek před kritickou sekci.
- Nyní fungovaly opravy obou chyb
- Bohužel se rozbila jiná maličkost – autentizace
 - Šlo se přihlásit libovolným heslem



Cvičení

- Specifikace ZŘ již byla zákazníkem odsouhlasena, ale nyní požaduje provedení drobné úpravy – jak se zachováte?
- Zákazník hlásí chybu v internetovém obchodě, který jste mu dodali – při dvojím kliknutí na tlačítko „Odeslat objednávku“ dojde k dvojnásobnému založení objednávky v systému. Jak budete reagovat?
 - Opravit chybu ve vývojovém prostředí zabere 2 hodiny, do kdy slíbíte, že bude mít zákazník novou verzi?



Cvičení

- Začíná růst chybovost vašich dodávek. Co s tím uděláte?
- Zákazník rozporuje váš odhad 5 MDs na změnu kódu u jednoho produktu (kód se používá v DB jako primární identifikátor). Jak mu to vysvětlíte?
- Zákazník není spokojen se stávajícím dodavatelem a poptává vás na převzetí údržby systému. Jaké informace si vyžádáte?



Diskuse

- Komentáře
- Otázky
- Připomínky
- Upřesnění
- Poznámky
- ...

