

Testování softwaru - koncept kvality

Radek Mařík

CA CZ, s.r.o.

September 14, 2007



- 1 Proč testovat
 - Studie softwarových projektů
 - Charakteristika testera
 - Typické problémy vývoje softwaru
- 2 Definice testování softwaru
- 3 Koncept teorie kvality
 - Pojem kvality
 - TQM

Studie softwarových projektů

IBM's Consulting Group, June 1994

- průzkum 24 významných společností ^[?]
- 55% softwaru vyviuto za cenu větší než plánované,
- 68% vyvíjeno delší dobu než predikováno,
- 88% muselo být podstatně přenavrhnuto.

The Standish Group, 1994

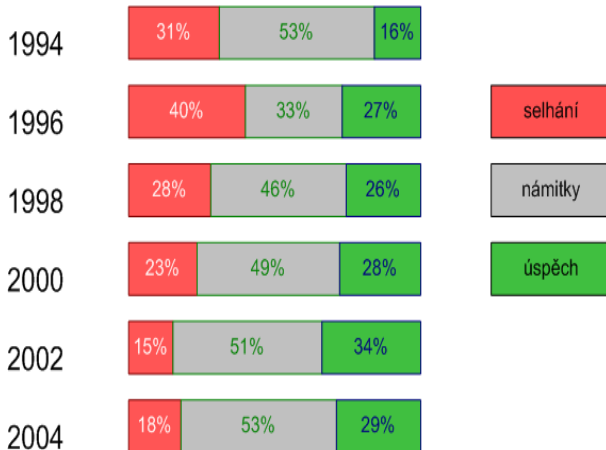
- studie 8 380 projektů,
- 31% softwarových projektů přerušena,
- náklady 53% dokončených projektů se pohybují okolo 189% původních odhadů,
- z těchto 53% pouze 42% obsahuje původní sadu navrhovaných vlastností a funkcí,
- pouze 9% z těchto projektů bylo ukončeno v dohodnuté době a ceně.

Statistika softwarových projektů

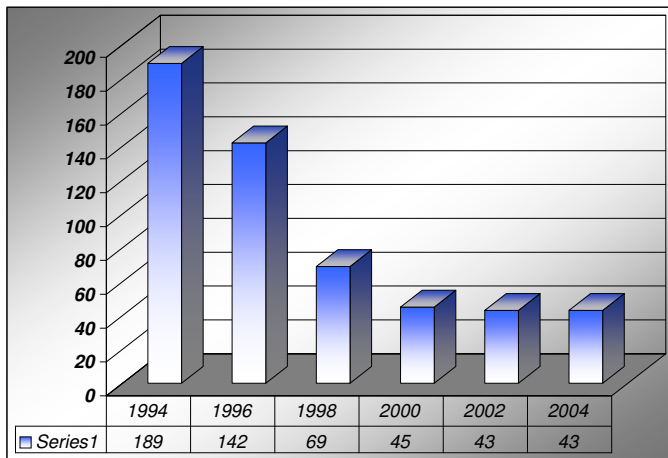
Obecně

- 5 ze 6 softwarových projektů je neúspěšných,
- 1/3 projektů je přerušena,
- projekty předávány za dvojnásobnou cenu než dohodnuto,
- projekty se předávají za dvojnásobně dlouhou dobu než se plánuje.

Vývoj úspěšnosti projektů (CHAOS)



Rozpočty projektů (CHAOS)



Poměr vývojář/tester: [Kit95]

Obecně

- minulost: 1 tester, 9 vývojářů,
- nové trendy: 2 testeři, 3 vývojáři,
- může být 10:1 až 1:10,

Konkrétně

- Microsoft, 1992: 2:3,
- Lotus (pro 1-2-3 Windows): 2:1,
- průměr 4 velkých firem (1992): 1:2
Microsoft, Borland, WordPerfect, Novell

Odborný profil dovedností testera

Znalosti

- znalost systému či produktu,
- znalosti a zkušenosti s programováním
 - porozumění,
 - příprava podpory,
- analytické postupy,

Postupy

- orientace na detaily,
- schopnost myslet kreativně,
- dobrá představivost

Osobní profil dovedností testera

Charakter

- trpělivost,
- sebe-motivace,
- vytrvalost.

Týmová práce

- dobré komunikační dovednosti,

Ariane 5

Situace

- Řada motorů na tekuté a tuhé palivo nahrazena několika s větším tahem.
- 4.června, 1996, 40 s po startu ve výšce okolo 3700 m se nosič odklonil od své dráhy, rozlomil a explodoval.
- Raketa, nesené 4 satelity nepojištěny, 500 miliónů \$.



Selhání nosiče Ariane 5

Status testování

- Žádost o přetestování stabilizační plošiny převzaté z Ariane 4 v podmínkách Ariane 5 byla “vetována CNES z důvodu vysokých nákladů”.
- Sextant Avionique po havárii potvrdila, že by závadu svými testy detekovala.

Chyba

- softwarová vyjímka v obou Stabilizačních referenčních systémech (SRI).
- nechráněný převod z 64bitového reálného čísla na 16bitové celé číslo.
- SRI má význam pouze před zvednutím nosiče, ačkoliv je operativní ještě dalších 50 s.
- přetečení nastalo z důvodu rozdílných drah.

Shrnutí Arian 5

Typické chyby při procesu vývoje softwaru

- nedostatek času - veto na testování
- malé či chybně rozložené náklady - veto na testování,
- chybné nebo chybějící požadavky - jak dlouho by měl podsystém fungovat,
- chyby v kódu - nechráněné převody,
- opakované využití - změna specifikací,
- řada chyb vzniká striktním oddělením vývoje softwaru a jeho testování.

Radiační předávkování

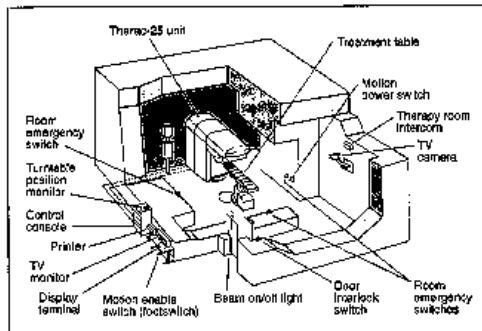


Figure 1. Typical Therac-25 facility.

Therac 25

- červen 1985 - leden 1987
- lineární urychlovač
- používaný v lékařství k ozařování rakovinných nádorů,
- povrchové tkáně ozařovány elektrony,
- pro hlubší tkáně gama záření,
- 6 incidentů přezáření, z toho 3 smrtelné,
- 20000 rad místo 86 rad,
- systém reálného času vytvořený 1 programátorem,
- neexistující formální specifikace a testovací kritéria,
- hardwarové zámky nahrazeny programovými,
- pokud byla vstupní data změněna mezi 1 až 8 s, pak zářič a polohovací stůl pracovaly v různých módech,
- k nastavování logické proměnné použita inkrementace bytové proměnné.



Shrnutí Therac 25

- uživatelské rozhraní kontra bezpečnost,
- složitý návrh
- systémové testování není dostatečné,
- chybějící specifikace,
- typicky problémy systémů:
 - paralelních (angl. parallel)
 - souběžných (angl. concurrency).

Zkušenosti z chyb



Proslulé chyby I

Oběžná dráha Apollo 13: program testován za pomalu měnících se podmínek. Při velké dynamice došlo k vydělení nulou na netestované cestě.

Mariner let k Venuši: 80 miliónů \$,
záměna – za + vedla k odklonu z dráhy,

Minutí Merkuru: proměnná Fortranu *DO10I*

DO 10 I=1.5

DO 10 I=1,5

Selhání rakety Patriot: během Války v zálivu v 1991 kvůli kumulativní chybě v časové synchronizaci
(ve skutečnosti: 0.34 s, 100 hodin; navrženo: 14 hodin),

F16 simulace: letadlo se překlápělo při překročení rovníku,

Návrh jaderné elektrárny: v roce 1979 muselo být 5 jaderných elektráren uzavřeno z důvodu poddimenzování potrubí, velikost vektoru počítána jako součet složek, modul byl napsán studentem na praxi.



Proslulé chyby II

Sonda Marsu Vymazán přistávací modul s cílem uvolnit paměť. Modul navigace antény k Zemi vyžadoval navigační funkce obsažené v přistávacím modulu.

Selhání regulace síťového systému Kalifornie: 1998

Nezbyl čas na řádné testování komunikačního systému.
Vzniklé zpoždění stálo přibližně 90 miliónů \$.

Zpoždění otevření letiště v Denveru: 1995 Chyby v systému řízení zavazadel způsobily rozdrčení automatizovaných vozíků se zavazadly o stěny. Letiště bylo otevřeno o 16 měsíců později se ztrátou 3.2 miliardy \$ a s manuálním zavazadlovým systémem.

Porucha Pendolina



- 2005-2006
- 89 závad za první měsíc provozu,
- závady v obslužném software pomocných měničů (topení, osvětlení),
- případná ztráta se pohybovaly okolo 43 miliónů euro,



Testování softwaru - výchozí definice

- Hetzel 1973** Testování je proces určení věrohodnosti, že program či systém dělá to, co se o něj předpokládá.
- Myers 1979** Testování je proces spouštění programu či systému s úmyslem nalézt chyby.
- Hetzel 1983** Testování je jakákoliv aktivita s cílem vyhodnotit atribut či schopnost plnění požadovaných výsledků programem nebo systémem.

Testování softwaru - přehled definic

Testování je

- kontrola programů vzhledem ke specifikacím,
- nalézání chyb v programech,
- určení míry akceptování uživatelem,
- ujištění se o tom, že systém je připraven k používání,
- získání důvěry, že program pracuje,
- prezentace, že program běží správně,
- demonstrace toho, že program je bez chyb,
- porozumění omezení výkonnosti programu,
- učení se toho, co systém není schopen dělat,
- hodnocení schopností programu,
- verifikace dokumentace.

Testování

je měření **kvality** softwaru.

Definice kvality

Rozsah od inženýrských specifikací na úrovni dílny až po definice na úrovni společnosti:

Webster's New World Dictionary Kvalita je fyzická či jiná charakteristika, která definuje základní podstatu věci či jednu z jejích vyznačných vlastností.

Crosby 1979 Kvalita je mírou souhlasu s požadavky.

ISO 9000 Kvalita je souhrn vlastností a charakteristik produktu či služby, která se týká schopnosti uspokojit určených nebo vyplývajících potřeb.

Taguchi 1986 Kvalita je ztráta, kterou produkt způsobí společnosti po jeho dodání, způsobenou funkčními změnami a škodlivými účinky mimo těch, které vyplývají z vlastních funkcí.

Aspekty kvality

- operační podmínky - výkonnost v krátkodobém horizontu,
- spolehlivost - dlouhodobý horizont,
- pohled zákazníka,
- IKIWISI - Guaspari: "I Know It When I See It" [Kit95]

Ideální kvalita,

- kterou zákazník může očekávat, je,
- že každý produkt poskytuje cílenou výkonnost
- kdykoliv je použit,
- za všech zamýšlených operačních podmínek,
- po celou dobu jeho předpokládaného života,
- se žádnými škodlivými postranními efekty.

Kvalita softwaru

Kvalita znamená "splňovat požadavky zákazníka":

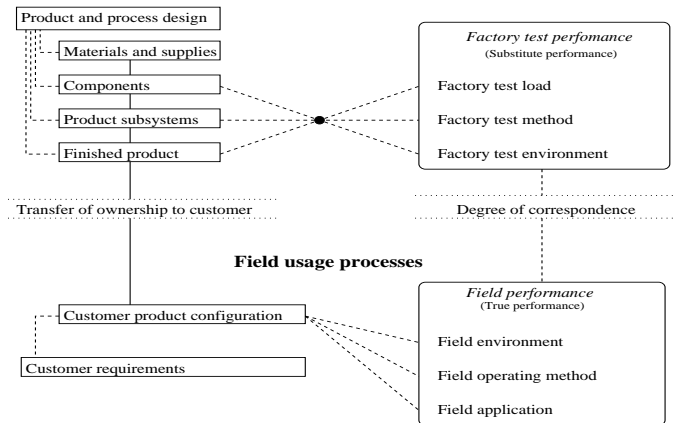
Faktory:

- *Funkčnost (externí kvalita)*
 - správnost,
 - spolehlivost,
 - použitelnost,
 - integrita.
- *Inženýrské řešení (vnitřní kvalita)*
 - efektivita,
 - testovatelnost,
 - dokumentace,
 - struktura.
- *Adaptabilita (budoucí kvalita)*
 - flexibilita,
 - opětné použití,
 - údržba.

Koncept kvality

Vztah mezi skutečnou kvalitou produktu pocítovanou zákazníkem a kvalitou měřenou na úrovni produkce:

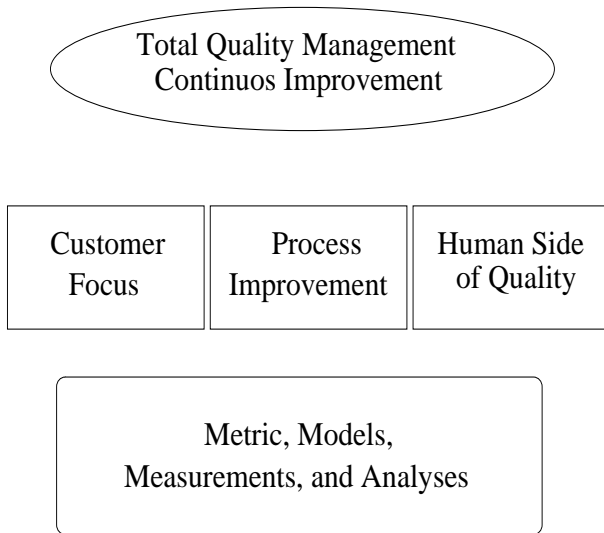
Factory production processes



Řízení celkové kvality (TQM) ^[Kan95]

- 1985 Naval Air Systems
- japonský způsob zlepšování kvality,
- znaky:
 - kontinuální zlepšování procesu,
 - zaměření na zákazníka,
 - lidská stránka problému,
 - metriky, modely, měření, analýza.
- Metodologie:
 - The Malcolm Baldrige National Quality Award (MBNQA) - USA, 1988,
 - ISO 9000, Evropská komise,
 - Hewlett-Packard's Total Quality control (TQC),
 - Motorola's Six Sigma Strategy,
 - IBM Market Driven Quality

Klíčové elementy vztahů TQM ^[Kan95]



Závěr

- Základní cíle testování.
- Testování jako aplikace teorie kvality.
- Základy teorie kvality.

Literatura I



Stephen H. Kan.

Metrics and Models in Software Quality Engineering.
Addison-Wesley, 1995.



Edward Kit.

Software Testing in the Real World.
Addison-Wesley, 1995.