

Fakulta informačních technologií
Vysoké učení technické v Brně



Příručka pro studenty předmětu Formální jazyky a překladače
Práce v týmu

Lukáš Rychnovský

Zbyněk Křivka

Radim Kocman

Projekt FRVŠ 673/2007/G1

Aktualizace 4. 9. 2014

Úvod

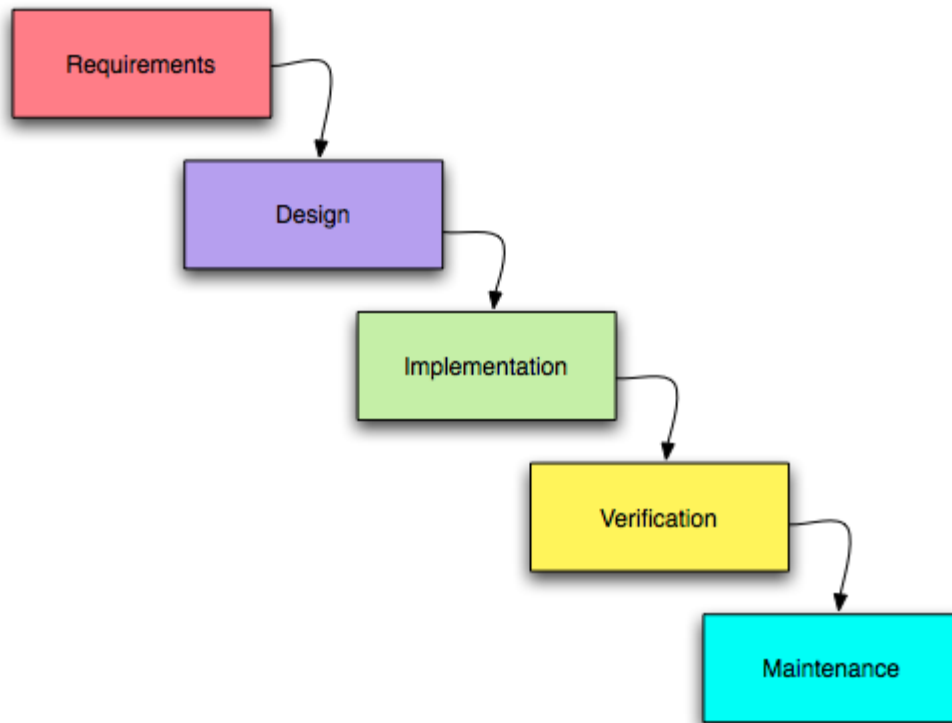
Projekt do předmětů *Formální jazyky a překladače* (IFJ) a *Algoritmy* (IAL) má hned dvě nezvyklá specifika. Za prvé projekt je týmový a za druhé společný pro oba předměty. Důvod k prvním je především kvůli náročnosti na hodnocení tak poměrně velkého projektu jako je překladač při tak velkých počtech studentů, jaký v současnosti oba kurzy navštěvuje (cca 500-700).

Protože programování překladače vyžaduje využívání nejrůznějších datových struktur a v případě týmové spolupráce nejlépe abstraktních datových typů, tak je sdružení projektu IFJ s IAL naprosto přirozené. Navíc IAL přináší další nová specifika do projektu jako nutnost implementovat pokročilejší vyhledávací struktury (např. hashovací tabulky, binární vyhledávací stromy, AVL stromy) a připravit si i osobní prezentaci projektu, aby se studenti naučili vystupovat a prezentovat postup a výsledky své práce.

Definice projektu

Projekt je časově omezený proces, který má za cíl vytvoření nějakého konkrétního produktu nebo služby. Každý projekt má svůj začátek, který započne definicí požadavků na produkt (*requirements*), plánování projektu a plánování zdrojů (lidských, finančních, materiálních, energetických, prostorových, komunikačních, ...). Výsledkem plánování projektu je projektový plán, který se sestává z plánu zdrojů, odhadů nákladů, časové náročnosti atp. Druhá fáze projektu je vytvoření (*implementation* - výroba) produktu. Během této fáze se spotřebovávají zdroje a přeměňují se na produkt. Po ukončení vývoje produktu se výsledek verifikuje oproti projektovým požadavkům a vyhodnotí se. Následuje dlouhodobější užitek z projektu, který přináší zpět vynaložené prostředky.

Základní model, tzv. vodopádový model, fází produktu při jeho vývoji/tvorbě v projektovém řízení je tedy následující.



1 Vodopádový model fází projektu. Zdroj: wikipedia.org

Pozice v projektu

V každém projektu je třeba rozdělit práci. Jednak přímo práci, kterou je třeba v rámci projektu udělat, ale také práci související s režii projektu. Podle typu práce na projektu, můžeme řešitele rozdělit do tří kategorií.

Vedoucí projektu: je hlavní řešitel, který je zodpovědný za projekt jako celek. Jeho náplní práce je většinou pouze dozor ostatních řešitelů a koordinace.

Sekční vedoucí: pokud je projekt veden víceúrovňovým řízením, pak se projekt dělí do menších částí takzvaných sekcí, jejíž každou zodpovědnou osobou je právě sekční vedoucí. Tento člověk se může z části podílet přímo na práci na projektu, z části na jeho vedení. Sekční vedoucí jsou koordinováni hlavním vedoucím projektu a zodpovídají se mu. Jsou zároveň odpovědní za vedení své sekce. K tomu jim slouží další řešitelé.

Pracovníci: řešitelé, kteří přímo vykonávají práci na projektu. Při víceúrovňovém přístupu jsou koordinováni sekčními vedoucími, případně pouze hlavním vedoucím.

Metriky hodnocení projektů

Každý projekt, aby jej bylo možno řídit a hodnotit, je třeba ohodnotit pomocí určitých metrik. Takovým měřítkem může být například počet lidí, kteří pracují na projektu, či počet týdnů, měsíců nebo let, po které bude projekt trvat. Některé z těchto metrik jsou pro některé metody hodnocení lepší, některé horší. Podíváme se nyní na některé z nich podrobněji.

Rozdělení projektů podle velikosti

Malý projekt (2-3 řešitelé): takto malé projekty většinou mohou fungovat stylem komunikace „každý s každým“. Všichni řešitelé ví, kdo co dělá, provázanost práce bývá minimální. Nutnost dokumentace je malá, neboť v případě problému je jasné, kdo danou část projektu má na starosti a lze se obrátit přímo na něj.

V případě IFJ se o malý projekt jedná například, pokud 2 lidé spolupracují na lexikálním nebo syntaktickém analyzátoru, případně psaní programové dokumentace. Komunikace mezi nimi může probíhat napřímo, mohou se pravidelně setkávat a daný problém řešit.

Střední projekt (3-10 řešitelů): u středně velkých projektů už řešitelé ztrácejí přesnou představu, kdo co přesně dělá a pod koho spadá ta která část projektu. Je tedy již potřeba vedoucí projektu, který celý koncept drží pohromadě. Dokumentace je nutná jako komunikační rozhraní mezi jednotlivými řešiteli.

Střední projekt svým rozsahem odpovídá celému projektu do IFJ. Již není možné, aby se celá skupina scházela, pokud potřebují spolupracovat na projektu. Komunikace v rámci projektu musí probíhat co nejrychleji. Jako výhodné v tomto případě je využívat emailové komunikace nebo instant messengerů (Skype, jabber, ICQ atp.). Schůzky celého projektu jsou občasné a řeší se koncepční záležitosti.

Velký projekt (10 a více řešitelů): v tomto případě se jedná již o rozsáhlé množství práce v rámci projektu a je třeba organizace, disciplíny a řádu. Organizace větších projektů je již třeba dělat víceúrovňově. Vedoucí projektu je mozkiem celého projektu a rozděluje práci mezi vedoucí jednotlivých sekcí. Zároveň pak od sekčních vedoucích sbírá informace o vývoji projektu. Dokumentace je nezbytná pro mezisekční komunikaci.

Rozdělení projektů podle délky trvání

Krátkodobý projekt (dny až týdny): jedná se většinou o menší projekty. Krátké a velké projekty nejsou běžné, neboť za několik dní se nestihne ustanovit organizace většího projektu ani mezisekční komunikace. Pro vnitřní potřeby projektu není třeba dokumentace, neboť všichni řešitelé mají většinu činností v živé paměti. U krátkodobých projektů také většinou nehrozí větší fluktuace řešitelů.

V případě projektu do IFJ je krátkodobý projekt jakýkoliv podprojekt, opět například lexikální nebo syntaktický analyzátor nebo psaní dokumentace. V tomto případě je třeba dát pozor na to, že někteří řešitelé nemají s programovacím prostředím zkušenosti a je třeba dostatečného času, než mohou vůbec na projektu efektivně pracovat (učí se teprve implementační jazyk nebo knihovny, neznají práci s hlavičkovými soubory atp.). Ve střednědobém projektu se takováto zdržení lépe plánují, u krátkodobého to může přinést problémy a je třeba s tím v projektovém řízení počítat.

Střednědobý projekt (týdny až měsíce): střednědobé projekty již mohou být i velké. Je tedy nutná dokumentace, jak pro vnitřní komunikaci, tak z důvodu možného časového odstupu v rámci projektu. V některých případech u střednědobých projektů již může zahrozit fluktuace řešitelů. V takovém případě je třeba věnovat dokumentaci zvlášť pozornost, neboť to může být jediný způsob, jak předat práci z jednoho řešitele na dalšího.

I v případě střednědobého projektu je při projektovém vedení třeba počítat se možným zdržením, jako u krátkodobého projektu. Zkreslení odhadů může vycházet z odhadů náročnosti práce podle nejsilnějšího člena týmu (často vedoucí projektu). Slabší členové pak často nedosahují výkonu a dochází opět k problémům.

Dlouhodobý projekt (měsíce až roky): u dlouhodobých projektů je třeba již od začátku počítat s odchody zaměstnanců a příchody nových. Zvlášť palčivé je to u sekčních vedoucích a hlavního vedoucího projektu. Celý projekt musí být veden tak, aby nebyl závislý na konkrétním personálním obsazení. Dokumentace je nepostradatelná, jak z důvodu předávání projektu dalším generacím, tak pro stávající řešitele pro připomenutí starších faktů.

Kombinace délky a velikosti projektu

Předchozí metriky (jak podle počtu lidí, tak podle délky trvání), nejsou však vždy pro posuzování projektu ideální. Neříkají mnoho o náročnosti projektu. I malý projekt může být dlouhodobý (opačná situace již tak častá není). Je tedy dobré zavést pojem „člověkodni“ (u studentů někdy spíše člověkonoc), který udává součin počtu lidí a předpokládané délky projektu. Tento součin však není volně převoditelný, když je odhadovaná složitost projektu 50 člověkodní, pak můžeme na měsíc najmout 3 lidi, ale není možné s 50 lidmi zvládnout tento projekt za hodinu. Rozdělení pak může být například:

Malý projekt (méně než 50 člověkodní): jedná se o malý a krátkodobý projekt, který se v případě potřeby dá urychlit přidáním dalšího řešitele (řešitelů).

Střední projekt (50 až 500 člověkodní): v tomto případě se již většinou nejedná o krátkodobý projekt, neboť by muselo být zapojeno kolem sta řešitelů (a jak jsme si řekli, krátkodobé velké projekty nejsou běžné). Při plánování projektu je třeba počítat

s pozicí vedoucího projektu, který svůj čas věnuje řízení projektu, nikoliv programování. Stejně tak je třeba počítat s vnitroprojektovou dokumentací.

Velký projekt (více než 500 člověkodní): jedná se o velký a dlouhodobý projekt se všemi aspekty popsanými výše. Hlavní vedoucí projektu a sekční vedoucí tráví svůj čas staráním se o projekt a rovněž dokumentace potřebuje dostatek času.

Další metriky

Samozřejmě je možné (a někdy i účelné) vymyslet mnoho dalších metrik, které nám mohou projekt charakterizovat. V projektech, které například nemají příliš předem daný rozsah, může být například metrikou počet vydaných verzí programu.

Rozdělení práce na projektu IFJ mezi jednotlivé řešitele

Týmy lidí jsou přibližně po 4 až 5 lidech (v závislosti na počtu studentů v daném akademickém roce). Jelikož se jedná o malý tým lidí, bude mít relativně jednoduchou strukturu. Není to nezbytně nutné, ale minimálně vhodné stanovit vedoucího projektu, který má u ostatních řešitelů dostatečnou autoritu (ať už odbornou nebo osobní).

Práce na projektu lze rozdělit například těmito dvěma způsoby:

- a) **hierarchicky**, kdy celý tým na inicializačních schůzkách provedou návrh celého systému a 1 či 2 úzce spolupracující řešitelé vytvoří kostru celého překladače bez implementovaných procedur a funkcí, ale s dobrou dokumentací sémantiky a chování jednotlivých funkcí. Dalším členům projektu se potom přidělí k implementaci všechny zatím neimplementované funkce, tak aby každý zodpovídal za disjunktní podmnožinu funkcí a sjednocení všech podmnožin pokrývalo celou implementaci. Při rozdělování podmnožin je třeba si uvědomit, že každá funkce má různou náročnost na implementaci a pro rovnoměrné rozdělení zátěže na jednotlivé řešitele je vhodné provést co nejpřesnější odhad této náročnosti.
- b) **vertikálně**, kdy lze práci překladače a interpretu rozdělit na několik fází či bloků, které spolu komunikují přes předem stanovené rozhraní. Postupem v tomto případě tedy je na iniciačních schůzkách řešitelů stanovit zodpovědnosti jednotlivých bloků a rozhraní mezi nimi. Vedoucí projektu potom hraje spíše úlohu dohlížejícího, který případně varuje, pokud některý člen týmu nedělá svou práci nebo markantně zaostává. Nástin možných bloků či úloh s relativně dobrou nezávislostí, aby je bylo možno implementovat zvlášť, je: lexikální analyzátor, syntaktický analyzátor jazykových konstrukcí (deklarace, definice, řízení programu), syntaktický analyzátor pro výrazy (aritmetické případně logické) interpret, implementace potřebných abstraktních datových typů a vestavěných funkcí zadaného jazyka (požadavky IAL), testování, vedení projektu a dokumentace.

Dokumentace projektu je také jeho nedílnou součástí. Dokumentace obsahuje dvě části: (1) programátorskou generovanou dokumentaci z komentářů ve zdrojových textech implementace (samozřejmě často je třeba ji také mírně ručně doplnit) a (2) uživatelskou manuálně psanou dokumentaci, která komentuje návrh a uživatelské vlastnosti vytvořeného překladače a interpretu. Na generování a sepisování dokumentace lze využít jiné přístupy než na samotnou implementaci (níže uvedené nejsou zdaleka jediné) jako například:

V případě a) vytvoří velkou část generované dokumentace právě vedoucí týmu, který vytváří kostru projektu a specifikuje funkčnosti všech deklarovaných funkcí. Pro ulehčení jeho práce by bylo vhodné z iniciačních schůzek provádět podrobné zápisy a případně i audio záznamy (POZOR! Jejich pořizování je spíše záložního charakteru než že by dostatečně nahrazovalo zápis.), které budou sloužit jako podklady při dokumentaci kostry implementace.

V druhém případě (b) není interní potřeba dokumentace na tak vysoké úrovni (až na dokumentaci rozhraní jednotlivých bloků a abstraktních datových typů). Proto je možné provádět dokumentaci až dodatečně (například ve fázi propojování bloků a jejich testování). V tom případě je z pedagogického hlediska dobré dokumentovat bloky, které daný člověk neprogramoval, protože tím bude nucen pochopit i části implementace, které přímo nevytvářel, což se mu může velice hodit v případě obhajoby projektu.

Neúspěšný projekt

Stejně důležité, jako dovést projekt do zdárného konce, je v případě neúspěšného projektu rozpoznání blížícího se problému. Nemusí to být pouze definitivní konec projektu, důležité je včas rozeznat například i nedostatek času k dokončení projektu a tedy jeho prodloužení.

Nedostatek času

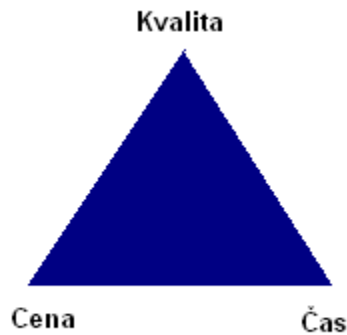
Nejčastějším problémem projektů bývá právě špatný odhad doby potřebné k dokončení projektu. To má za následek prodlužování projektu a neustálé odkládání doby dokončení. K rozeznání těchto problémů v samém počátku se používá systém průběžných cílů (milestones). Celý projekt je rozdělen do navazujících procesů, kde jeden začíná tam, kde končí předešlý. V případě problému s dokončením některého průběžného cíle se musí vedoucí projektu rozhodnout a na problém reagovat (většinou přidáním dalšího řešitele či upravením cíle).

Nedostatek řešitelů

Problém nedostatku řešitelů se projevuje stejně, jako předchozí případ. Řešením však nemůže být přidání řešitele, je třeba tedy upravit cíle projektu, či celý projekt prodloužit.

Nízká kvalita výsledku

Kvalita výsledku projektu je nedílným ukazatelem úspěchu celého projektu. Mezi třemi veličinami – kvalita, cena a čas platí následující vztah



2 Závislost Kvalita - Cena - Čas v projektovém řízení

Chceme-li zvýšit kvalitu výsledného projektu, pak zároveň musíme buď prodloužit čas, nebo zvýšit odhadovanou cenu projektu (případně obojí). Naopak, chceme-li snížit cenu projektu, pak zároveň prodlužujeme čas a snižujeme kvalitu. A posledním případem je zkracování délky projektu, což se projevuje buď zvýšením ceny, nebo snížením kvality.

Software pro podporu projektů

Pro podporu řízení a hodnocení projektů existuje mnoho různých softwarových nástrojů. Od jednoduchých primárně s jedinou funkcionalitou, až po komplexní systémy pro řízení velkých projektů.

Verzovací systémy (SVN/Git)

Verzovací systémy jsou nástroje pro udržování verzí zdrojových kódů a dokumentů, které také umožňují mnoha vývojářům spolupracovat na společném projektu. Koncept těchto systémů je postaven na klient-server architektuře, kde server uchovává aktuální verzi projektu a všechny změny od všech řešitelů. Klienti se k serveru připojují a udržují kód v konzistentním stavu mezi serverem a svým kódem.

Několik programátorů pak může zároveň pracovat na společném kódu. Každý edituje svoji vlastní kopii kódu a odesílá modifikace zpět na server. Tím, že je k dispozici i celá historie projektu, lze v případě potřeby vrátit změny, které z nějakého důvodu mohly způsobit problém.

Mezi nejpoužívanější verzovací systémy dnes patří např. Subversion (SVN) nebo Git. Oba nástroje se od sebe v určitých ohledech liší, na projekt IFJ však postačí libovolný z nich. Pro školní projekty můžete využít server Merlin, kde lze založit SVN/Git repozitář projektu.

http://en.wikipedia.org/wiki/Apache_Subversion

http://en.wikipedia.org/wiki/Git_%28software%29

http://merlin.fit.vutbr.cz/wiki/index.php?title=SVN_tutori%C3%A1l

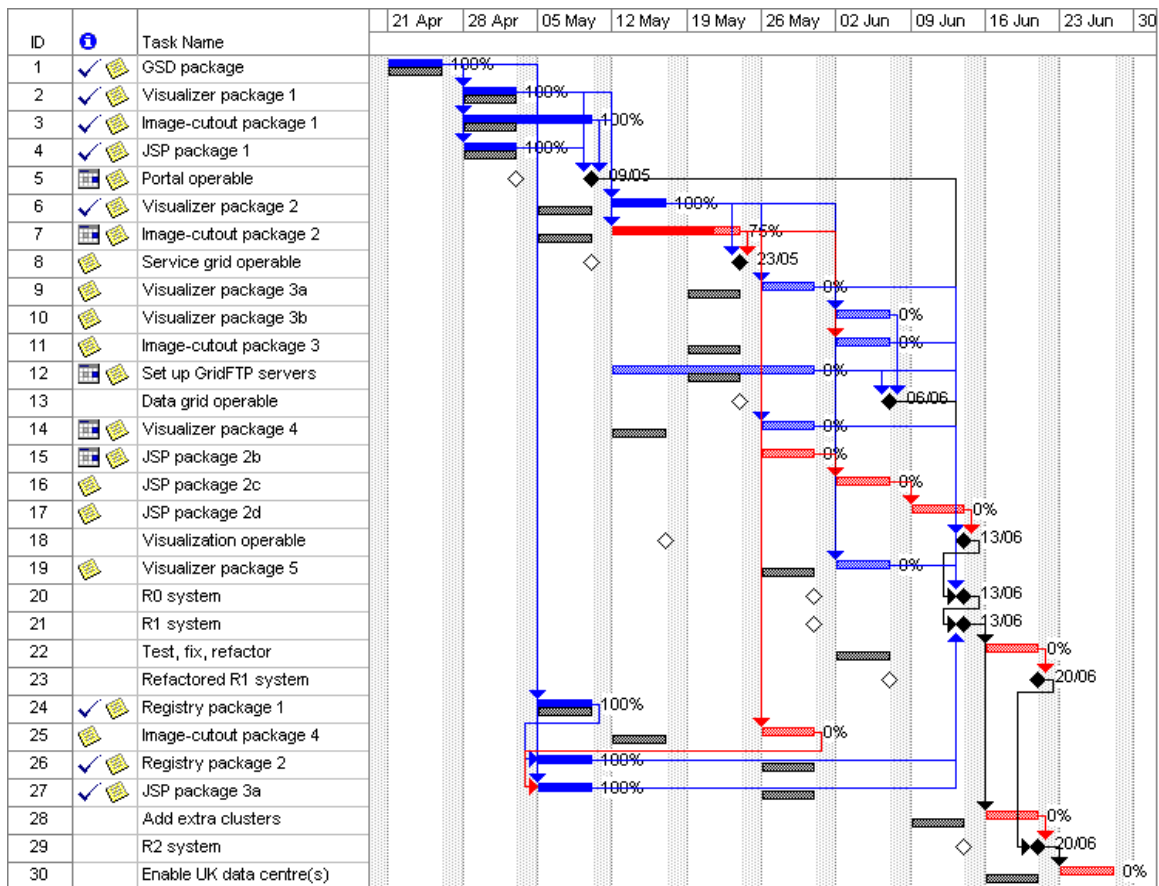
<http://www.fit.vutbr.cz/CVT/linux/faq.php#faq9>

Microsoft Project

Tento nástroj od společnosti Microsoft poskytuje komplexní prostředí pro efektivní správu projektu jako celku. Od přidělování úkolů, přes sledování časové osy až po případné finanční bilance projektu.

Poznámka: Studenti FIT VUT Brno mají v rámci akademické licence od společnosti Microsoft dostupnou poslední verzi tohoto nástroje.

http://merlin.fit.vutbr.cz/wiki/index.php/Dream_Spark



3 Ukázka MS Project

<http://office.microsoft.com/cs-cz/project/default.aspx>

Trac

Je webový opensource nástroj pro management projektu. Trac kombinuje wiki, verzovací systém spolu s tzv. tiketovým systémem. Ten spočívá v tom, že jednotlivé fáze a části projektu jsou označeny tikety, které mají vždy popis, termín a autora. Vyřešením tiketu a nahráním příslušné úpravy se tiket uzavře a příslušný vedoucí má přehled o postupu projektu.

<http://trac.edgewall.org/>