VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

EFEKTIVNOST STRATEGIÍ PRO ZÁLOHOVÁNÍ DAT

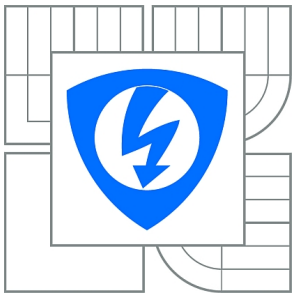
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE MARTIN ŠINDLER
AUTHOR

BRNO 2015

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF TELECOMMUNICATIONS

EFEKTIVNOST STRATEGIÍ PRO ZÁLOHOVÁNÍ DAT

EFFECTIVITY OF STRATEGIES FOR DATA BACKUP

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

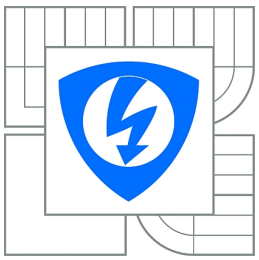
AUTOR PRÁCE — MARTIN ŠINDLER

AUTHOR

VEDOUCÍ PRÁCE — doc. Ing. KAREL BURDA, CSc.

SUPERVISOR

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student:	Martin Šindler	***ID:***	154667
Ročník:	3	***Akademický rok:***	2014/2015

NÁZEV TÉMATU:

Efektivnost strategií pro zálohování dat

POKYNY PRO VYPRACOVÁNÍ:

Nastudujte a popište současný stav teorie a praxe zálohování dat. Obsahem práce bude vysvětlení základních pojmů a matematického aparátu pro kvantitativní popis zálohování dat, popis známých strategií zálohování dat, vysvětlení rotačních schémat a popis prakticky používaných typů úložišť a systémů pro zálohování dat. Dále vyberte parametry pro hodnocení efektivnosti různých strategií zálohování, volbu těchto parametrů zdůvodněte a uveďte matematické modely pro jejich zjišťování. S jejich pomocí zjistěte hodnoty vybraných parametrů pro různé strategie zálohování a rotační schémata. Získané výsledky porovnejte a zhodnoťte. Na základě zjištěných zákonitostí navrhněte a realizujte softwarovou podporu pro výběr optimálního způsobu zálohování dat.

DOPORUČENÁ LITERATURA:

[1] Burda, K.: Mathematical model of data backup and recovery. International Journal of Computer Science and Network Security. 2014, roč. 13, č. 10, s. 16-25.
[2] Frisch A.: Essential System Administration. O'Reilly Media, Sebastopol 2002.

Termín zadání:	9.2.2015	***Termín odevzdání:***	2.6.2015

Vedoucí práce: doc. Ing. Karel Burda, CSc.
Konzultanti bakalářské práce:

doc. Ing. Jiří Mišurec, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

ABSTRAKT

Tato bakalářská práce řeší efektivnost záloh využitím matematického modelu, který určuje celkový objem záloh a objem dat nutných k obnovení daného datového prostoru, dále určuje dostupnost záloh při použití jednotlivých rotačních schemat. Ve vypracování jsou odvozeny jednotlivé matematické modely, které se dále dají využít pro určování nároků základních typů záloh. Modely jsou založeny na pravděpodobnosti změny dat v časovém intervalu určující zálohovací cyklus. Modely jsou dále implementovány do aplikace, která umožnuje výběr nejvhodnějšího rotačního schematu.

KLÍČOVÁ SLOVA

matematický model záloh, efektivnost záloh, strategie záloh, obnovení dat

ABSTRACT

This bachelor´s project is focused on efficiency of backups using mathematical models, which calculates overall capacity needed for backup and capacity required for complete recovery, it also compares availability of rotation schemes. Models are based on propability of data change in certain time interval specified as backup cycle. Models are implemented to the application, which can be used for choosing the best rotation scheme.

KEYWORDS

mathematical model of backups, backup efficiency, backup strategy, data recovery

ŠINDLER, Martin *Efektivnost strategií pro zálohování dat*: bakalářská práce. BRNO: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2015. 72 s. Vedoucí práce byl doc. Ing. Karel Burda, CSc.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Efektivnost strategií pro zálohování dat“ jsem vypracoval(a) samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

BRNO ....

podpis autora(-ky)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu doc. Ing. Karlu Burdovi, CSc. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

BRNO . . . . . .

. . . . . .

podpis autora(-ky)

OBSAH

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

1 ÚVOD

Tato bakalářská práce pojednává o problematice efektivnosti zálohování, popisuje a vysvětluje techniky zálohování a postupnou analýzou identifikuje výhody a nevýhody jednotlivých způsobů záloh. Dříve, než budou probrány konkrétnější informace, je nutné zamyslet se obecně nad zálohami a poukázat na motivaci, tedy proč vůbec zálohovat.

Co může představovat pojem zálohování? Zálohování se dá definovat jako zpracování a uložení dat do náhradního (záložního) prostoru takovým způsobem, aby bylo možné tato data obnovit do stavu, ve kterém byla v určitém čase v minulosti. Tento stav v určitém čase v minulosti je požadován s jistou přesností, která je dána požadavky na obnovu a může hrát klíčovou roli při zvolení zálohovací technologie.

Zálohování vychází zejména z hodnoty či ceny informací. Jde o informace, které jsou unikátní a nedají se znovu vytvořit, nebo je nové vytvoření finančně a časově velmi náročné. V takových situacích je investice do zálohování opodstatněná a výrazně snižuje rizika plynoucí ze ztráty výše zmíněných informací. Předpokladem je ovšem cenová výhodnost zálohování oproti novému vytvoření zálohovaných informací.

Cenová výhodnost může být stanovena i v jiných hodnotách, např. pro subjekt poskytující službu je důležité, aby byly ztracené informace k dipozici zákazníkovi za co nejkratší dobu, nehledě na to, jestli jsou data důležitá, či nikoliv. Dokonce může jít i o situaci, kdy zákazník za zálohy neplatí, ale pro subjekt poskytující úložiště (a zálohy) je důležité neztratit vůči zákazníkovi suverenitu a dobré jméno.[1].

Při hodnocení způsobu záloh musí být zodpovězeny alespoň tyto otazky:

- Co plyne ze ztráty informací? Je vůbec výhodné tyto informace zálohovat?
- Jaké jsou finanční prostředky pro realizaci záloh na základě požadavku?
- Jak velký objem dat se má zálohovat?
- Jak často má docházet k zálohám?
- Jaké jsou požadavky na rychlost obnovení?
- Jak dlouho má být záloha k dispozici?

První dva body by měly být v souladu a měly by korespondovat. V praxi se lze běžně setkat se situací, kdy jsou data hodnocena jako relativně cenná, finanční prostředky jsou ovšem omezené. Toto finanční omezení se projevuje v dalších zmíněných bodech a dochází pak k diskuzi s vlastníkem těchto dat (většinou jde o stejný subjekt, který financuje zálohování).

[1]Např. společnost poskytující mailový účet zdarma nechce dopustit, aby byla data v mailové schránce ztracena, i přes to, že společnost k obnově nic neváže.

Pokud se má nějaká činnost v IT považovat za nejméně oblíbenou, budou to právě zálohy. Jde o poměrně rutinní práci, která nemá přímou viditelnost pro vlastníky dat. Z dlouhodobého hlediska může tato rutina při neprofesionálním přístupu vést k částečnému či úplnému ukončení provádění záloh, což může být ovšem zjištěno až při nutnosti obnovy. Takové nedostatky mohou následně vést až k likvidaci společnosti, která tak přišla o kritická data nutná pro další funkci podnikání.

Právě proto, že zálohování je tak nezáslužná a přitom kritická činnost, musí se na její realizaci klást dostatečný důraz. Zálohování se totiž neprovádí na základě požadavků IT, zálohuje se na základě požadavků od samotného zákazníka či podnikové politiky.

Dalším neméně důležitým důvodem provádět vhodně zálohy je legislativní část, kdy jsou zálohy pro podniky ze zákona povinné (požadavky se ale v takovém případě odvíjejí od konkrétní legislativy země). Legislativní požadavky mohou být v některých situacích mnohem důležitější než zálohování z pohledu podniku. Běžným příkladem může být provoz poskytování internetového připojení. Samotný poskytovatel většinou nemá důvod zálohovat logy z činností zákazníků. Motivací může být poskytnutí těchto informaci při vyšetřování trestné činnosti, kdy je poskytovatel povinen informace poskytnout a zálohy jsou pak v jeho zájmu.

Požadavky na obnovu jsou den ode dne sofistikovanější, a tak se hledají stále efektivnější způsoby zálohování. Zvyšuje se nejen množství dat na úložištích, ale také požadavky na co nejrychlejší obnovu dat. Současným trendem je plná automatizace zálohování a umožnění koncovému klientovi provádět základní obnovy jednotlivých souborů bez další nutné spolupráce s IT. Například se může jednat o možnost procházet v čase verze souborů na sdíleném úložišti.

Velkým trendem je také ukládání dat do geograficky oddělených lokací. Data se ukládají buď na jinou lokalitu firmy, nebo se zálohovaná data replikují do prostředí třetí strany, přičemž většinou jde o poskytovatele specializovaného na daný typ služby [2].

Jak již bylo zmíněno, finanční prostředky jsou jednou z hlavních otázek při volbě záloh. Efektivní zálohy ovšem nejsou snadným úkolem a zvolení vhodného způsobu pro konkrétní situaci vychází z osobních zkušeností navrhovatele. V současnosti jsou požadavky na zálohy natolik různorodé, že vhodná volba zálohy vyžaduje dlouholetou zkušenost z dřívějších návrhů.

Co bude tedy obsahem této práce? V teoretické části budou zmíněny základní informace nutné pro obeznámení se s touto tématikou, budou zde uvedený jednotlivé typy metod pro zálohování. Na základě těchto metod budou popsány strategie zálohování. Budou zmíněna základní používaná média, na která se běžně zálohuje.

[2] Obecně jsou taková řešení označována jako cloudová.

Ve vypracování se následně zvolí vhodná kritéria pro porovnávání záloh. Vytvoří se matematické modely, které budou schopny na základě vstupních informací z těchto kritérií porovnat efektivnost záloh. Zobecněním těchto výsledků bude možné určit nejvhodnější strategii zálohování pro běžné podmínky.

V další části této práce bude realizace aplikace analyzující data určená k zálohování. Aplikace zhodnotí míru změn na datovém úložišti a vypočítá předpokládanou náročnost jednotlivých typů zálohovacích strategií. Aplikace bude vytvořena tak, aby byl výstup jednoduše čitelný a použitelný pro další zpracování.

Realizace jednotlivých částí této bakalářské práce by měly dopomoct čtenáři k ucelení informací o zálohovacích strategiích, přičemž se některé myšlenky dají aplikovat i na jiná odvětví. Mohou tak dopomoct k jinému pohledu na uchovávání a zálohování dat i v běžném životě. Jeden z citátů, který se s těmito myšlenkami velmi dobře ztotožňuje, zní následovně:

> „I am prepared for the worst, but hope for the best.“
>
> — Benjamin Disraeli

2 TEORETICKÁ ČÁST

V této kapitole bude probrán potřebný matematický model pro následující výpočty a odvození, dále budou rozebrány obecné principy základních typů záloh a jejich odlišnosti v případě záloh souborů, databází a bloků dat.

Budou zde také rozebrány algoritmy pro výměnu (rotaci) médií, které do značné míry ovlivňují efektivitu zálohování a obnovování. Následně bude uveden přehled typicky používaných typů médií, jejich praktické použití v dnešním světě a případně budou případně poznamenány výhody specifických řešení v určitých případech.

2.1 Zálohování

Zálohování patřilo k jedné z primárních zodpovědností pro IT oddělení od té doby, co měly počítače schopnost uchovávat data. Už od dob sdílených sálových počítačů byl efektivní způsob zálohování důležitým faktorem, zejména díky nestabilitě operačního systému i hardwarovým chybám. Tyto důvody přetrvávají i v současnosti. Dnešní způsob práce se ale výrazně změnil. Stejně tak i technologie ukládání dat, což vedlo k dalším výzkumům zaměřeným směrem efektivnosti záloh.

Z mnoha důvodů jsou zálohy systémů stále aktuálním tématem a i dnešní řešení záloh se nedají považovat za konečná. Hodně nedořešených problémů souvisí především s heterogenním prostředím, kde jsou systémy s různými operačními systémy, od různých výrobců, obsahující různé formáty dat, ukládané lokálně či vzdáleně, s různou důležitostí jednotlivých dat.

Tato část rozebere pojem zálohování z různých úhlů, podle používaných algoritmů výměny médií a algoritmů záloh, podle výběru médií, podle způsobů zálohování různých typů dat až k základním realizacím z praxe.

2.1.1 Typy záloh

Jak popsal Nelson, S. ve své knize [8] (ze které budu vycházet také při rozdělení typů záloh), zálohy jsou časové snímky z určitého časového období, uloženy ve všeobecně známém formátu, které jsou dostupné po určité časové období, během kterého se předpokládá užitečnost těchto dat.

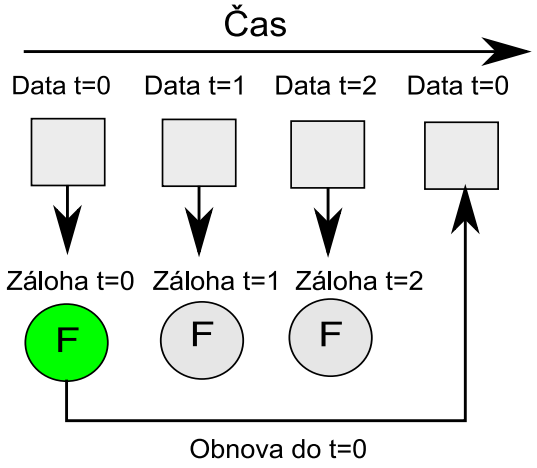
Mohou být vytvořeny různé typy záloh. Nejčastější jsou plné zálohy „full backup“, které obsahují kompletní snímek. Dalším hojně používaným způsobem je přírůstková záloha „incremental backup“. Méně používanou, přesto důležitou, metodu zálohy představuje rozdílová záloha „differential backup“.

Plná záloha

Tato metoda je nezávislá na předchozích zálohách a obsahuje veškerá data nutná k obnově (návratu) do stavu dat v období zálohy.

Je to nejjednodušší metoda záloh a dá se připodobnit k situaci, kdy si běžný uživatel uloží dokument na další médium pro zajištění dat v případě výpadku primárního zdroje. Typickým příkladem může být dodatečné uložení souboru na USB klíčenku. Pokud by bylo úkolem zazálohovat data o velikosti 20 TB, celková záloha za 10 dní by byla 200 TB dat.

Velkou výhodou plné zálohy je snadná obnovitelnost bez nutnosti kombinovat obnovu s dalšími médii z jiných záloh. Podstatnou nevýhodou je ovšem velikost záloh, která je identická se zálohovaným prostorem.[1]



Obr. 2.1: Plná záloha a obnova

Velikost plných záloh – Je přímo určena množstvím dat. U každého typu zálohy bude naznačen výpočet velikosti zálohy, který bude později použit při odvozování. Matematické vztahy primárně vychází z práce pana docenta Burdy[2].

Každa datová $n$tá jednotka bude označovat d_n. Pokud půjde o určení konkrétní datové jednotky v čase i, pak bude označení datové jednotky $d_n(t_i)$.

Označení dat potom bude následující:

$$D_{(t_i)} = [d_1(t_i), d_2(t_i), ..., d_n(t_i)] = [d_x(t_i)]_{x=1}^{n} \tag{2.1}$$

, kde x reprezentuje pořadové číslo datové jednotky v datech.

[1] Tento předpoklad nemusí být vždy pravdivý, pokud je zálohovací aparát schopný komprimovat data nebo dochází během procesu zálohování k deduplikaci.

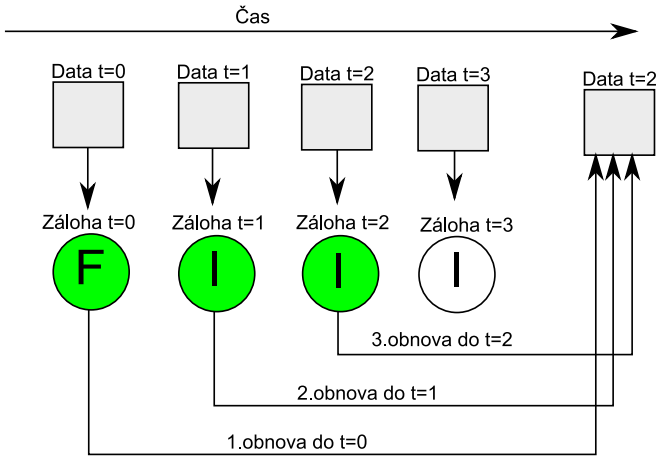
Na základě informací o plné záloze 2.1.1, je zřejmé, že se plná záloha z času i může značit $D_{(t_i)}$. Pro jednoznačné určení, že je jedná o plnou zálohu a ne o data samotná, pro odlišení tedy bude plná záloha označována $F_{(t_i)}$. Tedy:

$$F_{(t_i)} = D_{(t_i)} = [d_1(t_i), d_2(t_i), ..., d_n(t_i)] = [d_x(t_i)]_{x=1}^{n} \tag{2.2}$$

Přírůstková záloha

Tato metoda záloh zaznamenává pouze data, která se změnila od jakékoliv poslední zálohy. Spolu s plnou zálohou představují nejčastější způsob zálohy dat. Tento způsob zálohy zaznamenává pouze data změněná od poslední zálohy. Při využití předcházejícího příkladu, kdy je cílem zazálohovat 20 TB prostor, přitom změna dat za 1 den je 1 TB. První záloha by měla velikost celých 20 TB, protože se žádná záloha dříve neuskutečnila. Další každý den by byla přírůstková záloha pouze 1 TB. Za 10 dní by byla celková kapacita záloh velká 29 TB (20 TB první záloha a 9 x 1 TB ostatní přírůstkové zálohy).

Asi největší nevýhodou přírůstkové metody je problematická obnova, kdy pro obnovu musí být ve správném časovém sledu obnovena data ze všech médií od nejbližší plné zálohy (nebo první přírůstkové zálohy, pokud před tím žádné zálohy nebyly provedeny).



Obr. 2.2: Přírůstková záloha a obnova

Matematické vyjádření zálohy se odvíjí od změn provedených na datech v určitém časovém intervalu. Pro velikost přírůstkové metody musí být nadefinováno nové označení $Inc_{(t_i,t_j)}$, kde interval (t_i, t_j) určuje datové jednotky změněné v tomto období. Označení pro jednotlivé datové jednotky změněné v daném intervalu potom bude $inc_x(t_i, t_j)$. Pokud byla daná datová jednotka nezměněná, $d_x(t_j) = d_x(t_i)$, pak

je daná datová jednotka nula. V ostatních případech dojde k zazálohování datové jednotky.

Pro přírůstkovou zálohu pak platí:[2]

$$Inc(t_i, t_j) = [inc_x(t_i, t_j)]_{x=1}^{n} \tag{2.3}$$

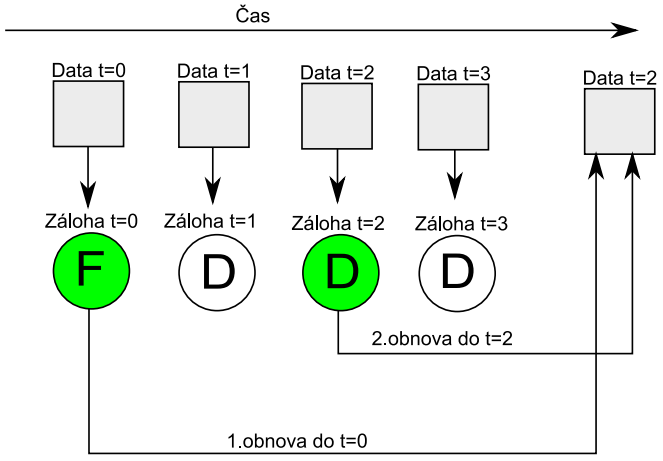
Rozdílová záloha

Při rozdílové záloze se zaznamenávají pouze změněná data od poslední plné zálohy. Někdy je označována jako komutativní přírůstková metoda.

Tato metoda je určitým kompromisem mezi plnou a přírůstkovou metodou. Není tak populární jako předchozí dvě metody, protože je vhodná jen pro situace, kdy dochází k relativně malé změně dat.

V případě větších změn výsledná záloha nemusí být tak výhodná oproti plné záloze. Velkou výhodou oproti přírůstkové metodě je jednodušší obnova, kdy pro obnovu všech dat postačuje poslední plná záloha a poslední rozdílová záloha.

Pokud uvedeme předchozí příklad, zálohy za 10 dní by měly celkovou velikost 65 TB (20 TB první záloha, 1 TB 2. den, 2 TB 3. den. . . 9 TB 10. den), přitom pro obnovu by stačila záloha z 1.dne a záloha ze dne, kdy chceme data obnovovat.



Obr. 2.3: Rozdílová záloha a obnova

Matematické vyjádření rozdílové zálohy je v podstatě totožné s přírůstkovou metodou. Jediným rozdílem je časový interval, kdy musí být provedena plná záloha v čase i, tj. $F_{(t_i)}$, přitom neexistuje žádná $F_{(t_k)}$, pro které by platilo: $i < k < j$, pak platí $Dif(t_i, t_j) = Inc(t_i, t_j)$.

Z výše uvedeného vyplývá, že první přírůstková záloha za plnou zálohou je identická s případnou rozdílovou zálohou, pro kterou platí, že mezi touto rozdílovou

zálohou a danou plnou zálohou neproběhla jiná plná záloha. Pak platí:[2]

$$Dif(t_i, t_j) = [dif_x(t_i, t_j)]_{x=1}^{n} \tag{2.4}$$

Víceúrovňová záloha

Toto řešení představuje jiný přístup než již zmíněné zálohy. Následně popsané zálohy ale víceúrovňové zálohy využívají, proto je nutné zmínit jejich princip. Obecně je víceúrovňová záloha pouze prostředek pro realizaci složitějších modelů. Pracuje na principu, kdy se zálohují data na základě závislosti předchozí nižší úrovně, přičemž nejčastější jsou úrovně 0 - 9.

Úroveň 0 představuje plnou zálohu (pro tuto úroveň neexistuje referenční záloha) a úroveň 1 je stejná jako klasická přírůstková metoda, tedy zazálohuje změny z poslední přírůstkové metody. Je však důležité podotknout, že ve víceúrovňové metodě je záloha závislá na záloze z nižší úrovně.

Příkladem může být situace, kdy je v pondělí provedena záloha úrovně 0. V úterý se provede záloha úrovně 1, která bude obsahovat změny od nejbližší nižší úrovně, tedy úrovně 0. Ve středu se provede záloha úrovně 2, ta bude obsahovat změny oproti úterní záloze úrovně 1. Dále bude provedena záloha úrovně 3 ve čtvrtek. Pokud bude provedena v pátek záloha úrovně 2, bude obsahovat rozdíl oproti nejbližší nižší záloze, tedy změny od úterní zálohy úrovně 1.

Pokud by bylo nutné vytvořit matematický model pro víceúrovňovou zálohu, zálohy s úrovní 0 by byly identické s plnou zálohou, tedy o $F_{(t_i)}$.

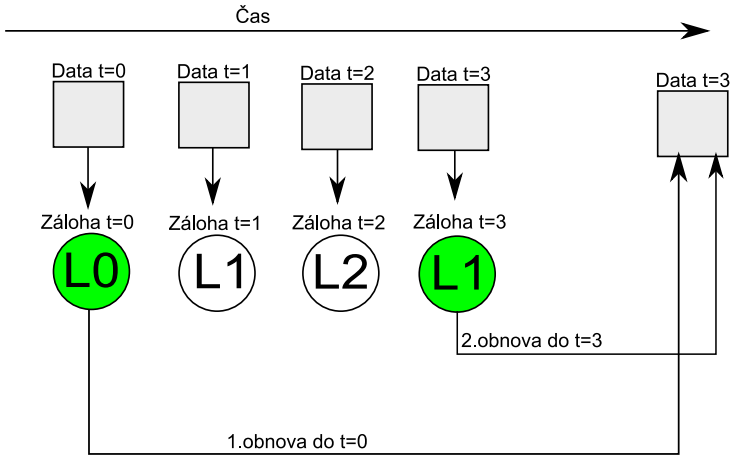
V ostatních případech záleží na tom, jestli je provedená předcházející záloha nižší úrovně nebo ne. Pokud je předcházející úroveň nižší, pak stávající záloha představuje přírůstkovou zálohu $Inc_{(t_i,t_j)}$, kde t_i a t_j jsou časové okamžiky předcházející a současné zálohy.

Jestliže je předcházející úroveň vyšší, pak stávající záloha není vztažena k předcházející záloze, ale k jakékoliv záloze, která má nižší úroveň. Takový popis koresponduje s rozdílovou zálohou $Dif_{(t_i,t_j)}$, kde t_i a t_j jsou časové okamžiky nejbližší nižší zálohy a současné zálohy.

Konsolidovaná záloha

Jedná se o způsob zálohování, který se snaží minimalizovat počet uskutečněných plných záloh, přičemž jsou plné zálohy k dispozici i z období provádění přírůstkových nebo rozdílových záloh.

Na první pohled se popis této situace může zdát jako protichůdný. Konsolidovaná záloha je většinou tvořena následujícím způsobem. [12]



Obr. 2.4: Víceúrovňová záloha a obnova

Vytvoří se plná záloha, která je většinou spojená s jinou údržbou znemožňující uživatelům s daty pracovat. Po dokončení plné zálohy se již dále provádí přírůstkové nebo rozdílové zálohy běžným způsobem.

Jakmile je nutné provést plnou zálohu, spustí se proces konsolidace, kdy se z iniciační plné zálohy a následujících záloh vytvoří virtuálně další plné zálohy.

Tento proces může být časově náročný, což však nemá vliv na běh běžných záloh.

Až se tímto způsobem vytvoří nová, plná záloha, může být využita stejným způsobem, jakým byla využita původní, iniciační plná záloha.

Přírůstkové a rozdílové zálohy jsou zpravidla ukládány na dočasný diskový prostor, který ulehčuje provedení konsolidace dat do plné zálohy. Zjednodušeně si lze představit, že jsou data vytvořena obnovou na dočasné úložiště, ze kterého se vytvoří virtuální záloha.

2.1.2 Zálohování datových jednotek

V této části budou rozebrány základní známé způsoby záloh dat podle povahy dat. Každý typ dat vyžaduje jiný přístup při zálohování, který co nejvíce zefektivní jak rychlost, tak další důležité aspekty dle požadavků prostředí. Nedá se však říct, že by byl jakýkoliv z dále rozebraných způsobů zálohování přežitek. Některé metody se stále využívají zejména kvůli jejich jednoduchosti (jak zálohy, tak obnovy). V této části budu vycházet z informací v knize Nelsona, S. [8] .

Zálohování souborů

Někdy také nazývána jako File Level Backup. Jedná se o snad nejjednodušší metodu, která pracuje asi nejpřímějším možným způsobem.

Obecně se tato metoda integruje do systému velice snadno pomocí instalace agenta, který komunikuje se systémem a zefektivňuje přístup k souborům. Ve výchozím stavu zálohovací program v tomto případě ukládá do zálohy všechny soubory viditelné na souborovém systému a není třeba dalšího nastavení.

Velkou výhodou této metody je jednoduchá dodatečná nastavitelnost výjimek či explicitně vybraných souborů. Dále se dá nadefinovat jiný způsob záloh pro specifické složky, např. způsobem uvedeným v tabulce 2.1. Složky se systémem a profily uživatelů se budou zálohovat jednou za týden v plné záloze. Ostatní dny budou profily zálohovány rozdílovou zálohou, systém už se ale zálohovat nebude. Zbylá data se budou zálohovat plnou zálohou každý zálohovací den.

Tab. 2.1: Seznam politik pro zálohování

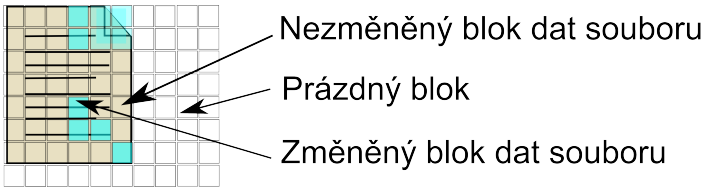
Složka	Typ zálohy	Den
C:\Users	Přírůstková	Po-Pá
C:\Users	Plná	So
C:\Windows	Plná	So
Ostatní	Plná	Po-So

Zálohování datových bloků

Zálohování datových bloků je v posledních letech více prosazováno. Tento způsob zálohování začínají preferovat i výrobci operačních systémů. Je vhodnější pro datová úložiště s velkým množstvím dat nebo velkým množstvím souborů.

Při blokové záloze se nezálohují jednotlivé soubory, ale data se načítají po blocích uložených v souborovém systému. Blokový přístup si zajišťuje agent nainstalovaný na operačním systému, který kontroluje přístup k datovým blokům.

To umožňuje lépe optimalizovat zejména přírůstkové a rozdílové zálohy. Do zálohy se ukládají jen části souborů, které se změnily, nikoliv celé soubory, což je znázorněno na obrázku 2.5.



Obr. 2.5: Souvislost mezi bloky dat a soubory

Dále se zvyšuje i rychlost samotného zálohování. To je patrné zejména u zálohování na pásky, kdy se využívá téměř konstantního toku dat na páskové médium, které v takové situaci vykazuje nejvyšší rychlost zápisu. Bloková obnova může být v některých případech problematická, zejména při obnově jednotlivých souborů. V takové situaci musí zálohovací software alokovat všechny bloky z předešlých záloh tak, aby byl schopný obnovit celý soubor. Obnova může být pak znatelně delší než v případě obnovy pomocí zálohování souborů. Bloková záloha se někdy kombinuje s procesem deduplikace, kdy se stejné bloky na úložišti ukládají pouze jednou a na stejné bloky se příště už jen využívá unikátní odkaz. Tento princip se může využít i při záloze a stejné bloky se tak nemusí zálohovat, protože již existuje reference. Bloková záloha tímto způsobem snižuje redundanci.

Zálohování databází

Databáze jsou specifický typ dat vzhledem ke způsobu zpracování dat. Pro zálohování vyžadují jiný postup než klasický souborový systém nebo datové bloky. Existuje několik typů databází. Obecně lze považovat SQL databázi jako referenční databázový systém, který svou strukturou může zastřešovat další možná řešení. Základní strukturou SQL je datová databáze obsahující uložená data a transakční logy, kde se ukládají změny provedené v datové databázi. Záznamy v transakčním logu slouží pro obnovu v případě násilného přerušení procesu (typickým příkladem může být

výpadek elektrického proudu). Je tedy zřejmé, že je nutné zálohovat jako datovou databázi, tak i transakční logy pro zajištění případného obnovení do konzistentního stavu.

Pokud se transakční logy zálohují častěji než samotná databáze, pak se transakční logy dají použít pro obnovení dat v datové databázi do původního stavu a dají se tak využít jako další forma obnovy, dokud nedojde k celkové záloze datové databáze. Další možností zálohování je vytváření rozdílových záloh datové databáze. Opět zde ale platí nutnost zálohování transakčních logů pro zachování konzistence při případné obnově.

Pro zálohování databází se používají dva základní přístupy. První variantou je zálohování dat na lokální disky nebo sdílenou složku pomocí interních obslužných aplikací - utilit. Tyto utility si samy řídí přípravu databází pro zálohování a odmazání transakčních logů po úspěšné plné záloze.

Je zde také možnost zálohování agentem třetích stran. V tom případě ovšem musí agent zajistit spuštění skriptů na přípravu databáze k záloze, dále řídí spouštění skiptů nutných k návratu databáze do původního stavu, což může být v některých situacích jako nestabilní řešení.

Zálohování pomocí snímků

Zálohování pomocí snímků, které je rovněž označováno jako zálohování pomocí snapshotů, představuje způsob zálohování, který přebírá výhody blokových záloh a záloh databází. Podstata snapshotů spočívá v zamknutí zálohovaného disku pouze na čtení. Následně se všechny změny provádí do jiného umístění. Existují základní dva způsoby realizace.

Copy on Write – „CoW“ je metoda snímkování, kdy se po vytvoření snímku originální data ponechávají na fyzicky stejném místě do té doby, než přijde požadavek na změnu dat. Vytvoření snaphostu totiž pouze vytvoří kopii metadat na nové úložiště, která se odkazují na původní data. Vytvoření snapshotu je tedy provedeno rychle. Jakmile má dojít ke změně určitého časového úseku, překopíruje se tato část do prostoru vyhrazeného pro snímky a původní data se přepíší změněnými daty. Metadata snímku jsou upravována tak, že alokační tabulka metadat se odkazuje na původní data, pokud nedošlo k jejich změně. V případě změny se tabulka ve snímku upraví a část tabulky metadat ukazuje na překopírovaná data.

Výhodou tohoto principu je zachování konzistentních dat na primárním úložném prostoru a vytvoření snímku není pro systém patrné. Není tedy třeba provádět žádné další zásahy do systému úložiště.

Velkou nevýhodou je ovšem výkonnost, protože při požadavku na zápis se musí provést jedno čtení dat a dvojí zápis. Nejprve se zapisují originální data na alternativní umístění a následně se zapisují nová data na originální umístění.

Redirect on Write – RoW pracuje na podobném principu jako CoW, tedy při požadavku se provede logický snímek (jehož součástí je i vytvoření nového metadata souboru s alokační tabulkou bloků na disku) a při čtení se neprovádí žádné další akce. V případě požadavku na zápis se data ukládají do volného prostoru na úložišti do jiných bloků a metadata s alokační tabulkou bloku se upravují pouze v aktuální tabulce. Pokud je tu požadavek na obnovu, stačí použít starou alokační tabulku a systém je prakticky ihned obnoven.

Nevýhodou tohoto principu je fragmentace dat na úložišti, která je způsobená ukládáním dat po snaphotu do dalších bloků na stejném úložišti. Tento problém bývá většinou řešen dodatečnou defragmentační [1] službou na úložišti. Takto řeší situaci i společnost NetApp.[9][2]

Výhodou snapshotu je „odemknutí souboru“ pro čtení a zajištění toho, že se data nebudou při zálohování měnit. Zálohovací řešení musí technologii CoW využívat.

2.1.3 Způsoby rotování médií

V této části budou představeny základní způsoby strategií výměny zálohovacích médií. Zdrojem informací v této části bude publikace od Shimowski, R. a Schmied, W.[16], která velice dobře popisuje tři základní nejpoužívanější strategie. [3]

Rotace pěti pásek

Rotace pěti pásek je nejjednodušší a nejméně nákladnou pro prvotní implementaci. Jak už ze samotného názvu vyplývá, pro tuto zálohu je nutné použít pět pásek, přičemž každý všední den se použije jedna páska. Tento systém je samozřejmě možné velice snadno rozšířit na šestidenní či sedmidenní schéma. Zálohovací pásky jsou obvykle označeny dny v týdnu kvůli snadné identifikaci. První záloha vyžaduje plnou zálohu, následně se provádí během jednotlivých dnů rozdílová či přírůstková záloha. Poslední den v týdnu se provede plná záloha. Na první pohled jde o velice levné řešení, protože je vyžadováno mít pouze pět pásek. Pokud se tento systém rozšíří

[1] Defragmentace je přeskládání jednotlivých fragmentů data k sobě tak, aby při čtení nemusel disk hledat části souborů v jiném sektoru. Defragmentací se snižuje doba načtení souboru.

[2] Zmínka o NetApp je opodstatněná. Jde totiž o společnost, která má koncept RoW patentován.

[3] Pro zjednodušení modelu je předpokládáno, že na kapacitu jedné zálohy postačuje jedna páska.

na více týdnů, správa a množství pásek násobně roste, efektivita tohoto schematu pak klesá.

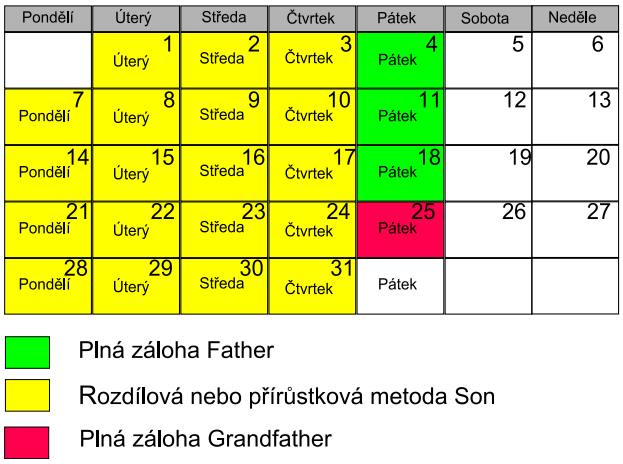
Obr. 2.6: Rotace pěti pásek

Grandfather, Father, Son

MetodaDědeček, Otec, Syn, která je označována zkratkou GFS, představuje jednu z nejpopulárnějších metod, které se dnes používají. GFS poskytuje historii záloh po celý rok. Tato výhoda je však vykoupena vyšší cenou, kdy je zapotřebí zachovat 20 záloh na jeden rok.

- **Syn** – Zálohy, které jsou rozdílové nebo přírůstkové se provádí v pondělí, úterý, středu a čtvrtek. Pro tuto zálohu je tedy zapotřebí 4 pásek.
- **Otec** – Záloha se provádí každý pátek kromě posledního pátku v měsíci. Čtyři pásky tedy obsahují plné zálohy, které poskytují zálohu po celý měsíc.
- **Dědeček** – Zálohy se provádí poslední pátek v měsíci. Těchto pásek je celkem 12, přičemž poskytují plné zálohy každého měsíce po celý rok.

Při první GFS záloze je provedena výchozí plná záloha, nehledě na to, o jaký den se jedná. To zajišťuje, že je k dispozici plná záloha v případě nutnosti obnovy před první plnou zálohou v GFS. Obrázek 2.7 ilustruje příklad rotace pásek GFS pro měsíc s 31 dny a čtyřmi pátky. Mezi plnými zálohami je možné kombinovat jakékoliv další zálohy dle požadavků.



Obr. 2.7: GFS rotace

Hanojské věže

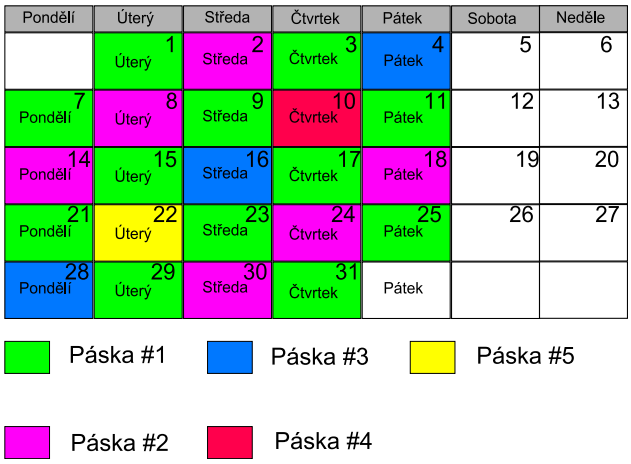
Metoda hanojských věží je založena na principu stejnojmenné hry. Jedná se o algoritmus, kdy se vzájemné zálohy vhodně prokládají a efektivně využívají daný počet zálohovacích médií. Počet pásek se dá dynamicky zvětšovat a zmenšovat, aniž by došlo k porušení rotace ostatních pásek.

Realizace této zálohy se ale liší. Některé zdroje uvádějí, že se jedná pouze o vhodně střídající se plné zálohy [16]. Jiné zdroje uvádějí, že jde o střídající se plné, rozdílové a přírůstkové zálohy realizované víceúrovňovou zálohou [13]. Z tohoto důvodu budou v práci popsaný oba principy, přičemž pro zjednodušení budou obě varianty používat pět zálohovacích médií. Způsob střídání pásek je velice podobný pro obě varianty. Nejprve bude popsána varianta s plnými zálohami. Střídání pásek je naznačeno na obrázku 2.8 a slovně je dá popsat následovně:

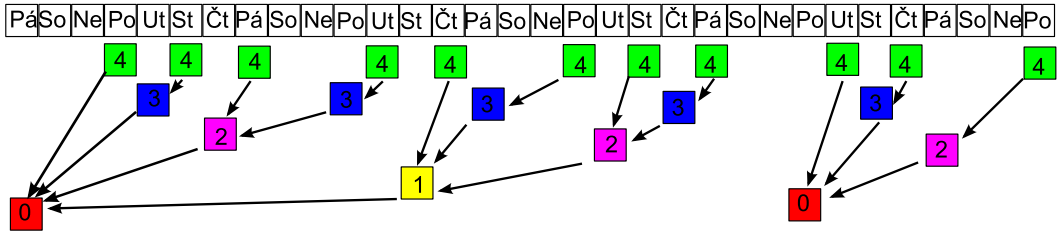
- Páska #1 Použitá pro ostatní dny než specifikované.
- Páska #2 Použitá každý 4. den.
- Páska #3 Použitá každý 8. den.
- Páska #4 Použitá každý 16. den. Střídá se s Páskou #4.
- Páska #5 Použitá každý 16. den. Střídá se s Páskou #5.

Hlavní nevýhoda této varianty spočívá ve vytváření plných záloh. Nopak velkou výhodou je jednodušší obnova, kdy není nutné kombinovat víc předchozích záloh.

Postup rozšiřování je v podstatě jednoduchý. Pokud by bylo požadováno rozšíření o další pásku, pásku #6, pak by tato páska byla použita na zálohu každý 32. den a střídala by se tak s páskou #5. Rotování ostatních pásek by zůstalo zachováno.



Obr. 2.8: Hanojské věže - plné zálohy



Obr. 2.9: Hanojské věže - víceúrovňové zálohy

Varianta pomocí víceúrovňové zálohy znázorněné na obrázku 2.9 je složitější. Pro jednotlivé zálohy jsou použity barevná rozlišení úrovní. Dále jsou úrovně posunuty i vertikálně a navíc jsou označeny číslem reprezentujícím úroveň zálohy.

Aby byla schémata porovnatelná, je využit model pěti úrovní a pěti pásek. Ve zvolené variantě jsou záměrně vynechány víkendy, a to opět kvůli jednoduššímu srovnání daného typu zálohy s ostatním typy.

Šipky naznačují návaznost jednotlivých záloh. Čtvrtá úroveň je podobná přírůstkové záloze, naopak nultá úroveň odpovídá plné záloze. Druhá, třetí a čtvrtá úroveň odpovídá určitým způsobem rozdílovým zálohám. Oproti běžným rozdílovým zálohám ovšem nemusí být vztaženy vždy jen k úrovni nula (plné záloze).

Schéma se postupně posunuje, nultá záloha se objevuje vždy posunutá v daném cyklu o jeden den. V tomto konkrétním případě se opakuje nultá záloha v pátek opět za šestnáct týdnů.

Jednotlivé případné způsoby obnov jsou patrné při průchodu cesty směrem proti šipkám. Např. pro obnovu z první středy je nutné obnovit zálohu z úrovně nula (páteční záloha), pak úrovně tři (záloha z úterý) a pak úrovně čtyři (záloha ze středy).

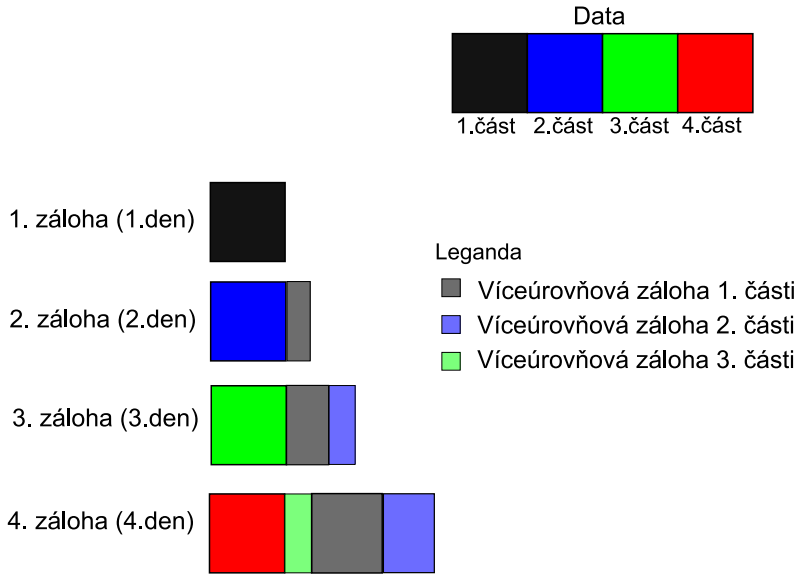
Algoritmus systému Amanda

Amanda, neboli Advanced Maryland Automated Network Disk Archiver, je jeden z nejznámějších open-source zálohovacích programů. Tento projekt je zajímavý, ale jen velice málo zdrojů popisuje samotný princip tohoto softwaru. Informace k tomuto softwaru jsou čerpány z [14] a [13].

Amanda byla vytvořena v roce 1991 na Marylandská univerzitě. Cílem bylo zálohování velkého množství pracovních stanic s jedním centrálním serverem. Amanda byla v roce 1999 registrována na SourceForge.Net, kde byla dále vyvíjena a upravována. V roce 2006 tento software používalo více než 20 000 organizací. Podporuje všechny běžně používané operační systémy, přesto klade největší důraz na Linux a také využívá komponenty původně vyvíjené pro tento systém.

Při vysvětlení principu zálohy bude vycházeno z předpokladu, že cyklus plných záloh je každý čtvrtý den. Umožňuje relativně variabilně nastavit způsob zálohování. Pro snadnější porovnání budou zvoleny nejzákladnější dva typy, které by měly představovat krajní případy implementace. Hlavní myšlenkou Amandy je maximální optimalizace záloh s rozprostřením velikosti záloh na celý zálohovací cyklus.

Existují základní dva způsoby režimu zálohy, přičemž jeden režim se spouští při prvním cyklu, druhý režim běží další týdny. Pro spuštění prvního cyklu se první den provede záloha poměrné části, v tomto případě první čtvrtiny. Další den se provede plná záloha další čtvrtiny a víceúrovňová záloha z první čtvrtiny. Další den se provede plná záloha třetí čtvrtiny a víceúrovňová záloha první a druhé čtvrtiny. Poslední den se provede plná záloha čtvrté čtvrtiny a víceúrovňová záloha první, druhé a třetí čtvrtiny, čímž se první režim dovrší a je k dispozici celá záloha dat.

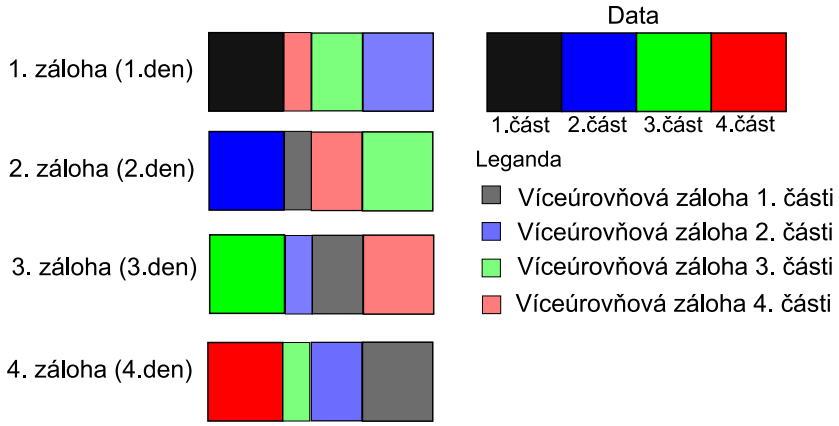


Obr. 2.10: První, iniciační režim

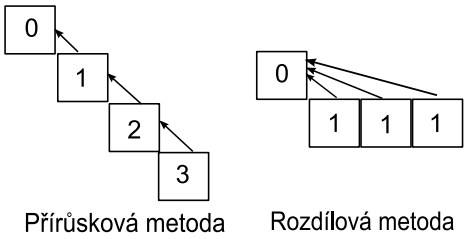
Druhý režim začíná ve chvíli dokončení prvního cyklu. V tomto režimu se provádí plná záloha jedné čtvrtiny dat a k tomu víceúrovňové zálohy zbylých tří čtvrtin.

To, jaká víceúrovňová záloha bude použita, záleží na konfiguraci. Tato práce se zaměří na limitní varianty, na situaci kdy jsou zálohy rozdílové a přírůstkové.

Konkrétní implementace bude dále podrobně popsána, v práci budou využity dvě metody víceúrovňové zálohy. V obou případech záloha spouští pro danou poměrnou část zálohu úrovně 0. Na obrázku 2.12 jsou znázorněny obě metody. V prvním případě následují každý další den zálohy dané části úrovně 1, dochází pak k rozdílovým zálohám. V druhém případě se každý den poměrná část zálohuje s vyšší a vyšší úrovní, pak dochází k přírůstkovým zálohám.



Obr. 2.11: Druhý, běžný režim



Obr. 2.12: Skládání víceúrovňových záloh

2.1.4 Média používaná při zálohování

Jedno z nejvíce opakovaných tvrzení ve způsobech zálohování se týká konce použitelnosti pásek. Nad jejich praktickým využitím se již několik let polemizuje. Realita je ovšem jiná. Magnetická zálohovací média mají několik zásadních předností oproti ostatním typům záloh. Mezi jejich největší přednosti patří:

- **Cena.** Disky se stejnou kapacitou je možné pořídit minimálně za dvojnásobnou cenu.
- **Mobilita.** Pásky jsou kompaktnější a lépe se přenášejí a uchovávají. Pásky se také dají jednoduše vyjmout ze zálohovací jednotky a uložit např. do sejfu v bance.
- **Kompatibilita.** Většina zálohovacích programů umí obhospodařovat fyzická média. Virtuální média vytvořená z disků většinou vyžadují speciální obslužný zásuvný modul (pokud vůbec existuje).

Přestože zálohování na pásky je stále primárním způsobem, v práci budou popsány také další varianty záloh na disky, disková pole, popř. cloudová řešení záloh.

Zálohy na pásky - LTO

Zálohovací pásky mají oproti diskům jeden zásadní rozdíl. Přístup čtení a zápisu na pásky se provádí sériově. Doba přístupu v případě hledání záznamu je tedy násobně nižší, v čemž je spatřována také jedna ze zásadních nevýhod. Samotné postupné čtení či zápis však může dosahovat rychlosti, která disky může i převyšovat. LTO technologie, poskytující jasné a přehledné řešení na trhu zálohování, byla vytvořena ve spolupráci společností Quantum, Hewlett Packard (dále jen HP) a International Business Machines (dále jen IBM) v roce 1998. Tyto společnosti vytvořily otevřený formát, který umožnil větší rozmanitost produktů založených na této technologii. Díky otevřenému standardu je tato technologie dále rozvíjena a vylepšována společnostmi, které se podílely na jejím vývoji. Každá nová generace je označena dalším pořadovým číslem, přičemž se kapacita zálohovací pásky v dané verzi vždy zdvojnásobí. Momentálně je na trhu LTO 6. generace. Ta poskytuje 6.25 TB zálohovacího prostoru na pásku. [4]

V tabulce 2.2 je patrný dosavadní vývoj jak rychlostí, tak kapacit v budoucích generacích LTO zařízení.[4]

Tab. 2.2: LTO - vývoj. (zdroj:www.lto.org)

Generace	Kapacita	Rychlost	V prodeji
3.	až 800 GB	až 160 MB/s	ano
4.	až 1.6 TB	až 240 MB/s	ano
5.	až 3 TB	až 280 MB/s	ano
6.	až 6.25 TB	až 400 MB/s	ano
7.	až 16 TB	až 788 MB/s	ne
8.	až 32 TB	až 1180 MB/s	ne
9.	až 62.5 TB	až 1770 MB/s	ne
10.	až 120 TB	až 2750 MB/s	ne

V dnešní době LTO plně nahradila produkt DLT, jehož výrobcem byla také společnost Quantum. Standard LTO se tak stal hlavním řešením pro zálohování na pásky. [5]

[4]Hodnoty kapacit jsou u LTO počítany s kompresí dat. Kapacita samotného média je v případě 6. generace přibližně 2.5 TB.

[5]V dnešní době většinu zálohování na pásku vytlačila technologie LTO. Alternativní zálohování jsou např. DDS, DAT, popř. DLT. Vzhledem k tomu, že se prakticky již nepoužívají, nebudou v tomto dokumentu dále popsány.

Zálohy na optická média

CD a DVD – CD byly používány k zálohovaní dat po relativně dlouhou dobu, avšak jejich kapacity nemohly konkurovat kapacitám zálohovacích pásek. S příchodem DVD se zálohování na optická media opět začalo používat (zejména v oblasti osobních počítačů).
Výhody optických médií:

- Dlouhá životnost v případě vhodné archivace médií
- Možnost náhodného přístupu v případě čtení
- Levné řešení pro zálohování dat z PC

Nevýhody optických médií:

- Životnost se při špatném uchování podstatně zkracuje
- Nemožnost přepisování dat
- Pomalý zápis na médium

V dnešní době již CD nehrají při zálohování podstatnou roli. DVD média je možné rozlišit na několik skupin podle technologie [14] :

- **DVD-RAM**. Tento formát se používá okrajově i přesto, že má velmi dobré vlastnosti z hlediska archivace. Hlavním problémem je ovšem nekompatibilita s běžnou DVD mechanikou.
- **DVD-R**. Toto médium není možné přepisovat, data se dají zapsat pouze jednou.
- **DVD-RW**. Toto médium je kompatibilní s běžným DVD.
- **DVD+RW**. Jde o přepisovatelné médium, které je kompatibilní s běžnými DVD mechanikami (ne však se zapisovatelnými mechanikami jiných typů). Některé DVD mechaniky potřebují upgrade firmware, aby byly schopné číst tato média.

Blu-Ray disk – BD je relativně novou technologií v optických médiích. Oficiální představení specifikace proběhlo v roce 2002, globálně byl ale přístupný až v roce 2006. BD má kapacitu 25 GB na jednovrstvé médium. Vysoká kapacita je umožněna hustším zápisem na povrch média, kdy jsou jednotlivé stopy vzdáleny pouze 0.32 μm. [1]

Nejvíce se prosazují BD přehrávače. Využití DB pro zálohy je relativně omezené hlavně kvůli ceně, chybovosti při zápisu a odolnosti média. Přesto může tato technologie být zajímavá zejména pro zálohování dat na osobním počítači.

Zálohy na disky

Se snižující se cenou disků může být vyhodnocena varianta zálohování na disky jako výhodná (i když v nejčastějších případech tomu tak zatím není). Existují základní dvě varianty záloh na disky, které budou popsány.

Zálohovací diskové jednotky – Zálohovací diskové jednotky byly ze začátku od výrobců zálohovacích programů velice málo podporovány. A pokud tito výrobci vůbec zálohu na disky podporovali, obecně se jednalo spíše o okrajové řešení, což se také projevovalo špatnou optimalizací pro disky (programy obecně přistupovaly k diskům jako k páskám). Postupem času, kdy se řešení záloh pomocí disků stávalo přístupnější alespoň pro enterprise řešení, si výrobci začali uvědomovat výhody použití disků. Mezi tyto výhody patří:

- Pomalý transfer dat ze zdroje nezpůsobí zpomalení zápisu, disky nejsou citlivé na rychlost přenášených dat. Pásky mají obecně problém s pomalým proudem zálohovaných dat, protože pásková mechanika musí neustále zastavovat a převíjet pásku na konec záznamu a čekat na další data. Disky takový problém nemají.
- Disky obecně podporují vícenásobný přístup, kdy jsou využívány více zdroji, aniž by došlo k citelnému zpoždění.
- Paralelní obnova souborů je možná díky rychlému přístupu a hledání v médiu.
- Doba načtení disku je v porovnání s načtením (převíjením) pásky zanedbatelná.
- Diskové zálohy mohou být optimalizovány pro další zálohu na pomalejší médium (pásku), např. deduplikací, spojením snímků. Dalším případem může být konsolidovaná záloha probírána v části 2.1.1.

Některá řešení, která získávají postupně na oblibě, nazírají na pásky jako na neperspektivní způsob záloh v krátkodobém horizontu. Tento pohled nemusí být ovšem zcela pravdivý. Jeden z důvodů, proč by taková situace neměla nastat, je cena disků, která je pořád alespoň násobná oproti standardním řešením pomocí pásek. Dalším důvodem může být nutnost uchovávat zálohy fyzicky mimo místo zálohy kvůli ochraně proti poškození při požáru nebo při útoku malwaru. Na diskové zálohy by se mělo nahlížet jako na doplňkové řešení, které může optimalizovat dobu záloh a obnov. Zálohy na diskové jednotky by měly být brány v úvahu pokud:

- Data nemohou být přenášena dostatečnou rychlostí k páskovým zálohám při zachování minimálního datového toku nutného pro konstantní zápis na pásku, ale zálohy z připojené zálohovací diskové jednotky na páskovou jednotku tento datový tok zajistí.
- Pomalý čas obnovy při více požadavcích je neakceptovatelný.

- Obnova dat musí začít na požádání, bez zbytečných prodlev.
- Je nutné provádět více obnov z jedné zálohy v jednom čase.

Virtuální páskové knihovny – VTL představují diskové pole, které se tváří pro zálohovací program jako pásková knihovna. Toto zařízení se v podstatě chová jako „black-box“[6], který vidí zálohovací program jako knihovnu. VTL mají oproti běžné páskové knihovně tyto výhody:

- Knihovna je schopná pracovat s jednou páskou souběžně ve více procesech obnovy.
- Načítání virtuálních pásek je bez prodlevy.
- Může být implementováno v situacích, kdy zálohovací program přímo nepodporuje zálohy na diskové jednotky.
- Umožňuje konfigurace, kdy je VTL distribuována pomocí SAN řešení a umožňuje tak centralizovat zálohy ze všech zálohovacích serverů na jedno úložiště.

Výhodami oproti zálohám na diskové jednotky není výkonnost (ta může být v obou řešeních při použití disků podobná). Jednou z výhod je, že zálohy ve VTL se chovají jako pásky. Virtuální pásky mohou být ve stavu jako vyjmuté, což zabraňuje případnému malwaru vymazání pásek. Další výhodou je chování knihovny, kdy koncept VTL může vyřešit nutnost koupě dalších licencí při přechodu na diskové zálohy. Přesto mají VTL několik nesporných nevýhod:

- Média nemohou být fyzicky odpojená a uložená mimo systém
- Hardwarová komprese dat většinou není možná bez dodatečného rozšíření a dalších finančních prostředků. To zvýhodňuje zálohu na pásky, protože VTL vyžadují prakticky dvojnásobný úložný prostor.[7]
- Knihovna se chová jako pásková knihovna, a proto nemusí umět využít některé výhody umožněné u záloh na diskové jednotky, jako např. paralelní obnovy.

Z výše uvedeného je patrné, že VTL může být v některých případech výhodné, ale stále trpí některými nedostatky z klasických páskových knihoven. [12]

Zálohování pomocí poskytovatelů třetích stran

Média využitá pro zálohovaní nemusí být vždy hmatatelná. Může se jednat o virtuální prostředí mimo lokální firemní prostředí.

Vzhledem ke zrychlování přístupu na internet začíná být rozdíl mezi řešením přímo v dané lokalitě (označováno jako on-premise) nebo řešením uloženým u poskytovatele služby na internetu (v cloudu) z pohledu zákazníka totožný. Navíc poskytovatelé mohou nabídnout řešení představující pro jednotlivce pro jednotlivce

[6]Tento výraz představuje zařízení, které má skrytý mechanizmus. Tento mechanizmus však pro samotný logický celek není podstatný.

[7]Hardwarová komprese umožňuje uložit na pásky přibližně dvojnásobné množství dat.

neúměrné prvotní investice, které by se nevrátily. Dalším důvodem může být řešení geografické nezávislosti pro chod podniku z pohledu případného výpadku [8]. Řešení záloh pomocí cloudu jsou pro menší podniky novinkou. Zatím se jednalo pouze o služby určené velkým společnostem. Poskytovatelé však začínají orientovat své produkty i na menší společnosti a v některých ohledech mohou být levnější než on-premise služby.

Z pohledu způsobu realizace je možné cloudová řešení rozdělit na dva základní typy:

- Programová (softwarová) řešení, kdy se za službu plati v licencích, zálohovacím prostoru u poskytovatele nebo propustností u poskytovatele. Obvykle se jedná o řešení vhodná pro menší společnosti.
- Fyzická (hardwarová) řešení, která přinášejí přidanou hodnotu a mohou sloužit jako podpůrná technologie pro další procesy ve firemním prostředí. Cloudové řešení je přístupné po instalaci fyzického zařízení v podniku.

Programová cloudová řešení záloh – Umožnují realizaci záloh bez nutných požadavků na koupení fyzického zařízení. Typickým příkladem takových služeb může být relativně nově zavedená služba od společnosti Microsoft. Jedná se o řešení Microsoft Azure, kdy zákazník může využít stávajícího programového vybavení od firmy Microsoft (Microsoft DPM) nebo výchozího zálohování integrovaného přímo v operačním systému Windows. To umožňuje za poplatek zálohovat data přímo do cloudu, cena se přitom řídí pouze od celkového množství uložených dat.[6]

Fyzická cloudová řešení záloh – Vyžadují pro realizaci zakoupení fyzického zařízení. Příkladem může být společnost Riverbed, která se zaměřuje zejména na propojení velkých datových center.

Tato společnost nabízí optimalizaci přenosů dat díky akceleraci služeb a optimalizaci síťového propojení mezi datovými centry mezi kontinenty. Díky těmto optimalizacím přes velké vzdálenosti je umožněno efektivní zálohování velkých dat pomocí sofistikovaného zařízení, které nahrazuje síťové zařízení připojené k poskytovateli internetu.

Toto zařízení optimalizuje data zazálohovaná ze serverů v daném datovém centru a komprimovaná a zašifrovaná data se posílají do úložiště v cloudu.

V případě nutnosti obnovy (ať už počítače, či serveru) je proces obnovy téměř automatizován díky softwaru uloženému na straně počítačů a obnova z jakéhokoliv stavu je provedena bez větších technických potíží.

[8] Někdy označována jako business continuity, tedy schopnost udržet si chod firemního prostředí v případě geografického výpadku podpůrných technologií.

Řešení rovněž umožnuje obnovu celého datového centra v nové lokaci. To umožňuje využití SteelHead (na obrázku 2.13) zařízení pro zálohu pomocí služby SteelFusion.[15] . Zařízení SteelFusion umožňuje „akcelerovat“ propustnost připojení využitím optimalizovaných síťových uzlů služby RiverBed pracujících mezi sebou na interním protokolu.

Obr. 2.13: Riverbed SteelHead EX 1160 (http://www.wansolutionworks.com/)

2.2 Matematický aparát

V této části bude vysvětlen matematický aparát nutný ke správnému odvození a pochopení matematického modelu pro zálohování. Velký důraz bude kladen zejména na pravděpodobnost, která je zejména v části odvození klíčová.

2.2.1 Základní definice

Náhodná veličina X

Je reálná funkce definovaná na množině všech elementárních jevů, která každému jevu přiřadí reálné číslo.

Hustota pravděpodobnosti náhodné veličiny X

definované na intervalu <a,b> je nezáporná, reálná funkce definovaná vztahem:

$$f(x) = \lim_{x \to 0} \frac{Px \leq X < x + h}{h}. \tag{2.5}$$

Distribuční funkce

Jde o reálnou funkci, která přiřazuje každé hodnotě x_i náhodné veličiny X pravděpodobnost, že X nabude hodnoty menší než toto x_i, se nazývá distribuční funkce F(x). Je definována vztahem:

$$F(x) = P(X \leq x) = \sum_{x_i < x} PX = x_i. \tag{2.6}$$

2.2.2 Exponenciální rozdělení

Toto rozdělení má spojitá náhodná veličina X, která představuje dobu čekání do nastoupení (poissonovského) náhodného jevu nebo délku intervalu (časového nebo délkového) mezi takovými dvěma jevy (např. doba čekání na obsluhu, vzdálenost mezi dvěma poškozenými místy na silnici).

Závisí na parametru λ , což je převrácená hodnota střední hodnoty doby čekání do nastoupení sledovaného jevu.

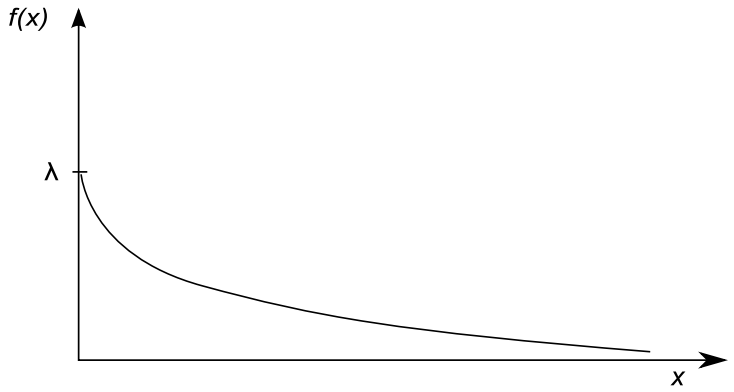
Náhodná veličina X má exponenciální rozdělení E (λ) právě tehdy, když je hustota pravděpodobnosti 2.14 dána vztahem:

$$f(x) = \begin{cases} 0 & pro \quad x < 0 \\ \lambda \cdot e^{-\lambda \cdot x} & pro \quad x \geq 0. \end{cases} \tag{2.7}$$

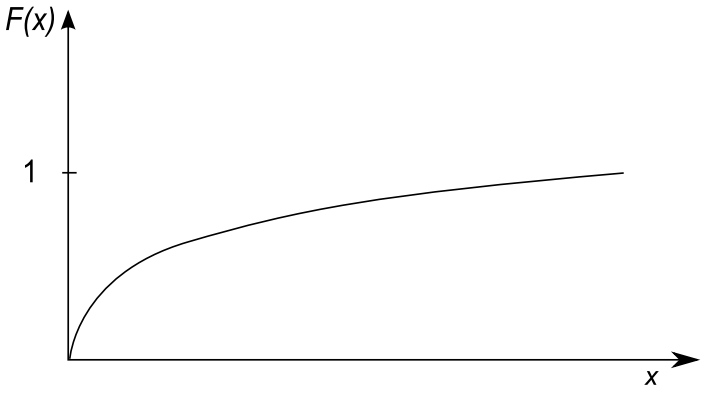
a distribuční funkce 2.15 je dána vztahem: [11]

$$F(x) = \begin{cases} 0 & pro \quad x < 0 \\ 1 - e^{-\lambda \cdot x} & pro \quad x \geq 0. \end{cases} \tag{2.8}$$

Dalším typickým příkladem využití exponenciálního rozdělení je porucha výrobků, kde se zanedbává předpoklad poruchy způsobené opotřebením. Výskyt poruchy se pak chová jako by si výrobek nepamatoval, jak dlouho již pracuje. Tento typ rozdělení se proto někdy označuje jako rozdělení „bez paměti“.



Obr. 2.14: Hustota pravděpodobnosti



Obr. 2.15: Distribuční funkce

3 ODVOZENÍ VÝPOČTŮ VELIKOSTI ZÁLOH

Tato kapitola primárně vychází z práce pana docenta Burdy[2], ve které jsou rozebrány myšlenky stanovení velikostí záloh na základě odvozeného modelu výpočtů velikostí záloh při použití základních typů záloh.

Tyto myšlenky budu interpretovat a použiji je pro odvození vztahů pro další typy záloh a rozšířím model o možnost rotování záloh. V této kapitole budu postupovat co nejpodrobněji, aby byla patrná návaznost myšlenek a bylo zajištěno jasné interpretování podstaty věci.

V předchozí kapitole jsem u jednotlivých typů záloh uvedl základní matematické popisy. Z těchto popisů budu dále vycházet a v souladu s prací [2] provedu odvození výpočtů velikostí jednotlivých záloh a následně také celkových kapacit nutných pro pokrytí požadavků na zálohu daného datového prostoru. Také připomenu vztah 2.1, který matematicky definuje datový prostor.

3.1 Výpočty plné zálohy

Plná záloha je základním typem záloh, jak už bylo zmíněno v 2.1.1, pro plnou zálohu platí jednoduchý vztah 2.2. Velikost datového prostoru je možné popsat jako $|D_{t_i}| = |d_{t_i}| \cdot n$. Případné zvětšení záloh se tedy odvíjí pouze od zaplnění datového prostoru. Zaplnění datového prostoru má z dlouhodobého pohledu rostoucí charakter, který může být lineární nebo exponenciální, přičemž záleží na povaze dat ukládaných na daný prostor.[1]

Pro velikost plné zálohy tedy platí:

$$|F(t_i)| = |D| = |d| \cdot n. \tag{3.1}$$

3.2 Výpočty přírůstkové zálohy

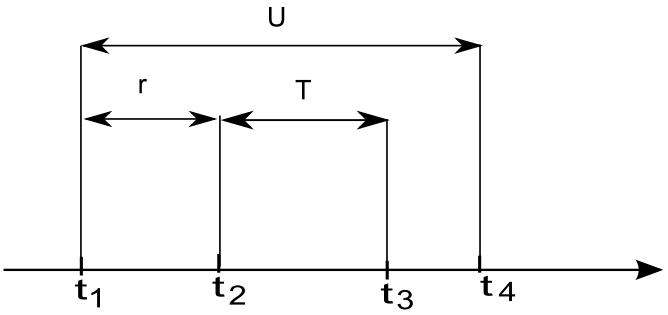
Při určování velikosti přírůstkové zálohy bude situace již složitější a pro možnost výpočtu budu muset použít určité pravděpodobnostní rozdělení, kterým bych mohl vypočítat dobu čekání na změnu dané datové jednotky.

Pro tyto účely je velice vhodné Exponenciální rozdělení bez posunutí, kdy hustota pravděpodobnosti představuje aktuální pravděpodobnost události změny datové jednotky a distribuční funkce představuje pravděpodobnost změny datové jednotky za daný uplynulý čas. V tomto mě zajímá, zda v určitém časovém úseku

[1] Zohlednění lineárního či exponenciálního růstu je mimo rozsah této práce. Dále budu předpokládat konstantní velikost dat.

opravdu došlo ke změně. Tuto situaci popisuje vztah distribuční funkce exponenciálního rozdělení 2.8.
Pro distribuční funkci popisující pravděpodobnost daného jevu při zahrnutí vztahu 2.6 platí: $P(t) = P(T \leq t) = 1 - e^{-\lambda \cdot t}$.

λ představuje míru intenzity změn jedné datové jednotky za daný časový interval. Obrázek 3.1 potom popisuje zálohu datové jednotky v čase, kdy se během t_1 a t_4 prováděla pravidelná přírůstková záloha.



Obr. 3.1: Pravděpodobnostní rozdělení „bez paměti“

Přitom mě zajímá pravděpodobnost, jestli k záloze datové jednotky dojde v čase t_2 nebo t_3. Z obrázku je zřejmé, že k záloze v čase t_2 dojde pouze v případě, že došlo ke změně v časovém intervalu r, a k záloze v čase t_3 dojde pouze v případě, že ke změně došlo v intervalu T.

Pravděpodobnost, že se v daném časovém intervalu datová jednotka změní, označím jako p. Pro ni potom platí přímo vztah 2.8, tedy:
$F(t) = P(T \leq r) = 1 - e^{-\lambda \cdot r}$. Pro další odvozování zavedu pravděpodobnost případu, kdy nedojde ke změně dat v daném časovém intervalu. Tuto pravděpodobnost budu označovat q.

Vzhledem k tomu, že pravděpodobnost nezálohování má přesně opačný logický význam, vypočítá se odečtením p od jedničky, tedy: $q = 1 - p$. Pokud dosadím za p, dostávám pak:

$$q = 1 - (1 - e^{-\lambda \cdot r}) = e^{-\lambda \cdot r}. \tag{3.2}$$

Pokud by bylo cílem vypočítat pravděpodobnost neprovedení zálohy v intervalu (t_1, t_2), vztah 3.2 byl postačující a stačilo by dosadit míru změny λ a časový interval r.

Cílem je ale zjištění provedení či neprovedení zálohy v intervalu T, tj. (t_2, t_3), což by se dalo vyjádřit podmíněnou pravděpodobností $P(U > r + T | U > r)$.

To je možné interpretovat jako pravděpodobnost, že nedošlo ke změně do času t_2 a nedojde ke změně ani v časovém intervalu T. Podmíněná pravděpodobnost má jednu unikátní vlastnost, kterou vyjadřuje vztah:

$$q = P(U > r + T|U > r) = P(U > T).[7] \tag{3.3}$$

Pokud nedošlo ke změně v intervalu r, pravděpodobnost, že nedojde ke změně na intervalu $r + T$, je stejná jako pravděpodobnost, že nedojde ke změně v intervalu T. Tato vlastnost se označuje jako exponenciální rozdělení „bez paměti“. Systém se totiž chová, jako by zapomněl, že nedošlo ke změně v intervalu r, a pravděpodobnost počítaná pouze v intervalu T odpovídá situaci, kdy by se 0 časové osy posunula do času t_2. Využiji vztah 3.3 a dosadím do pravé části ze vztahu 2.6: $q = P(U > r + T|U > r) = P(U > T) = 1 - P(U \leq T) = 1 - (1 - e^{-\lambda \cdot T})$. Pro pravděpodobnost nezměnění datové jednotky v časovém intervalu T platí tedy vztah:

$$q = e^{-\lambda \cdot T}. \tag{3.4}$$

Pravděpodobnost, že ke změně dojde, je analogicky:

$$p = 1 - e^{-\lambda \cdot T}. \tag{3.5}$$

Velikost přírůstkové zálohy se přímo odvíjí od počtu změněných datových jednotek. Pokud tedy použiji vztah 3.5, který udává změnu datové jednotky na intervalu T, kde $T = (t_i, t_j)$, velikost přírůstkové zálohy tedy je:

$$|Inc(t_i, t_j)| = |D| \cdot p = |D| \cdot (1 - e^{-\lambda \cdot T}). \tag{3.6}$$

3.3 Výpočty rozdílové zálohy

Velikost rozdílové zálohy je z matematického pohledu stejná jako jako přírůstková záloha. Jde pouze o speciální přírůstkové zálohy, kdy platí podmínka, že interval (t_i, t_j) představuje zároveň dobu od poslední plné zálohy, nehledě na to, kolik během tohoto intervalu proběhlo dalších (jiných, než plných) záloh.

Na základě výše uvedeného při splnění podmínky poslední plné zálohy v $t_i = t_{Full}$, T_{full} představuje právě toto období. V tomto případě platí vztah:

$$|Dif(t_{Full}, t_j)| = |D| \cdot p = |D| \cdot (1 - e^{-\lambda \cdot T_{Full}}). \tag{3.7}$$

3.4 Výpočty víceúrovňové přírůstkové zálohy

Víceúrovňová přírůstková záloha je pouze obdobou kombinace všech dosud zmíněných kombinací. Pro matematický popis je nutné rozdělit situaci do dvou základních možností:

- Záloha úrovně $l = 0$ spustí plnou zálohu, nehledě na to, jaké zálohy byly provedeny před tím. V tom případě platí vztah 3.8.
- Záloha úrovně $l \neq 0$ spustí buď přírůstkovou zálohu, pokud bezprostředně této záloze předcházela záloha s nižší úrovní. Nebo rozdílovou zálohu, jestliže mezi zálohou nižší úrovně a právě spuštěnou zálohou byly další zálohy s vyšší úrovní, než právě spuštěná záloha V tom případě platí vztah 3.9. Přitom T_{Lev} je doba od zálohy s nižší úrovní.

  Záloha úrovně 0 probíhá v čase t_j. Pak platí: Pro $l = 0$:

$$|Levl(0)(t_j)| = |F(t_j)| = |D|. \tag{3.8}$$

  Pro $l \neq 0$:

$$|Levl(l)(t_i, t_j)| = |D| \cdot p = |D| \cdot (1 - e^{-\lambda \cdot T_{Lev}}). \tag{3.9}$$

4 ODVOZENÍ PARAMETRŮ ZÁLOH

Na základě odvození vztahů pro výpočet velikosti základních typů záloh můžu vypracovat celý model zálohování a určit celkové nároky na úložný prostor/média a vzhledem k požadavkům tak zvážit výhodnost jednotlivých modelů v konkrétní implementaci v praxi. Informace budu v této kapitole čerpat zejména z práce pana docenta Burdy [2].

První, intuitivní parametr, je objem všech záloh C. Tuto informaci získám jednoduchým sečtením všech pořízených záloh, přičemž počet záloh budu značit M. Obecnou zálohu pořízenou v čase t_i budu značit $B(t_i)$, nehledě na to, o jaký typ zálohy se bude jednat. Objem všech záloh může být v praxi důležitý pro investice, což je pro implementaci nezbytné. Obecně pro celkový objem zazálohovaných dat platí:

$$C = \sum_{i=1}^{M} |B(t_i)|. \tag{4.1}$$

Druhým důležitým parametrem bude doba obnovy v případě nutnosti obnovy. Na obrázku 2.1.1 je zachycena situace obnovy z tří záloh, které jsou na sobě navázány. Pokud je tu požadavek na obnovu dat do $D(t_2)$, musí se využít $F(t_0)$, $Inc(t_1)$ a $Inc(t_2)$. Součet všech záloh je $M = 4$, počet nutných záloh pro obnovu do požadovaného času je však menší a je dán třemi logicky navázanými zálohy.

V tomto případě tedy pro množinu U_i platí: $U_3 = \{t_0, t_1, t_2\}$. Doba obnovy je zřejmě přímo úměrná množství dat, které se musí obnovit.[1] Proto bude výhodnější porovnávat množství dat (či pásek) nutných pro obnovení. Tento parametr budu značit jako R_i, pro data D_i, pak platí:

$$R_i = \sum_{k \in U_i} |B(t_k)|. \tag{4.2}$$

V tom případě pak platí pro obnovu dat do $D(t_2)$ velikost součtu záloh $R(3) = |B(t_0)| + |B(t_1)| + |B(t_2)|$. Pro střední hodnotu objemu záloh značených jako R pak platí:

$$R = \frac{1}{M} \cdot \sum_{i=1}^{M} R_i. \tag{4.3}$$

[1] V práci zanedbávám případné zpomalení způsobené hardwarovou implementací. Tyto analýzy jsou mimo rozsah této práce a úzce souvisejí s volbou hardwaru. Zejména páskové jednotky mohou vykazovat velké rozdíly při spojitém a nespojitém zápisu či čtení dat.

4.1 Plná záloha

Vzhledem k tomu, že každá plná záloha obsahuje zálohu všech datových jednotek z daného časového okamžiku, celkový objem dat je roven součtu všech pořízených plných záloh. V případě, že objem dat se nemění, může být D_i nahrazeno D. Pak lze napsat:

$$C = \sum_{i=1}^{M} |F(t_i)| = |D| \cdot M. \tag{4.4}$$

Pro objem obnovovacích záloh není třeba více záloh, protože každá záloha obsahuje všechny informace potřebné pro obnovu. Objem obnovovacích záloh je tedy roven velikosti dané zálohy, pak platí vztah:

$$R = |F(t_i)| = |D|. \tag{4.5}$$

4.2 Přírůstková záloha

Nejprve musím nadefinovat velikost jednotlivých záloh. Všechny předešlé přírůstkové zálohy jsou referenční pro každou další přírůstkovou. Pro každou platí přímo vztah 3.6, přitom první záloha nemá referenci. Potom jde o plnou zálohu, pak platí:

$$|B(t_i)| = \begin{cases} |D| & \text{, když } i = 1, \\ |D| \cdot (1 - e^{-\lambda \cdot T}) = |D| \cdot (1 - q) & \text{, v případě } i \ = 2, 3 \ldots M. \end{cases} \tag{4.6}$$

Pro celkový objem záloh je nutné sečíst jednotlivé pořízené zálohy, což provedu prostým sečtením a úpravou:

$$\begin{aligned} C &= \sum_{i=1}^{M} |B(t_i)| \\ C &= |D| + |D| \cdot (1 - q) + |D| \cdot (1 - q) + |D| \cdot (1 - q) + |D| \cdot (1 - q) \\ C &= |D| \cdot (1 + 1 - q + 1 - q + 1 - q + 1 - q) \\ C &= |D| \cdot (5 - 4 \cdot q) \\ C &= |D| \cdot (5 - (5 - 1) \cdot q). \end{aligned}$$

Obecně (pro počet záloh $M =$) pak lze psát:

$$C = |D| \cdot [M - (M - 1) \cdot q]. \tag{4.7}$$

Nyní odvodím vztah pro střední objem obnovovacích záloh pro přírůstkové zálohy. Budu vycházet z platností vztahů 4.2, 4.3 a 3.6. Nejdříve musím určit velikosti jednotlivých obnov v uvažovaném modelu pěti záloh. Pro obnoveni dat do specifického časového okamžiku je nutné mít k dispozici všechny předcházející zálohy, což zachycuje i obrázek 2.1.1.

$$\begin{aligned} R_1 &= |D| \\ R_2 &= |D| + |D| \cdot (1-q) \\ R_3 &= |D| + |D| \cdot (1-q) + |D| \cdot (1-q) = |D| + 2 \cdot |D| \cdot (1-q) \\ R_4 &= |D| + 3 \cdot |D| \cdot (1-q) \\ R_5 &= |D| + 4 \cdot |D| \cdot (1-q). \end{aligned}$$

Součet těchto hodnot je nutný při výpočtu středního objemu obnovovacích záloh. V případě, že $M = 5$, platí:

$$\begin{aligned} R_i &= R_1 + R2 + R_3 + R_4 + R_5 \\ R_i &= 5 \cdot |D| + 10 \cdot |D| \cdot (1-q) \\ R_i &= 5 \cdot |D| \cdot [1 + 2 \cdot (1-q)] = 5 \cdot |D| \cdot (1 + 2 - 2q) = 5 \cdot |D| \cdot (3 - 2q). \end{aligned}$$

Nyní mohu odvodit vztah pro střední objem obnovovacích záloh. V případě, že $M = 5$. Dosazením do vztahu 4.3 pak platí:

$$\begin{aligned} R &= \frac{1}{5} \cdot 5 \cdot |D| \cdot (3 - 2 \cdot q) \\ R &= \frac{1}{2} \cdot 2 \cdot |D| \cdot (3 - 2 \cdot q) \\ R &= \frac{1}{2} \cdot |D| \cdot (6 - 4 \cdot q). \end{aligned}$$

Obecně pak mohu psát:

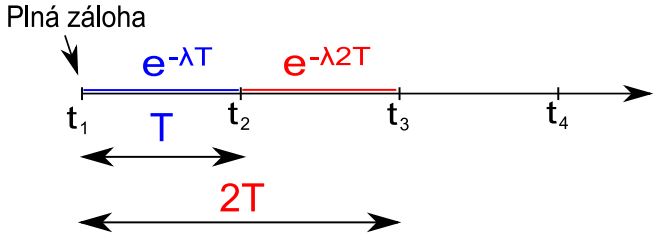
$$R = \frac{1}{2} \cdot |D| \cdot [(M+1) - (M-1) \cdot q]. \tag{4.8}$$

4.3 Rozdílová záloha

Jestliže vezmu v potaz, že rozdílová záloha akceptuje jako referenční zálohu pouze tu první (což je považováno za plnou zálohu), je nutné nejdříve nadefinovat velikost jednotlivých záloh.

Na obrázku 4.1 je naznačný způsob odvození vzorce 4.9. První záloha bude plná, protože neexistuje žádná plná záloha, ke které by mohly být zálohy vztaženy. Každý další interval představuje interval T. Celková pravděpodobnost neprovedení zálohy se s počtem těchto intervalů zmenšuje přímou úměrou. Jednoduše by se dalo za T doplňovat $(i-1)$ násobky T. Přitom $e^{-\lambda \cdot T} = q$, pak platí tento vztah:

$$|B(t_i)| = \begin{cases} |D| & , \text{když } i = 1, \\ |D| \cdot (1 - e^{-\lambda \cdot (i-1) \cdot T}) = |D| \cdot (1 - q^{(i-1)}) & , \text{v případě } i \ = 2, 3 \ldots M. \end{cases} \tag{4.9}$$



Obr. 4.1: Výpočet velikostí rozdílové zálohy

Takto definovanou velikost rozdílové zálohy již mohu použít pro výpočet celkového objemu záloh, popř. pro výpočet středního objemu obnovovacích záloh.

Pokud $|B(t_i)|$ z rozdílové zálohy dosadím do vzorce 4.1 [2], zde je uvedeno odvození vztahu:

$$\begin{aligned} C &= \sum_{i=1}^{M} |B(t_i)| \\ C &= |D| + |D| \cdot (1-q) + |D| \cdot (1-q^2) + |D| \cdot (1-q^3) + |D| \cdot (1-q^4) \\ C &= |D| \cdot (1 + 1 - q + 1 - q^2 + 1 - q^3 + 1 - q^4) \\ C &= |D| \cdot [5 - (q + q^2 + q^3 + q^4)]. \end{aligned}$$

Po úpravě pomocí geometrické posloupnosti pak mohu vztah napsat v této formě:

$$C = |D| \cdot [M - (q^4 + q^3 + q^2 + q)]. \tag{4.10}$$

[2] za předpokladu, že počet záloh $M = 5$.

Pro výpočet středního objemu obnovovacích záloh nejprve musím vypočítat objem obnovovacích záloh pro všechny případy a následně tuto hodnotu vydělit celkovým počtem záloh, čímž získám střední hodnotu:

$$
\begin{aligned}
R_1 &= |B(t_1)| = |D| \\
R_2 &= |B(t_2)| = |D| + |D| \cdot (1 - q) \\
R_3 &= |B(t_3)| = |D| + |D| \cdot (1 - q^2) \\
R_4 &= |B(t_4)| = |D| + |D| \cdot (1 - q^3) \\
R_5 &= |B(t_5)| = |D| + |D| \cdot (1 - q^4) \\
R &= \frac{1}{M} \cdot \sum_{i=1}^{M} R_i \\
R &= \frac{1}{M} \cdot |D| \cdot (1 + 1 + 1 - q + 1 + 1 - q^2 + 1 + 1 - q^3 + 1 + 1 - q^4) \\
R &= \frac{|D|}{M} \cdot (9 - q - q^2 - q^3 - q^4) \\
R &= \frac{|D|}{M} \cdot (10 - 1 - q - q^2 - q^3 - q^4).
\end{aligned}
$$

Vztah lze obecně zapsat v tomto tvaru:

$$R = \frac{|D|}{M} \cdot [2M - 1 - (q^4 + q^3 + q^2 + q)]. \tag{4.11}$$

5 PARAMETRY ROTAČNÍCH SCHÉMAT

V předchozí kapitole jsem odvodil parametry pro základní typy záloh. Parametry dále rozšířím tak, aby odpovídaly jednotlivým rotačním schématům (typům rotování médií). Tyto upravené parametry pak mohu přímo využít při porovnávání záloh při použití daného rotačního schématu. Celkový objem záloh a střední obnovovací objem ovšem nejsou jediné relevantní parametry pro porovnání jednotlivých rotačních schémat. V praxi je velmi důležitým parametrem také dostupnost daných záloh provedených v minulosti.

5.1 Dostupnost obnovy

Pro pochopení dostupnosti si musím položit důležitou otázku týkající se rotace médií. V části 2.1.3 jsem popsal jednotlivé rotace médií. Co je ale motivací k použití rotování? Média by se mohla vždy po záloze vyjmout a uložit do archivu. Takový postup je sice možný, ale neekonomický. Počet pásek uložených v archivu by roslt, bylo by možné obnovit data z jakéhokoliv období, ale režie na správu takového množství médií by byly neúnosné.

Jakmile jsou tyto pásky vyhodnoceny jako nedůležité nebo pravděpodobně nepotřebné pro obnovu, recyklují se a využívají se opět pro nové zálohy. Každé schéma rotace má jinou délku života média a porovnání někdy velice různých typů rotačních schémat vychází z praxe. Pro objektivní posouzení kvality dostupnosti obnovy musím zohlednit nejčastější typy požadavků na obnovu.

Nejčastějším požadavkem na obnovu jsou případy, kdy jde o omylem smazaná nebo poškozená data, která jsou v praxi používána každý den. V takových případech se dá předpokládat nutnost mít zálohu těchto dat za posledních čtyřiadvacet hodin.

Dalším, velice častým případem, je situace, kdy jsou data měněna méně často, např. několikrát do týdne. V takových případech se chyba projeví až po více jak čtyřiadvaceti hodinách a obvykle se vyžaduje jakákoliv verze zálohy dat, ovšem kromě té poslední (té, co byla provedena v posledních čtyřiadvaceti hodinách).

Méně častým, ale stále důležitým případem, jsou situace, kdy se soubory editují na týdenní bázi. Typickým případem mohou být týdenní reporty, zápisy ze schůzí, týdenní data připravená k odevzdání zákazníkovi. V takových případech se dá předpokládat konzistence souborů během víkendu. Jde o zálohy dat z jednoho, až tří týdnů zpět. Tuto úvahu omezím na období jednoho měsíce, který představuje dostatečnou rezervu pro běžné požadavky.

Pokud budu značit celkovou dostupnost obnovy jako A udávající průměr jednotlivých možných dostupností, přitom každá dostupnost bude mít hodnotu 1 (zá-

Obr. 5.1: Dostupnost obnovy

loha je dostupná) nebo 0 (záloha není k dispozici). Jednotlivé možnosti pak budou a_1 pro obnovu z posledních čtyřiadvacet hodin až a_5 pro obnovu z 3. týdne zpět, pak lze napsat:

$$A = \frac{\sum_{i=1}^{i=5} a_i}{5}. \tag{5.1}$$

5.2 Odvození parametrů pro rotační schémata

Nyní mám tři parametry, celkový objem, střední obnovovací objem a dostupnost obnovy. Tyto tři paramametry použiji pro porovnání jednotlivých rotačních schémat. Než ale budu moct přistoupit k porovnání, musím odvodit parametry pro jednotlivá rotační schémata. Vzhledem k tomu, že celkový objem a střední obnovovací objem je odlišný, pokud jde o první týden nebo o jakýkoliv další týden [1], budu vyhodnocovat parametry pro první a jakýkoliv následující týden.

5.2.1 Záloha GFS

Nejprve odvodím celkový objem a střední obnovovací objem pro případ, kdy jde o první týden a první záloha je tedy plná. Pro parametry označující první týden bude použit index W_1, pro týden jiný než první použiji W_2+, tedy např. C_{W_1} nebo C_{W_2+}.

[1] Přirůstkové a rozdílové zálohy pracují jako plné zálohy, pokud předtím k žádné záloze nedošlo.

$$
\begin{aligned}
C_{W_1} &= \sum_{i=1}^{M} |B(t_i)| \\
C_{W_1} &= |D| + |D| \cdot (1-q) + |D| \cdot (1-q) + |D| \cdot (1-q) + |D| \\
C_{W_1} &= |D| \cdot (1 + 1 - q + 1 - q + 1 - q + 1).
\end{aligned}
$$

$$
C_{W_1} = |D| \cdot (5 - 3 \cdot q). \tag{5.2}
$$

Pro střední obnovovací objem GFS rotace při prvním spuštění nejprve zjistím obnovovací objem pro jednotlivé obnovy:

$$
\begin{aligned}
R_1 &= |D| \\
R_2 &= |D| + |D| \cdot (1-q) \\
R_3 &= |D| + 2 \cdot |D| \cdot (1-q) \\
R_4 &= |D| + 3 \cdot |D| \cdot (1-q) \\
R_5 &= |D|.
\end{aligned}
$$

Střední obnovovací objem GFS rotace pro první týden platí:

$$
R_{W_1} = \frac{1}{M} \cdot \sum_{i=1}^{M} R_i.
$$

$$
R_{W_1} = \frac{1}{5} \cdot |D| \cdot (11 - 6 \cdot q). \tag{5.3}
$$

Pro další týdny musím upravit vztahy tak, že pondělní záloha bude provedena přírůstkovou metodou. Pondělní záloha nebude plná, ale bude se jednat o přírůstkovou zálohu od posledního pátku. Tedy, časový interval bude trojnásobný oproti běžnému dennímu intervalu, $(1 - q^3)$:

$$
\begin{aligned}
C_{W_2} &= \sum_{i=1}^{M} |B(t_i)| \\
C_{W_2} &= |D| \cdot (1 - q^3) + |D| \cdot (1-q) + |D| \cdot (1-q) + |D| \cdot (1-q) + |D| \\
C_{W_2} &= |D| \cdot (1 - q^3 + 1 - q + 1 - q + 1 - q + 1).
\end{aligned}
$$

$$
C_{W_{2+}} = |D| \cdot (5 - q^3 - 3 \cdot q). \tag{5.4}
$$

Pro střední obnovovací objem GFS rotace dalších týdnů je analogické se středním objemem pro první týden. Jediný rozdíl je ve výpočtu přírůstkové zálohy na první den dalšího týdnu, který odpovídá opět $|D| \cdot (1 - q^3)$. Potom platí:

$$R_{W_2} = \frac{1}{M} \cdot \sum_{i=1}^{M} R_i.$$

$$R_{W_2} = \frac{1}{5} \cdot |D| \cdot (15 - 4 \cdot q^3 - 6 \cdot q). \tag{5.5}$$

Pokud se podívám na časové rozložení dostupnosti 5.1 a porovnám ho s rotací GFS 2.7, za všechny typy obnov vyjde $a_i = 1$.

Záloha z posledního dne, posledních čtyřiadvaceti hodin, obnova v tomto týdnu i obnovy z předchozích 3 týdnů jsou k dispozici. Po dosazení do vztahu 5.1 platí pro dostupnost obnovy $A_{GFS} = 1$.

5.2.2 Pět pásek

Pokud porovnám rotaci pěti pásek a GFS, je patrné, že celkový objem i střední obnovovací objem je shodný. To platí jak pro první týden, tak pro případný další týden zálohy. Dále tedy budu předpokládat, že vztahy 5.2,5.3,5.4 a 5.5 platí pro rotaci pěti pásek.

Pro dostupnost obnov je ovšem situace jiná. Metoda pěti pásek předpokládá dostupnost pouze jednoho týdne. Nehledě na to, jestli byla provedena záloha pro poslední týden, celkový součet bude $\sum a_i{}_{i=1}^{i=5} = 3$, dostupnost zálohy pak $A_{5T} = 0.6$.

5.2.3 Hanojské věže - plná záloha

Hanojské věže s využitím plných záloh pracují na shodném principu, jako by šlo o běžnou sekvenci plných záloh, přičemž přidanou hodnotu tomuto systému vytváří důmyslný systém rotace pásek. Ten způsobuje dostatečnou retenci zálohy pro případnou obnovu.

Po využití všech zálohovacích pásek [2] je kapacita konstantní a odpovídá počtu záloh.

Pro celkový objem lze tedy použít vztah 4.5, pro střední obnovovací objem lze použít vztah pro objem obnovovacích záloh plné zálohy 4.5. Přitom je vzhledem k plným zálohám irelevantní, zda se jedná o první či další týden záloh.

I přes použití plných záloh má schéma Hanojských věží přednost v dostupnosti obnovovacích médií. Pokud porovnám princip rotace s dostupností, při obnovování

[2] Předpokládám použití pěti pásek.

31. dne v měsíci je k dispozici záloha z posledních čtyřiadvacet hodin (páska č. 2) i další záloha z posledního týdne (páska č. 3).

K dispozici je i páska z předešlého týdne (páska č. 5). Z období dva týdny zpět žádná záloha není, ale existuje páska ze tří týdnů zpět (páska č. 4).

V každém případě je vždy dostupná páska ze dvou nebo tří týdnů zpět (střídavě). Čtyři zálohy z pěti tedy odpovídají $A_H = 0.8$.

5.2.4 Hanojské věže - víceúrovňová záloha

Tato varianta je mnohem komplexnější než varianta plných záloh. K co největší podobnosti a porovnatelnosti s ostatními zálohami bylo nutné zvolit variantu, která není stejná v týdenním cyklu. Využil jsem model víceúrovňové zálohy zmíněný v knize Prestona, W. [13], který jsem upravil tak, aby odpovídal počtu pěti pásek. Schéma střídání pásek je zachyceno na obrázku 2.9.

Z obrázku je patrné, že se start cyklu posunuje po jenom dnu, životnost pásky s úrovní 0 jsou tři týdny. Páska s úrovní 0 se opět použije při záloze v pátek po šestnácti týdnech, což je celkový cyklus, z kterého mohu přesně vypočítat střední obnovovací objem.

Každý týden ale vzniká jiný celkový objem dat, který by představoval komplexní výpočet pro každý týden. V tomto případě nebude účelné počítat přesný objem záloh každého týdne a proto vypočítám celkový objem záloh za šestnáct týdnů a vypočítám z něj průměrnou hodnotu objemu záloh na jeden týden.

Po odvození všech osmdesáti vztahů pro všechny zálohy v šestnácti týdnech, jejich sečtení a následném vydělení šestnácti týdny celková kapacita na jeden týden odpovídá vztahu:

$$C_{TOH_{LevW}} = \frac{1}{16} \cdot |D| \cdot (80 - 3q^{12} - 2q^{10} - 8q^6 - 10q^4 - 8q^3 - 12q^2 - 32q). \quad (5.6)$$

K výpočtu středního obnovovacího objemu Hanojských věží jsem podle obrázku 2.9 sestavil jednotlivé součty pásek nutných pro obnovení dat k určitému dni. Těchto osmdesát vztahů jsem následně sečetl a vydělil počtem osmdesáti záloh. Střední obnovovací objem tedy odpovídá:

$$R_{TOH_{Lev}} = \frac{1}{10} \cdot |D| \cdot (30 - 3q^{12} - 2q^{10} - 4q^6 - 3q^4 - q^3 - 3q^2 - 4q). \quad (5.7)$$

Pro dostupnost pásek jsem vybral dvě limitní varianty. Nejlepší situaci, tedy obnovu v poslední den před spuštěním zálohy s úrovní 0. V takovém případě není možné obnovit data jen z druhého týdne nazpět. Dostupnost záloh lze považovat za totožnou při použití Hanojských věží s plnými zálohami, tedy $A_H = 0.8$. Pokud by

však došlo k záloze s úrovní 0, všechny dříve použité zálohy by byly nepoužitelné a pak by byla dostupnost obnovy pouze $A_H = 0.1$. Z tohoto důvodu se běžně používá schéma s větší retencí úrovně 0 a jedna. Další možností je využití dvou pásek s úrovní 0.

Předmětem této práce není nalezení nejvhodnější varianty Hanojských věží, ale porovnání dané metody s ostatními strategiemi záloh, a proto vyjdu z průměru těchto hodnot. Dostupnost záloh bude tedy $A_H = 0.45$.

5.2.5 Amanda - přírůstková metoda

Oproti ostatním typům záloh Amanda pracuje na jiné filozofii. Jednotlivé zálohy se dělí na části, které se provádí jiným způsobem záloh. Největší prioritou je rovnoměrné zálohování bez větších rozdílů mezi zálohami, což významným způsobem zjednodušuje odhad velikosti zálohovacího období. Už nedochází ke krátkým zálohám přes týden a neúměrně dlouhým zálohám přes víkend, zálohy jsou tedy srovnatelné a dochází k menšímu ovlivnění jiných služeb.

Amanda umožňuje variabilně nadefinovat úrovně jednotlivých záloh a zvolit si tak nejoptimálnější zálohovací algoritmus. Cílem této práce není určení nejvhodnějšího algoritmu Amandy, proto zvolím krajní možnosti, přičemž parametry ostatních možností se budou nacházet v intervalu zvolených variant.

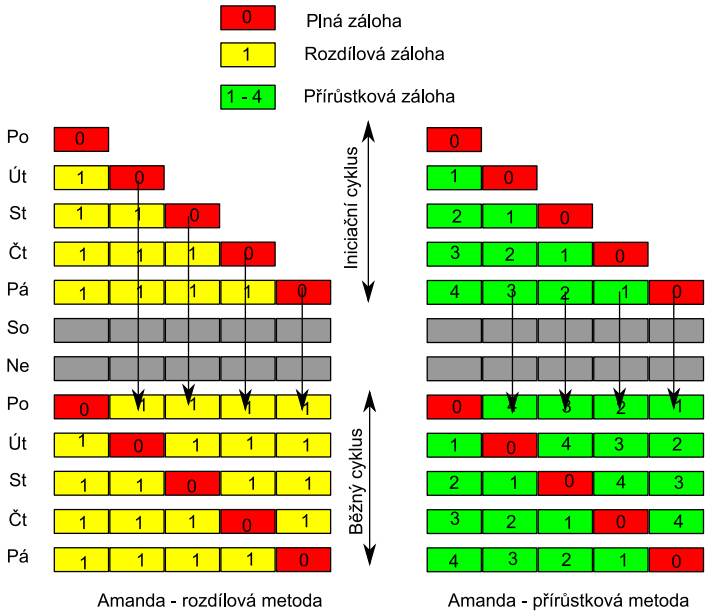
Obrázek 5.2 porovnává dvě zvolená schémata Amandy. Pro obě budou odvozeny potřebné parametry.

Pro přírůstkovou metodu nejprve určím objem záloh. Pro jednotlivé dny v prvním týdnu platí:

$$\begin{aligned} C_1 &= \frac{1}{5}|D| \\ C_2 &= \frac{1}{5}|D| + \frac{1}{5}|D| \cdot (1-q) \\ C_3 &= \frac{1}{5}|D| + \frac{2}{5}|D| \cdot (1-q) \\ C_4 &= \frac{1}{5}|D| + \frac{3}{5}|D| \cdot (1-q) \\ C_5 &= \frac{1}{5}|D| + \frac{4}{5}|D| \cdot (1-q). \end{aligned}$$

Po sečtení všech záloh a úpravách je celkový týdenní objem zálohy přírůstkové Amandy v prvním týdnu:

$$C_{AIncW1} = |D| \cdot (3 - 2 \cdot q). \tag{5.8}$$



Obr. 5.2: Zvolené varianty Amandy

Pro obnovení je nutné použít všechny zálohy, proto je střední obnovovací objem v prvním týdnu roven celkovému týdennímu objemu záloh [3], tedy:

$$R_{AInc_{W1}} = |D| \cdot (3 - 2 \cdot q). \tag{5.9}$$

Pro přírůstkovou metodu Amandy v dalších týdnech je objem záloh pro každý den definován stejným vztahem. Vždy se jedná o plnou zálohu poměrné části a zbývající část představuje přírůstková metoda. Opět zde musím započítat delší prodlevu způsobenou víkendem:

$$\begin{aligned}
C_1 &= \frac{1}{5}|D| + \frac{4}{5}|D| \cdot (1 - q^3) \\
C_2 &= \frac{1}{5}|D| + \frac{4}{5}|D| \cdot (1 - q) \\
C_3 &= \frac{1}{5}|D| + \frac{4}{5}|D| \cdot (1 - q) \\
C_4 &= \frac{1}{5}|D| + \frac{4}{5}|D| \cdot (1 - q) \\
C_5 &= \frac{1}{5}|D| + \frac{4}{5}|D| \cdot (1 - q).
\end{aligned}$$

[3] U Amandy je ale obnova možná až v případě, že je zazálohovaný celý první týden.

Po sečtení všech záloh a úpravách je celkový týdenní objem zálohy přírůstkové Amandy v dalších týdnech:

$$C_{AInc_{W2}} = |D| \cdot \left(\frac{25 - 4q^3 - 16q}{5} \right). \tag{5.10}$$

Pro střední obnovovací objem je vzhledem k závislosti na předešlém týdnu nutné zohlednit vždy týden před samotnou obnovou, proto bude konkrétně pro tento algoritmus uveden střední obnovovací objem pro druhý a třetí týden. Od třetího týdne už je výpočet stejný pro jakýkoliv další týden. [4]

Postup je analogický. Sečtou se tedy všechny zálohy pro daný týden a vypočítá se aritmetický průměr. Po sečtení a úpravách je vztah pro střední obnovovací objem definován takto:

$$R_{AInc_{W2}} = |D| \cdot \left(\frac{118 - 20q^3 - 73q}{25} \right). \tag{5.11}$$

Pro třetí týden je střední obnovovací objem již shodný s celkovým objemem zálohy za týden, protože pro obnovení je vždy nutné sečíst pět po sobě jdoucích záloh. Pro další týdny tedy platí:

$$R_{AInc_{W3+}} = |D| \cdot \left(\frac{25 - 4q^3 - 16q}{5} \right). \tag{5.12}$$

Pro výpočet dostupnosti musí být bráno v úvahu relativně krátké období retence. Jde o velmi podobnou situaci jako u rotace pěti pásek, zálohy se po týdnu recyklují, $A_A = 0.6$.

5.2.6 Amanda - rozdílová metoda

Rozdílová verze Amandy, využívá pro zálohy pouze úroveň nula a jedna, takto docílí rozdílového zálohování. Použiji tedy vztahy pro výpočet rozdílových záloh 4.9. Nejprve opět odvodím objem záloh v prvním týdnu:

$$\begin{aligned}
C_1 &= \frac{1}{5}|D| \\
C_2 &= \frac{1}{5}|D| + \frac{1}{5}|D| \cdot (1 - q) \\
C_3 &= \frac{1}{5}|D| + \frac{1}{5}|D| \cdot (1 - q + 1 - q^2) \\
C_4 &= \frac{1}{5}|D| + \frac{1}{5}|D| \cdot (1 - q + 1 - q^2 + 1 - q^3)
\end{aligned}$$

[4] I v aplikace BackupCalc se bude konkrétně u této rotace brát jako relevantní střední obnovovací objem pro druhý a třetí týden.

$$C_5 = \frac{1}{5}|D| + \frac{1}{5}|D| \cdot (1 - q + 1 - q^2 + 1 - q^3 + 1 - q^4).$$

Po sečtení všech záloh a úpravách je celkový objem zálohy rozdílové Amandy v prvním týdnu:

$$C_{ADif_{W1}} = |D| \cdot \left(\frac{15 - q^4 - q^3 - q^2 - q}{5} \right). \tag{5.13}$$

Střední obnovovací objem pro první týden lze určit. Je ale důležité podotknout, že nemá příliš velký význam. Schopnost plné obnovy je až po prvním cyklu záloh, proto by v podstatě neměl být brán v potaz. Pro výpočet jednotlivých obnov platí:

$$\begin{aligned}
R_1 &= \frac{1}{5}|D| \\
R_2 &= \frac{2}{5}|D| + \frac{1}{5}|D| \cdot (1 - q) \\
R_3 &= \frac{3}{5}|D| + \frac{1}{5}|D| \cdot (1 - q^2) + \frac{1}{5}|D| \cdot (1 - q) = \frac{3}{5}|D| + \frac{1}{5} \cdot (2 - q - q^2) \\
R_4 &= \frac{4}{5}|D| + \frac{1}{5}|D| \cdot (3 - q - q^2 - q^3) \\
R_5 &= \frac{5}{5}|D| + \frac{1}{5}|D| \cdot (4 - q - q^2 - q^3 - q^4).
\end{aligned}$$

Při použití vztahu 4.3 a úpravách je vztah pro střední obnovovací objem systému Amanda:

$$R_{ADif_{W1}} = |D| \cdot \left(\frac{8 - q^3 - q^2 - q}{5} \right). \tag{5.14}$$

Pro další týdny je nutné počítat s modelem, kdy se zálohy provádí pouze všední dny. Proto se do výpočtů musí zahrnout víkendové okno způsobující zvýšení pravděpodobnosti na změnu. Každý den prodlevy způsobí u rozdílové zálohy umocnění koeficientu q. Pro jednotlivé objemy záloh. V dalším týdnu tedy platí:

$$\begin{aligned}
C_1 &= \frac{1}{5}|D| + \frac{1}{5}|D| \cdot (1 - q^6) + \frac{1}{5}|D| \cdot (1 - q^5) + \frac{1}{5}|D| \cdot (1 - q^4) + \frac{1}{5}|D| \cdot (1 - q^3) \\
C_2 &= \frac{1}{5}|D| + \frac{1}{5}|D| \cdot (1 - q^6) + \frac{1}{5}|D| \cdot (1 - q^5) + \frac{1}{5}|D| \cdot (1 - q^4) + \frac{1}{5}|D| \cdot (1 - q) \\
C_3 &= \frac{1}{5}|D| + \frac{1}{5}|D| \cdot (1 - q^6) + \frac{1}{5}|D| \cdot (1 - q^5) + \frac{1}{5}|D| \cdot (1 - q^2) + \frac{1}{5}|D| \cdot (1 - q) \\
C_4 &= \frac{1}{5}|D| + \frac{1}{5}|D| \cdot (1 - q^6) + \frac{1}{5}|D| \cdot (1 - q^3) + \frac{1}{5}|D| \cdot (1 - q^2) + \frac{1}{5}|D| \cdot (1 - q) \\
C_5 &= \frac{1}{5}|D| + \frac{1}{5}|D| \cdot (1 - q^4) + \frac{1}{5}|D| \cdot (1 - q^3) + \frac{1}{5}|D| \cdot (1 - q^2) + \frac{1}{5}|D| \cdot (1 - q).
\end{aligned}$$

Po sečtení a úpravě je celkový objem pro další týdny roven:

$$C_{ADif_{W2}} = |D| \cdot \left(\frac{25 - 4q^6 - 3q^5 - 3q^4 - 3q^3 - 3q^2 - 4q}{5} \right). \tag{5.15}$$

Výpočet středního obnovovacího objemu pro druhý týden je podobně jako u přírůstkové varianty Amandy vztažený k výpočtům objemů záloh z prvního týdne. Po sečtení a vydělení pěti dny platí:

$$R_{ADif_{W2}} = |D| \cdot \left(\frac{115 - 14q^6 - 12q^5 - 14q^4 - 15q^3 - 15q^2 - 20q}{25} \right). \tag{5.16}$$

V dalších týdnech již obnova vždy představuje využití všech stejně velikých záloh v počtu jednoho cyklu zpět. V tom případě tedy platí, že $C_{A_{W2}} = R_{ADif_{W3+}}$. Pro výpočet středního obnovovacího objemu platí tedy stejný vztah:

$$R_{ADif_{W3+}} = |D| \cdot \left(\frac{25 - 4q^6 - 3q^5 - 3q^4 - 3q^3 - 3q^2 - 4q}{5} \right). \tag{5.17}$$

Výpočet dostupnosti pro rozdílovou variantu Amandy je stejný jako dostupnost pro přírůstkovou variantu, přičemž obnova je možná jen týden zpět, $A_A = 0.6$.

V této kapitole jsem si připravil matematické vztahy pro popsaná rotační schémata. Tyto vztahy byly implementovány v aplikaci pro výpočet parametrů jednotlivých záloh. Užití aplikace bude popsáno v další kapitole.

Na základě výpočtů v této aplikaci bude snadnější porovnat jednotlivá rotační schémata a zvolit pro to nejefektivnější a nejvhodnější.

6 POROVNÁNÍ ROTAČNÍCH SCHÉMAT

V této kapitole využiji aplikaci BackupCalc pro výpočty a následné porovnání zvolených kritérií. Výsledky budou reprezentovány v grafickém zpracování.

6.1 Výpočty celkových objemů záloh

Celkový objem záloh je zpracován jak pro první, tak další týdny záloh. Při porovnání jednotlivých objemů záloh v prvním týdnu je důležité zmínit, že rotační schéma Amandy docílí plného zazálohování datového úložiště až na úplném konci týdne. Jiná schémata mají již od první zálohy k dispozici alespoň jednu zálohu pro případ nutné obnovy, avšak Amanda v iniciačním cyklu obnovu všech dat neposkytuje.

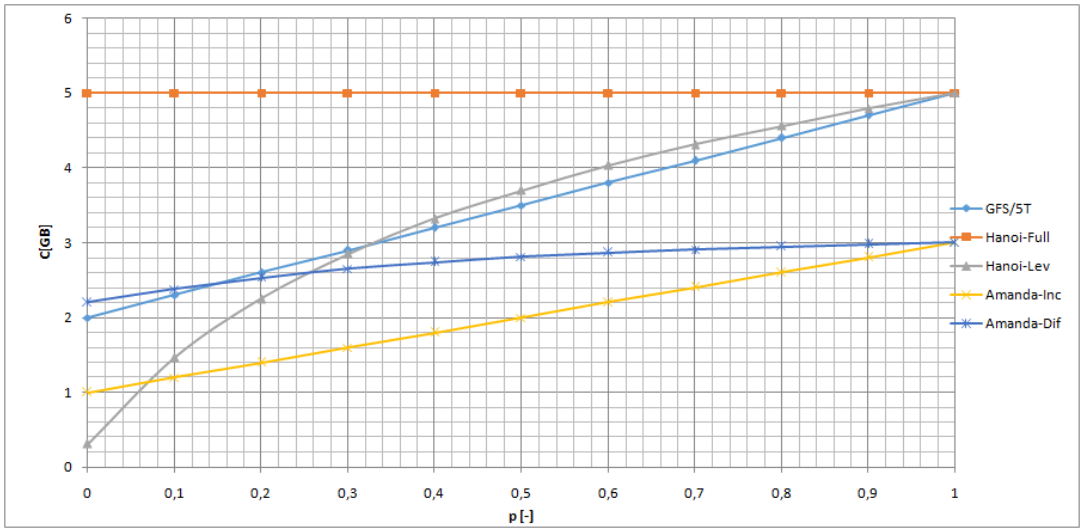
Ve všech následujících úvahách budu vycházet z toho, že míra změn na úložišti odpovídá rozmezí 0 - 0.25 [1]. Tato míra změn odpovídá i dokumentu [10] (graf. „CDFs of files by age“).

Při porovnání jednotlivých schémat 6.1 je patrný trend, kdy zvětšující se míra změn zvětšuje také objem záloh. Většina typů záloh se při zvětšování míry změn blíží k pětinásobku celkového objemu úložného prostoru, pouze Amanda se blíží k trojnásobku. Důvodem takového rozdílu je zřejmě postupné zálohování prostoru, které netvoří při první záloze plnou zálohu daného úložiště.

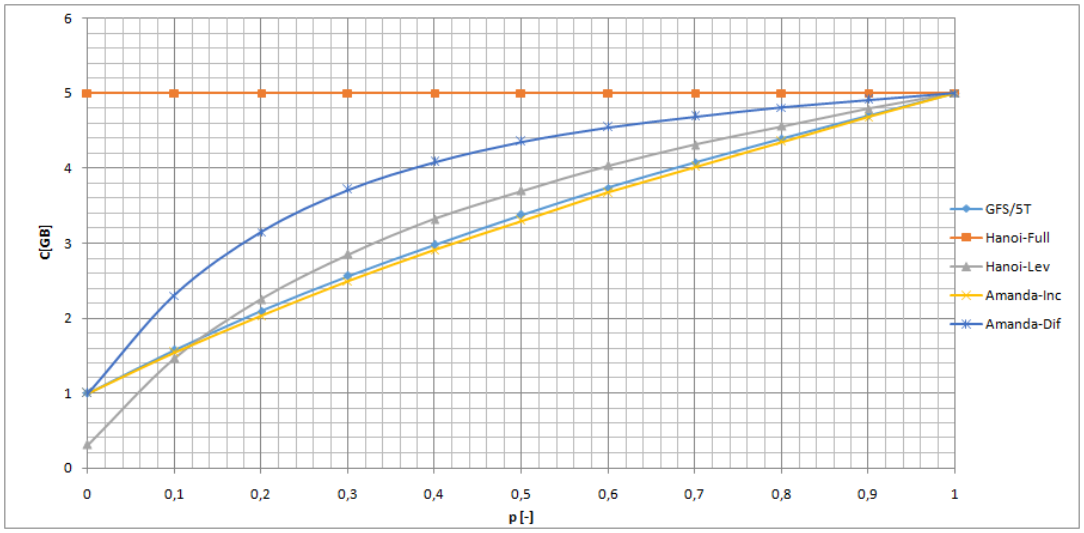
Hanojské věže vykazují velmi nízké hodnoty celkového objemu pásek. Je však nutné podotknout, že jde o průměrnou hodnotu objemu na období šestnácti týdnů. Důvodem je komplexnost a variabilita této zálohy. Objem záloh za první týden není tak podstatný jako objem záloh pro další týdny.

Podstatnější je ovšem objem záloh pro druhý týden a další týdny, přičemž porovnání je vykresleno na obrázky 6.2. Podobně jako u prvního týdne se jeví jako nejméně výhodné schéma Hanojských věží s plnou zálohou, ostatní vykazují přibližně srovnatelné hodnoty, ať už jde o jakoukoli míru změn. Velmi slušných výsledků dosahuje schéma Hanojských věží při použití víceúrovňové zálohy, zatímco jako méně výhodná se jeví Amanda při použití rozdílových záloh.

[1] Tento předpoklad vychází z praktických zkušeností ve firemním produkčním prostředí.



Obr. 6.1: Celkový objem záloh v prvním týdnu

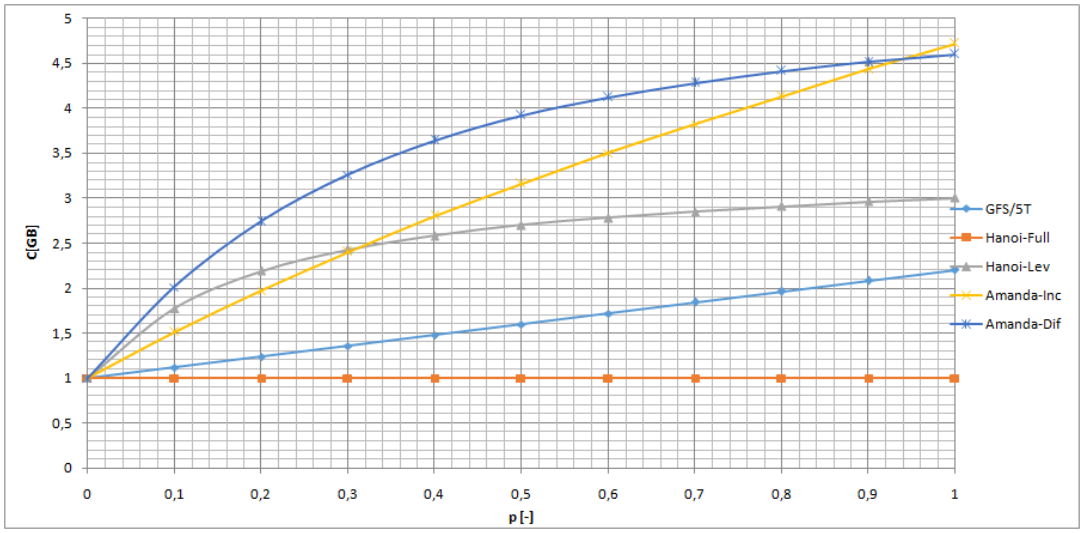


Obr. 6.2: Celkový objem záloh v dalších týdnech

6.2 Výpočty středního objemu obnovy

I když je celkový objem důležitý parametr, výpočet středního objemu obnovy je mnohem zásadnější, zejména ve spojitosti s dodržením SLA dostupnosti dat. Dalším důvodem je stále klesající cena úložného a zálohovacího prostoru.

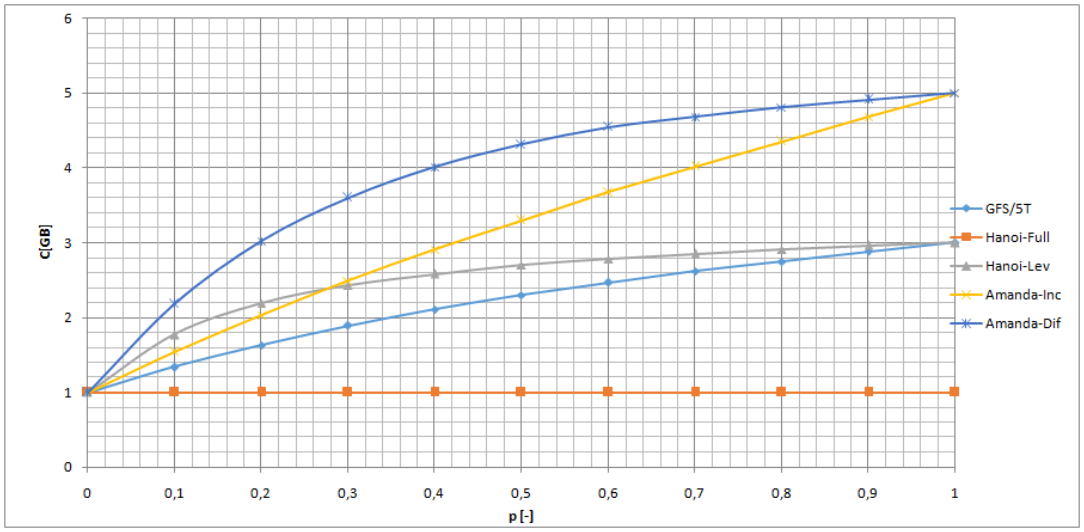
Obecně však neplatí, že velikost středního objemu obnovy odpovídá době obnovy. Přesto je tento parametr užitečným vodítkem pro určení přibližné doby obnovy. Musí se však zohlednit také technologie používaná pro zálohování.



Obr. 6.3: Střední objem obnovy první týden

Obrázek 6.3 popisuje střední objem obnovy pro první týden. Oproti objemu záloh je naopak nejvýhodnější právě schéma Hanojských věží s plnou zálohou. Tento výsledek je logický, protože při obnově postačuje jedna záloha pro obnovení všech dat, zatímco jiné metody vyžadují kombinaci více záloh, tedy starší plné a novější přírůstkové (přírůstkových) nebo rozdílové. Jako další nejvýhodnější metoda se jeví schéma GFS, popř. 5T. Ty vykazují v rozmezí míry změn od 0 do 0.25 jen nepatrné navýšení středního obnovovacího objemu. V této závislosti je nejméně výhodné použití Amandy s rozdílovými zálohami a Hanojských věží s víceúrovňovou zálohou.

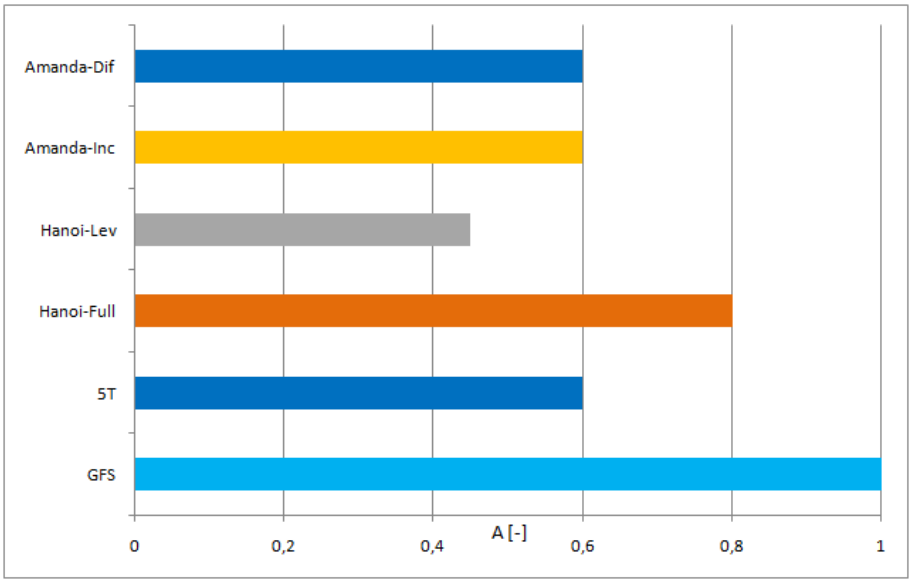
Graf na obrázku 6.4 porovnává střední obnovovací objem pro další týdny. Situace je podobná s prvním týdnem. Podstatnější změnou je navýšení středního obnovovacího objemu u GFS a 5T. Důvodem je prodleva snížení počtu plných záloh (v prvním týdnu cyklus začíná a končí plnou zálohou), dále se zvětšuje záloha o prodlevu v zálohách přes víkend. To zvětšuje pondělní přírůstkovou zálohu, navíc je při obnově nutné využít plnou zálohu z minulého týdne.



Obr. 6.4: Střední objem obnovy další týdny

6.3 Výpočty dostupnosti

V případě porovnávaných záloh jsem počítal s konstantním počtem zálohovacích médií. V praxi se média nakupují ve velkých sériích s dostatečnou rezervou, avšak pro porovnání jsem však zvolil situaci, kdy je k dispozici omezený a stejný počet médií.



Obr. 6.5: Dostupnost záloh

Všechna rotační schémata mají k dispozici pět zálohovacích médií. Dostupnosti jsou porovnány v grafu na obrázku 6.5. Největší dostupnost mají zálohy pomocí

GFS a další vhodné řešení představuje rotační schéma Hanojských věží. Nejhorší variantou je rotační schéma Hanojských věží při použití víceúrovňové zálohy.

7 ZÁVĚR

7.1 Cíle práce

Tato bakalářská práce měla za cíl shrnout problematiku zálohování z teoretického i praktického pohledu. Práce popsala matematické modely běžně používaných strategií záloh a byly zvoleny parametry záloh pro porovnání.

Práce dále podrobně vysvětlila rotační schémata a uvedla nejčastěji realizované principy zálohování. V práci jsou také podrobně popsána fyzicky používaná média, včetně jejich výhod a nevýhod.

Na základě zvolených parametrů jsem vytvořil aplikaci, která mi pomohla zjednodušit výpočty a porovnání jednotlivých rotačních schémat, čímž jsem také mohl zvolit nejvhodnější strategii pro zálohování.

7.2 Výběr parametrů

Pro porovnání schémat jsem zvolil tři parametry nejlépe vystihující požadavky pro zálohování. Prvním parametrem byl celkový objem záloh, který určuje kapacitní nároky zálohování. Druhým parametrem byl střední obnovovací objem záloh odpovídající střední době obnovy. Třetím parametrem byla dostupnost zálohy při zohlednění běžných retencí pro dané rotační schéma.

Vzhledem k tomu, že zálohy v dnešní době obsahují kombinace základních typů záloh (nepoužívají se pouze plné, pouze rozdílové nebo přírůstkové zálohy), byla pro porovnání použita až celková rotační schémata. Toto porovnání lépe odpovídá reálně nasazeným strategiím záloh a výsledky z této práce se dají lépe uplatnit v praktických situacích.

I přes to, že jsem uváděl a odvozoval vztahy pro parametry v prvním týdnu, z praktického hlediska je mnohem důležitější porovnání parametrů pro další týdny, protože jejich vliv je zásadnější. Vyhodnocení nejvhodnější zálohy se tedy bude odvíjet pouze od dalších týdnů.

Zvolené parametry jsou v některých případech protichůdné a pro výběr nejvhodnější strategie zálohování je velmi důležité vyjasnit si požadavky na zálohování. Obecně je největší důraz kladen na rychlost obnovy. Při omezeném rozpočtu na zálohovací média může ale hrát větší roli požadavek na co nejmenší objem záloh. Pokud jde o prostředí, kde dochází k velmi časté obnově, pak je nejdůležitějším parametrem dostupnost záloh.

Tabulka 7.1 srovnává jednotlivá schémata na základě zvolených parametrů a míry změn na daném úložišti. Pro zvolení nejoptimálnější varianty by mělo dojít nejdříve k analýze datového prostoru a zjištění míry změn. Podle míry změn a podle nejdůležitějšího kritéria lze vybrat nejvhodnější rotační schéma. Jednotlivé varianty jsou uvedeny v každé buňce tabulky a jsou označeny pořadovým číslem od nejvhodnějšího k nejméně vhodnému, přičemž v tabulce jsou vždy uvedené jen tři nejvhodnější varianty.

Míra změn	Požadavky na rychlou obnovu	Požadavky na minimálni objem záloh	Požadavky na dostupnost
0 - 0.1	1.Hanoi-Full 2.GFS/5T 3.Amanda-Inc	1.Hanoi-Lev 2.Amanda-Inc 3.GFS nebo 5T	1.GFS 2.Hanoi-Full 3.5T nebo Amanda-Inc nebo Amanda-Dif
0.1 - 0.25	1.Hanoi-Full 2.GFS nebo 5T 3.Amanda-Inc	1.Hanoi-Lev 2.Amanda-Inc 3.GFS nebo 5T	1.GFS 2.Hanoi-Full 3.5T nebo Amanda-Inc nebo Amanda-Dif
0.25 - 1	1.Hanoi-Full 2.GFS nebo 5T 3.Hanoi-Lev	1.Amanda-Inc 2.GFS nebo 5T 3.Hanoi-Lev	1.GFS 2.Hanoi-Full 3.5T nebo Amanda-Inc nebo Amanda-Dif

Tab. 7.1: Porovnání rotačních schémat

Je důležité podotknout, že efektivnost některých rotačních schémat je velmi ovlivněna dodatečným nastavením, zejména v jakých dnech dochází jsou zálohy prováděny. U Amandy je možné vhodně zvolit kombinace přírůstkových a rozdílových záloh, zatímco u Hanojských věží je možné zvolit víc pásek pro úroveň 0, apod. Při výběru těchto parametrů jsem kladl důraz na co nejpřehlednější porovnání všech rotačních schémat.

7.3 Shrnutí výsledků

Výsledky v tabulce 7.1 se dají částečně zobecnit. Tabulka uvádí nejvhodnější varianty zálohování, ale není možné jednoznačně určit, které rotační schéma je to nejvhodnější. Toto zobecnění mi umožní určit nejefektivnější strategii zálohování.

Míra změn na běžném úložišti se pohybuje do $p = 0.25$, proto zúžím výběr na tento rozsah. Nejdůležitějšími požadavky při výběru vhodného rotačního schématu jsou rychlost obnovy a dostupnost záloh, přičemž požadavky na co nejmenší objem záloh je již méně časté. Menší objem záloh by mohl být důležitý, pokud by se prováděly zálohy do cloudového řešení a poplatky by se tak odvíjely od velikosti záloh. Nebo by síťové propojení neposkytovalo dostatečnou rychlost uploadu pro uložení zálohy na cloudové úložiště. Cloudová řešení zatím nejsou primární variantou, proto budou požadavky na minimální objem záloh méně důležité.

V případě, že jsou vhodně zvoleny podmínky pro obnovu, nemusel by být kladen tak velký důraz na rychlost obnovy. Je tedy velmi podstatné vyjednání správně SLA pro službu zálohování. Nejvhodnějším parametrem tedy budou požadavky na dostupnost.

Na základě výše uvedeného je tedy zřejmé, jaká strategie pro zálohování je obecně nejvhodnější. Nejdůležitější je vhodné vyjednání požadavků a stanovení vhodných SLA. Jednalo by se o on-premise řešení, kdy není kladen důraz na minimální velikost záloh, přitom by byla nejdůležitější možnost obnovy po relativně dlouhé časové období. Nejefektivnější rotační schéma pro tuto strategii zálohování by tedy bylo GFS. Pro přímé srovnání dat v konkrétním případě lze využít aplikaci, která byla vytvořena jako součást mé bakalářské práce, aplikace BackupCalc.

7.4 Další rozvoj práce

Případný budoucí vývoj této práce by se mohl zaměřit na navrhnutí nového, efektivnějšího rotačního schématu, které by volbou parametrů umožňovalo optimalizovat vlastnosti zálohování dle požadavků daného zákazníka.

Další možnost vývoje spočívá ve vytvoření rozšířené aplikace BackupCalc, umožňující pracovat multiplatformně. Aplikace by mohla být do budoucna využita jako agent a posílala by zálohovacímu serveru parametry pro vhodnější nastavení parametrů tohoto nově vytvořeného rotačního schématu.

LITERATURA

[1] Blu-ray Disc Association. *Blu-Ray Technologie - popis* [online]. [cit. 2014-10-28]. Dostupné z URL: <http://blu-raydisc.com/en/AboutBlu-ray/WhatisBlu-rayDisc/BDvs.DVD.aspx>.

[2] BURDA, K. *Mathematical model of data backup and recovery. International Journal of Computer Science and Network Security.* 2014, roč. 13, č. 10, s. 16-25 Dostupné z URL: <http://paper.ijcsns.org/07_book/201407/20140703.pdf>.

[3] CHERVENAK, A. L.; VELLANKI, V.; KURMAS, Z. *Protecting File Systems: A Survey of Backup Technogies* - Joint NASA and IEEE Mass Storage Conference, poslední aktualizace r. 1998. Dostupné z URL: <http://pdb-d.eng.uiowa.edu/~achiang/jclub/summer03/Chervenak_PFSASBT.pdf>.

[4] HP, IBM, Quantum. *LTO Technologie - popis* [online]. [cit. 2014-10-26]. Dostupné z URL: <http://www.lto.org/technology/>.

[5] MCGOWAN, P.; WEEKELY, S. *Quantum DLTtape Handbook - Eight Edition* [online]. [cit. 2014-10-26]. Dostupné z URL: <http://downloads.quantum.com/sdlt320/handbook.pdf>.

[6] Microsoft. *Azure Backup* [online]. [cit. 2014-10-29]. Dostupné z URL: <http://azure.microsoft.com/cs-cz/services/backup/>.

[7] NELSON, B. L. *Stochastic modeling: analysis & simulation. Mineola: Dover Publications*, 1995, xiv, 321 s. ISBN 04-864-7770-3.

[8] NELSON, S. *Pro Data Backup and Recovery. Berkeley, CA: Apress*, 2011, 280 s. ISBN 978-1-4302-2663-5.

[9] NETAPP. *Are All Snapshots Created Equal?: Data Protection [online].* Poslední aktualizace 13. 3. 2012, [cit. 2014-09-26]. Dostupné z URL: <http://community.netapp.com/t5/Technology/Are-All-Snapshots-Created-Equal/ba-p/83211>.

[10] NITIN, A.; BOLOSKY, W. J.; LORCH, J. R.; DOUCEUR, J. R. *A Five-Year Study of File-System Metadata. [online]. [cit. 2014-11-22].* Dostupné z URL: <http://research.microsoft.com/pubs/72896/fast07-final.pdf>.

[11] OTIPKA, P.; ŠMAJSTRLA, V. *Pravděpodobnost a statistika. 1.* vyd. Ostrava: Vysoká škola báňská - Technická univerzita, 2006. 266 s. ISBN 80-248-1194-4.

[12] PRESTON, G. *Enterprise systems backup and recovery: a corporate insurance policy.* Boca Raton: CRC Press, c2009, xix, 306 s. ISBN 14-200-7639-6.

[13] PRESTON, W. C. *Backup & Recovery: Inexpensive Backup Solutions for Open Systems.* Sebastopol, CA: O´Reilly Media, Inc., 2006. First Edition. ISBN 978-0-596-10246-3.

[14] RISCH, A. *Essential system administration. 3rd ed.* Sebastopol, CA: O'Reilly, 2002, xxiv, 1149 s. ISBN 05-960-0343-9.

[15] RiverBed. *RiverBed SteelFusion* [online]. [cit. 2014-10-29]. Dostupné z URL: <`http://www.riverbed.com/products/branch-office-data/backup-consolidation.html`>.

[16] SHIMONSKI, R.; SCHMIED, W. *MCSA/MCSE exam 70-292 study guide and DVD training system: managing and maintaining a Windows Server 2003 environment for an MCSA certified on Windows 2000.* Rockland, Mass.: Syngress, 2003, 737 s. ISBN 1-932266-56-9.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

DSP	číslicové zpracování signálů – Digital Signal Processing
IT	Informační Technologie
USB	Universal Serial Bus - běžné rozhraní na PC
LTO	Linear Tape Open - typ zálohovacího řešení
VTL	Virtual Tape Library - virtuální knihovna z disků
SAN	Storage Area Network - připojení úložiště přes síťovou distribuci
CD	Compact disk - typ optického zálohovacího média
BD	Blu-Ray Disk - typ optického zálohovacího média
DVD	Digital Versatile Disk/Digital Video Disk - typ optického zálohovacího média
DSS	Digital Data Storage - typ zálohovacího řešení
DAT	Digital Audio Tape - typ zálohovacího řešení
DLT	Digital Linear Tape - typ zálohovacího řešení
SDLT	Super Digital Linear Tape - typ zálohovacího řešení, nástupce DLT
VHS	Video Home System - typ zálohovacího řešení
DPM	Data Protection Manager - centralizovaný zálohovací software s rozšiřitelnými moduly
SQL	Structured Query Language - databáze s přístupem přes tento standard
IOPS	Input/Output Operations Per Second - používáno pro měření výkonu vstupně-výstupních operací úložiště
GFS	Grandfather, Father, Son - způsob rotace zálohovacích medií
CoW	Copy on Write - vytváření záloh pomocí snapshotů s nutností kopírovat originální data
RoW	Redirect on Write - vytváření záloh pomocí snapshotů bez nutnosti kopírovat originální data.

SLA	Service-level Agreement
GUI	Graphical User Interface
Amanda-Dif	Rotační schéma Amanda s použitím víceúrovňové zálohy na principu rozdílové zálohy
Amanda-Inc	Rotační schéma Amanda s použitím víceúrovňové zálohy na principu přírůstkové zálohy
Hanoi-Lev	Rotační schéma hanojské věže s použitím víceúrovňové zálohy
Hanoi-Full	Rotační schéma hanojské věže s použitím pouze plných záloh
5T	Rotační schéma pěti pásek

SEZNAM PŘÍLOH

A POPIS APLIKACE BACKUPCALC

Aplikace BackupCalc je vytvořená v C# a vyžívá standardního grafického rozhraní operačního systému Windows. Pro svůj běh vyžaduje .NET Framework 4.5 (což je v dnešní době běžná komponenta ve Windows prostředí).

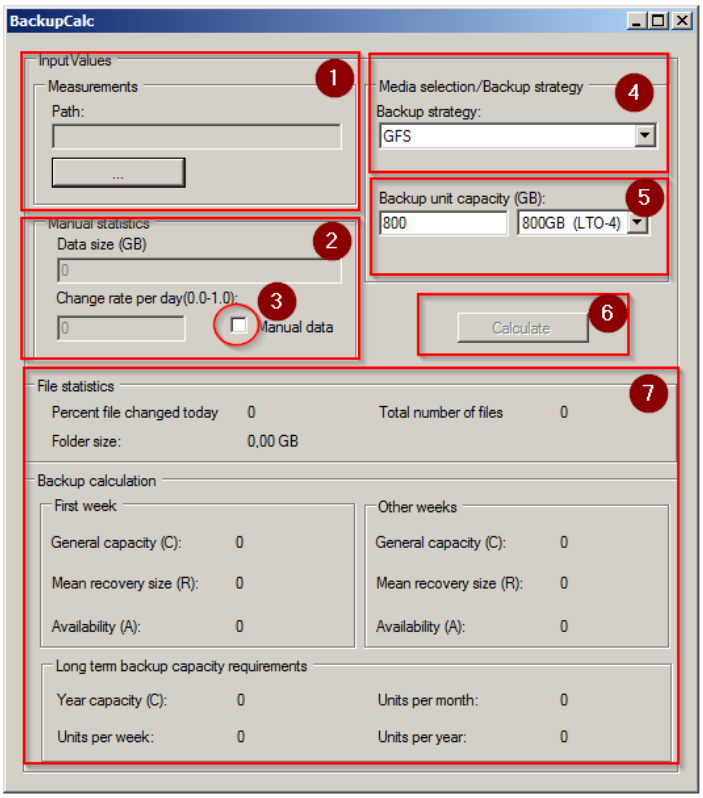
Hlavní myšlenkou aplikace BackupCalc je usnadnění volby nejvhodnějšího schématu pro zálohování. Aplikace umožňuje zvolit složku či disk a provést analýzu dat. V případě analýzy složky aplikace projde adresářovou strukturu a ověřuje čas poslední změny souboru a porovnává čas změny s aktuálním časem.

Aplikace sčítá celkový počet souborů, počet souborů změněných v daný den a počítá celkovou velikost analyzované složky (disku). Výše uvedené informace určí míru změn dané složky a dále vypočítat celkovou velikost složky a míru změn jako vstupní proměnné pro výpočty. Pokud jsou informace o velikosti složky (disku) a míry změn k dispozici, je možné je zadat ručně a aplikace v tom případě využije zadané údaje jako vstupní proměnné pro výpočty.

A.1 GUI aplikace

Na obrázku A.1 je patrné rozvržení GUI. Jednotlivé části podrobně popíšu a vysvětlím jejich účel. Popisované části jsou označeny čísly.

1. Výběr složky pro analyzovanou složku (disk), tlačítko „ ... “ slouží pro výběr složky, po dokončení výběru se složka objeví v poli „Path“.
2. Pokud jsou k dispozici informace o velikosti a je známá i míra změny, mohou se prostřednictvím tohoto pole data vložit.
3. Zaškrtávací políčko umožňuje manuální vložení dat do pole č. 2.
4. Výběr umožňuje zvolit rotační schéma.
5. V pravé části pole je možný výběr LTO technologií, zvolený typ média se v levém okně zobrazí jako kapacita daného média. Pokud se jedná o jiná média než LTO, je možné vstupní hodnotu zadat přímo do levého okna „Backup unit capacity (GB)“.
6. Po zvolení složky pomocí pole č. 1 lze spustit analýzu složky, aplikace v reálném čase přepočítává a zobrazuje již zanalyzovaná data v poli č. 7.
7. Pole č. 7 zobrazuje výsledky výpočtů aplikace, v horní části jsou k dispozici informace o analyzované složce. Níže jsou k dispozici informace o hlavních třech parametrech, celkového objemu záloh „General capacity (C)“, středním obnovovacím objemu „Mean recovery size (R)“ a o dostupnosti záloh „Availability“. Ve spodní části okna jsou k dispozici informace o nárocích na zálohovací úložiště, jako jednotka je zvolena hodnota zadaná v poli č. 5.



Obr. A.1: Aplikace BackupCalc

A.2 Dodatečné informace k aplikaci

BackupCalc je jednoduše použitelný program, který pracuje v režimu časového přerušení ve více vláknech. Hlavním důvodem použití více vláken je zamezení situace, kdy by výpočet obsahu složky způsobil „zamrznutí“ GUI a aplikace by neodpovídala a jevila by se jako nestabilní. Aplikace po spuštění aplikace očekává výběr vstupních informací pro výpočet záloh, ve výchozím stavu očekává výběr složky či disku, po výběru složky ale samotný výpočet informací nebude uskutečněn. Výpočet se spustí po stisknutí tlačítka „Calculate“. Aplikace v tom případě spustí výpočet, mezivýsledky jsou v reálném čase přepočítávány.

Aplikace umožňuje získat zálohovací statistiky i pomocí přímo zadaných vstupních dat, tedy velikosti úložiště a míry změn za 24 hodin v rozsahu od 0 do 1, kdy 0 představuje situaci, kdy k žádné změně nedošlo. Hodnota 1 představuje situaci, kdy se změnila všechna data na úložišti. V případě ručního zadání se využívají stejné proměnné jako pro analýzu dat ve složce. Virtuálně je celkový počet souborů nastaven na 100 a poměr nově vytvořených souborů je počítán z míry změny zadávané

v poli č. 2 do okna „Change rate per day(0.0-1.0)“.

Střední obnovovací objem pro rotační schéma „Amanda - Incremental“ není vztažen k prvnímu a druhému týdnu, ale až k druhému a třetímu týdnu. První týden v tomto rotačním schématu nelze považovat za skutečný obnovovací objem, nedošlo by k obnově všech dat. Od třetího týdne je již střední obnovovací objem konstantní, podobně jak je tomu u jiných rotačních schémat v třetím týdnu.

B OBSAH PŘILOŽENÉHO CD

CD obsahuje zdrojové soubory k aplikaci BackupCalc vytvořené v Microsoft Visal Express 2013 for Desktop. Přiložená tabulka obsahuje seznam adresářů a jejich obsah. Pro zjednodušení je uvedená předpokládaná cesta k disku CD označená jako R:.

Adresář	Popis obsahu
R:\Source	Zdrojové soubory projektu C# ve Visual Studio 2014.
R:\TextSrc	Zdrojové soubory bakalářské práce.
R:\Compiled	Zkompilovaný soubor „BackupCalc.exe“.