MĚŘENÍ VLASTNOSTÍ PILE SYSTÉMU: SPOTŘEBA DISKOVÉHO PROSTORU

Milan Tomeš

Univerzita Pardubice, Fakulta ekonomicko-správní, Ústav systémového inženýrství a informatiky

Abstract: *Pile is an innovative system architecture, which stores data into one complex structure. There is shown a theoretical assumption of storage requirement, which is experimentally verified.These experiments are presented: assimilating different types of input data and examining the correlation between the size of the input data, the size of the resultant Pile structure, and the number of relations created by the engine to assimilate the input data.*

Keywords: *Pile, Text Assimilation, Relation, Storing Data, Benchmarking*

1. Úvod

Tato práce se věnuje technologii Pile, která přichází s novým přístupem při práci s různými daty. Při použití Pile nejsou data do informačního systému kopírována, ale asimilována a vytváří tak jednu komplexní strukturu, kde se nevyskytuje redundance dat.

Tento článek si klade za cíl experimentálně ověřit jeden z teoretických předpokladů, který se týká vlastnosti Pile struktury a to je spotřeba diskového prostoru v závislosti na velikosti vstupních dat.

Jsou provedeny dva druhy experimentů a to za prvé asimilace různých typů vstupních dat a sledování závislosti mezi velikostí vstupu a výstupu a za druhé množství relací vytvořených vybranou sw implementací při asimilaci vstupních dat. Protože je však oblast použití Pile značně rozsáhlá, tento článek se zaměřuje pouze na práci s textovými soubory.

V článku jsou nejprve popsány vlastnosti technologie Pile nezbytné pro pochopení jejích specifik a významu provedených experimentů. Dále jsou popsány podmínky, v jakých experimenty proběhly a znázorněny vlastní výsledky s příslušnými komentáři. V závěru jsou pak výsledky zhodnoceny a navrženy další postupy.

2. Popis problematiky

V této je provedena obecná formulace problematiky seznámení s vlastním Pile systémem pro pochopení specifik toho systému.

2.1 Popis technologie Pile

Jak je uvedeno v [3] jedná se o jednoduché matematické řešení, založené na grafové struktuře, poskytující uspořádané vztahy, které jsou spojené rekurzivním způsobem, umístěné ve třech dimenzích, které společně tvoří jeden celek vzájemných vztahů.

Pile je radikálně odlišný přístup k datům, datovým strukturám a práci s výpočetní technikou, založený na spojeních, neboli relacích, ale nikoliv ve smyslu současných databázových systémů. Namísto znázorňování a ukládání dat, Pile zaznamenává pouze vztahy mezi základními datovými prvky a skladuje vztahové vzory ve zcela odlišné struktuře. Pile umožňuje simulovat, manipulovat a propojovat libovolná data, reprezentovat a vytvářet uživatelská data pouze ze vztahových vzorů.

Dle [3] je pro porozumění Pile přístupu třeba rozlišovat mezi uživatelskými a vnitřními daty. Uživatelská data jsou to, co obvykle uchovávají databáze nebo soubory, např. texty, adresy, účty, hudbu a tak dále. Ta nejsou nikdy ukládána ve strukturách Pile systému, zůstávají externí a nejsou dále vyžadována, jakmile jsou jednou asimilována do Pile struktury. V tomto systému jsou jen spojení a spojení mezi spojeními. Při porovnání s tradičním způsobem práce je tento systém homogenní, neboť jsou zde jen spojení. Více je tato problematika rozpracována v [2].

Základním smyslem Pile je registrovat data a být schopný data reprezentovat, ale ne data skladovat. Pile registruje data kompletně a bezeztrátově pomocí jejich asimilace, ne pomocí jejich kopírování a ukládání do struktury. Asimilace dat v kontrastu s ukládáním dat znamená, že položka uživatelských dat (množina libovolných datových elementů) je rozložena do relací mezi jednotlivými datovými elementy, jak ukazuje Obr. 6. [3]

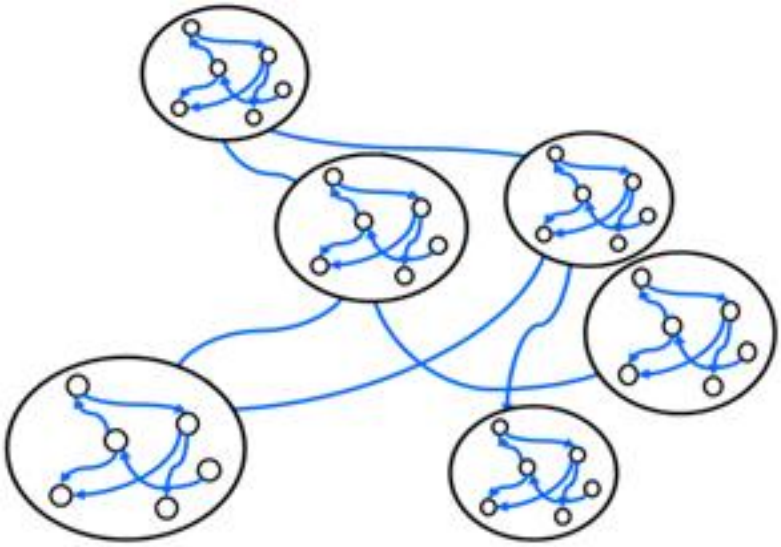
V současné výpočetní technice jsou data ukládána s obrovskou redundancí, každý řetězec, který se vyskytuje v různých dokumentech v jednom nebo různých adresářích je ukládán zvlášť. To je náročné nejen na spotřebu diskového prostoru, ale také na vyhledávací čas. [1]

Zdroj [1] uvádí příklad: *The rain was over and the sun was shining again*.

Redundance je významná. Písmeno „*a*“ a „*n*“ se vyskytuje 6x, „*s*“ 4x, dvojice „*in*“ 4x, atd. V důsledku toho velikost dokumentů, či datových struktur prudce roste, neboť znaků jsou v různých znakových sadách stovky a počet slov v jazycích jsou tisíce.

Naproti tomu náš nervový systém pracuje se vztahy. Každá interakce organismu s jeho prostředím je representována v jeho nervovém systému jako relace jistých stavů. Zacházení s těmito relacemi jako nezávislými entitami a interakce mezi nimi (vytváření nových vztahů se starými vztahy) se dá popsat jako myšlení. [1]

Pile také pracuje se vztahy (relacemi). Pile spojení je relace mezi dvěma dalšími Pile spojeními, je to referencovatelný objekt a může se s ním zacházet jako s nezávislou entitou. Tak můžeme prezentovat Pile jako síť vztahů (relací), jak ukazuje Obr. 6. [1]



Obr. 6: Pile jako síť vztahů

Zdroj: [1]

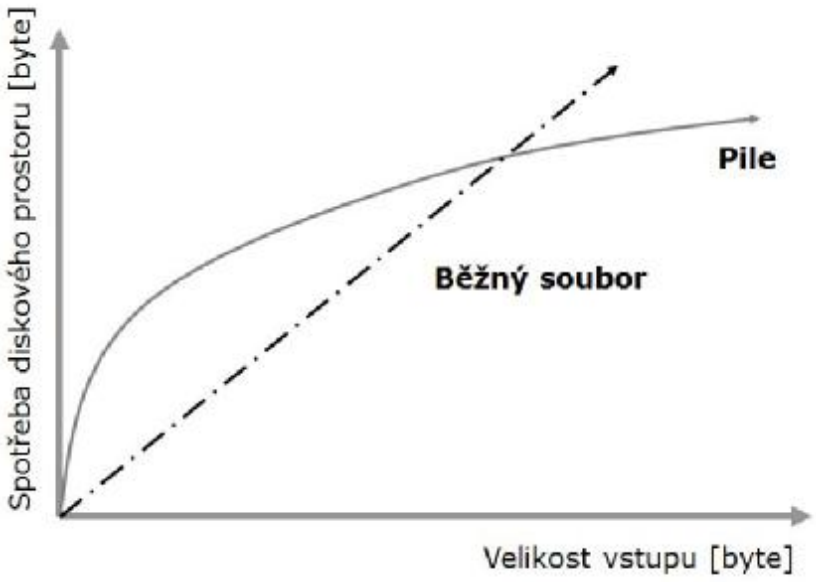
Na tomto obrázku je znázorněno, že Pile systém jako celek je objekt, který se skládá ze sítí vztahů mezi objekty, z nichž každý objekt sám může být dalším objektem složeným z dalších sítí vztahů mezi objekty, tedy že je to vlastně relace relací mezi relacemi s tím, že každý objekt (znázorněný na obrázku černým kruhem) může mít jinou úroveň detailnosti.

Pile systém má několik vlastností, které jsou nezbytné pro vytváření systému [1]:

- Konektivita – každé dva objekty v Pile systému mohou být spojeny
- Heterogenita – různé typy vstupních hodnot mohou být spojeny do Pile struktury
- Samo odkazovací – dynamika systému je specifikována systémem samotným.
- Distribuovaná struktura - může mít vice začátků, vnitřních hierarchií i konců
- Rozšiřující se – Pile systém může vzrůstat přidáváním nových spojení
- Vícenásobná dědičnost – každý Pile objekt má dva rodiče
- Spojení a objekty jsou stejné instance

2.2 Teoretické předpoklady

K základním vlastnostem Pile struktury patří, že každý objekt, pokud se již jednou vyskytuje, je znovupoužit a ve struktuře se nevyskytuje redundance. Předpokládá se, že se projeví jistý kompresní efekt, jak ukazuje následující Obr. 7.

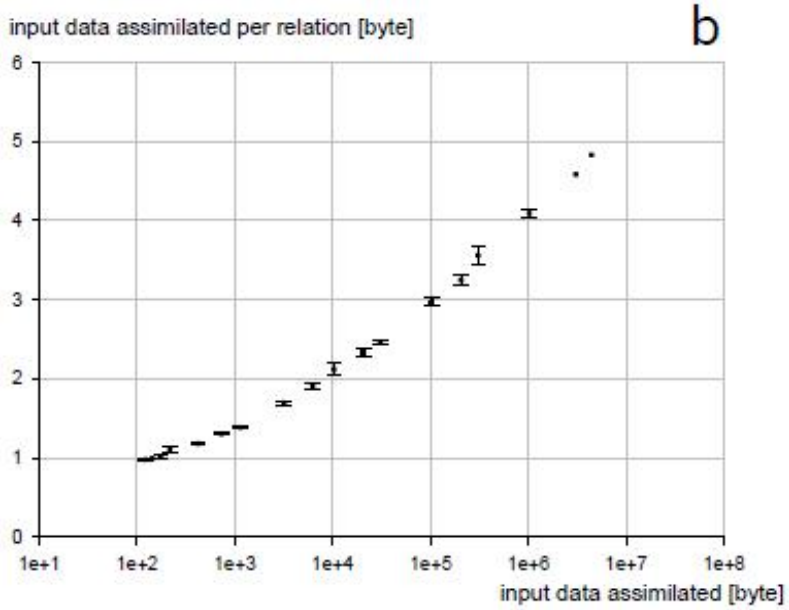


Obr. 7: Teoretický předpoklad velikosti struktury Pile

Zdroj: Upraveno na základě [1]

Na obrázku je znázorněno, že u běžného souboru se s velikostí vstupu rovnoměrně zvětšuje velikost výstupu (čerchovaná přímka je osa grafu). Naproti tomu u Pile je z počátku na tvorbu struktury spotřebováno více místa, což je dáno potřebou vytvořit mnoho základních relací. Nicméně v pozdější fázi se již mnoho relací opakuje a tak můžou být znovupoužity a není třeba vytvářet nové. Tím se spotřeba místa snižuje.

Obr. 8 ukazuje teoretický předpoklad, který zdůvodňuje předchozí obrázek. Tím, že se počet relací snižuje, ale velikost vstupních dat roste, tak s růstem vstupu by se měl zvyšovat poměr velikosti dat na jednu relaci.



Obr. 8:Teoretický předpoklad množství dat na jednu relaci.

Zdroj: [5]

3. Podmínky měření

3.1 Vstupní data

Měření bylo prováděno výhradně na textových souborech. Ty byly získány z Projektu Gutenberg [6]. Jednalo se o literární díla v angličtině, jak ukazuje následující Tab. 6:

Tab. 6: Zdrojová data

Název díla	Název souboru	Velikost souboru [kB]
Zápisky Leonarda da Vinci	7ldvc10.txt	1392
Sun Tzu: The Art of War	132.txt	344
Pohádky bratří Grimmů	2591.txt	550
Příběhy Sherlocka Holmese	advsh12.txt	590
King James bible	kjv10.txt	4445
Kompletní dílo W. Shakespeakera	shaks12.txt	5583

Zdroj: vlastní

Převážně však byla využívána Bible, která byla pro potřeby měření rozdělena do souborů podle jednotlivých knih. Důvodem bylo samozřejmě používání podobného jazyka a slov, tedy aby mohlo dojít k co nejlepší znovupoužitelnosti celé Pile struktury.

3.2 Nástroje

Pro měření Pile byl použit *Pile Concatenations procesor*. Dle [4] jde o poměrně rozsáhlý projekt od firmy PileSystems inc., která se zabývala komerčním nasazením Pile technologie. Jeho cílem bylo prezentovat možnosti Pile na reálných datech a experimentálně ověřit vlastnosti Pile, přesto stále zůstává na úrovni demo aplikace.

Obsahuje původní experimentální *Pile engine*, který byl implementován autorem této technologie Erez Elulem. Je to stále prototyp (napsán v jazyce C/C++), může však být využit jednak pro budoucí vývoj a také jako nástroj pro vyhledávání v textových souborech.

Je založen na principu vícevrstvé architektury. *Pile space* je logický prostor, kde jsou data skladována, bez ohledu na jejich význam. *Pile engine* je jádro, které

umožňuje vytváření a údržbu relačního Pile prostoru. *Pile agent* dává k dispozici nástroje pro práci s Pile prostorem, je uzpůsoben pro určitý typ dat (text, biologická data, apod.) a *Pile klient* je front-end vrstva pro práci s konkrétními daty (textové soubory, apod.).

Z tohoto pohledu je v tomto článku hodnocena výkonnost dvojice *engine-agent* jako celku.

Dle [4] je toto demo určeno pro účely testování a prezentaci možností Pile technologie. Pile engine je zde použit jako *zřetězující procesor*, ve smyslu spojování textových řetězců. Tomuto způsobu zpracování se také říká asimilace dat (viz kapitola 1).

Aplikace pracuje tak, že asimiluje textové soubory, zadané uživatelem a ukládá logovací soubory se záznamy o vlastnostech Pile struktury, ze kterých pak byly realizovány výstupy uváděné v tomto článku.

4. Vlastní měření

V této kapitole jsou prezentovány jednotlivá měření a výstupy, kterých bylo dosaženo. Z toho je zřejmé, že zde leží těžiště celé práce.

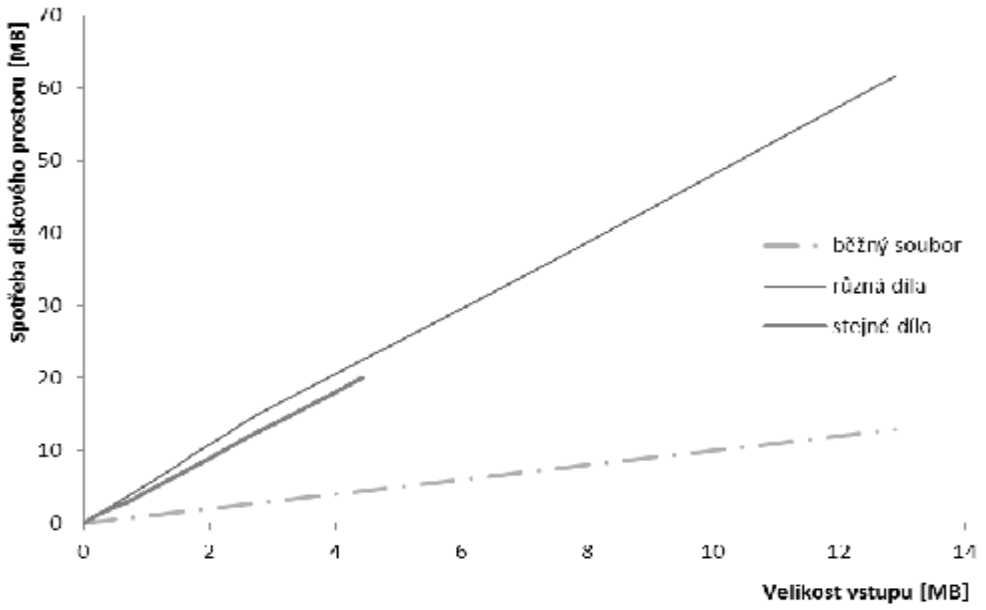
Jsou realizovány tyto dvě základní skupiny experimentů:

- Závislost mezi velikostí vstupních dat a velikostí výsledné struktury
- Počet relací vytvořených při asimilaci vstupních dat

4.1 Soubory s různými autory

Byla měřena spotřeba diskového prostoru (velikost výstupu) v závislosti na velikosti vstupu pro různé soubory s různými autory (viz Tab. 6). Byly použity různé zdrojové soubory a vkládány do výše uvedeného sw aplikace.

Výsledky prezentuje Obr. 9. Zde je pro větší přehlednost opět znázorněna čára běžného souboru. Je vidět, že graf je poněkud zdeformován a to z důvodu větší přehlednosti (čerchovaná čára by měla být vždy osa grafu pod úhlem 45 stupňů).



Obr. 9: Spotřeba diskového prostoru v závislosti na velikosti vstupu – různá díla

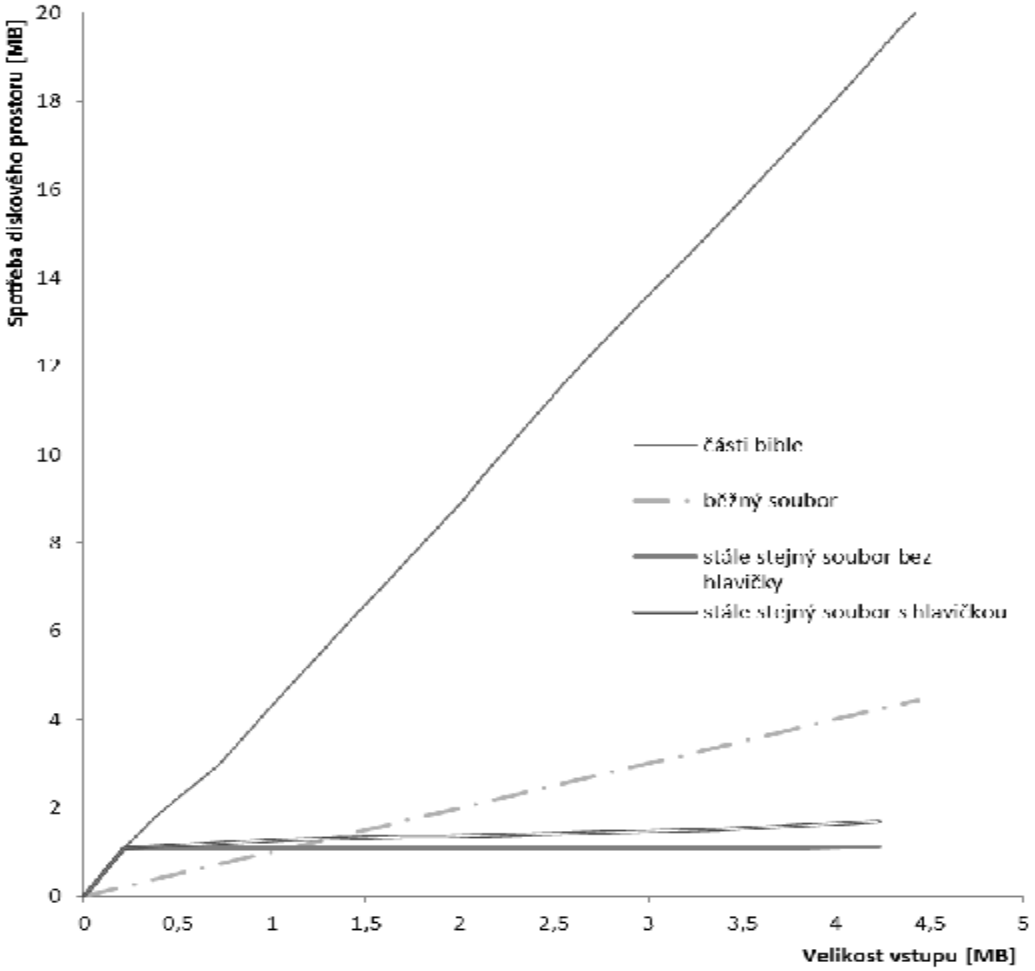
Zdroj: vlastní

Dále je zde pro porovnání znázorněna křivka z následujícího měření, která prezentuje, velikost struktury, pokud je text ze stejného literárního díla (plná silná čára).

Z Obr. 9 vyplývá, že spotřeba diskového prostoru Pile je naprosto neúměrně vysoká a práce s pamětí neefektivní. Na asimilaci cca 12 MB dat spotřebovala cca 65 MB dat.

4.2 Soubory s jedním autorem

Byla měřena spotřeba diskového prostoru (velikost výstupu) v závislosti na velikosti vstupu pro různé soubory, ale s podobným obsahem. Pro tyto potřeby byla použita Bible, rozdělená do 58 částí, podle jednotlivých knih.



Obr. 10: Spotřeba diskového prostoru v závislosti na velikosti vstupu – stejná díla

Zdroj: vlastní

Obr. 10 znázorňuje měření s různými variacemi vstupního jediného vstupního souboru, kde:

- Čerchovaná čára opět představuje osu pod úhlem 45°, pro lepší přehlednost
- Slabá plná čára znázorňuje křivku jednoho souboru, jehož části byly postupně asimilovány. Zde jednotlivé knihy z bible, jak je uvedeno v kapitole 3.
- Plná silná tmavá čára zobrazuje variantu, kdy na vstup byl vkládán opakovaně ten samý soubor (První kniha z Bible). Toto znázorňuje situaci, kdy by teoreticky nemělo dojít k žádnému růstu struktury - je to vlastně limitní případ znuvupoužitelnosti .
- Plná silná světlá křivka zobrazuje variantu, kdy na vstup byl také opakovaně vkládán ten samý soubor, ale byla v něm měněna hlavička, která obsahovala pořadové číslo (opět první kniha z Bible), důvod pro uskutečnění tohoto měření je uveden níže.

V hlavičce bylo uvedeno:

```
King James Bible, file number: ##
```

Z Obr. 10 vyplývá, že:

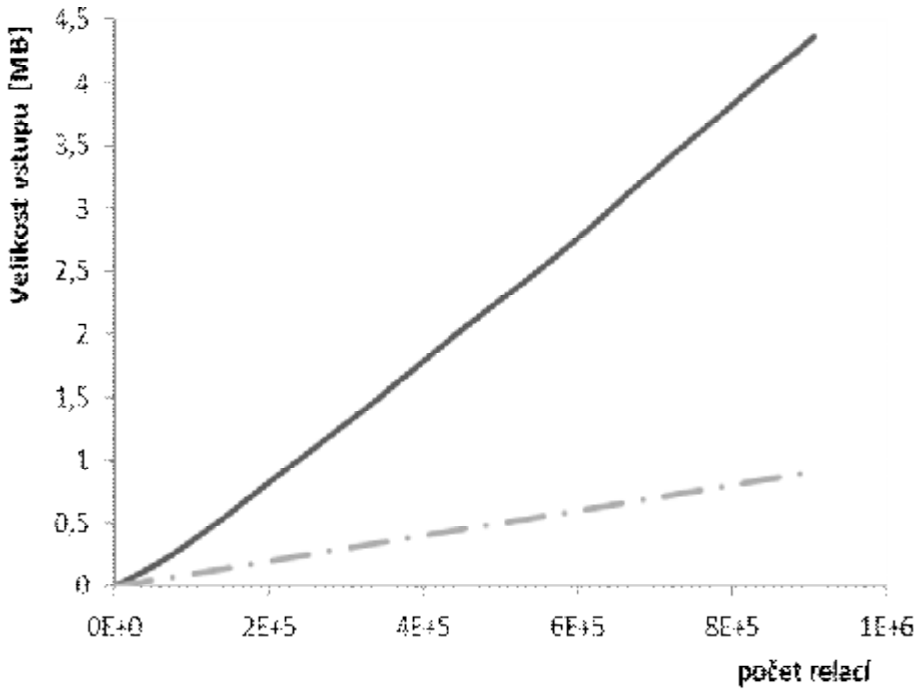
- Při běžném použití Pile neprokázal degresivní růst spotřeby diskového prostoru, ani když se jedná o velice podobné texty se stále se opakujícími výrazy. Naopak je lineární s prudkým sklonem.
- Pokud byl vkládán stále stejný soubor, aplikace to poznala, další nevkládala, vždy znovupoužila jediný soubor. Toto však je spíše záležitost softwarové implementace, než Pile struktury.
- Proto bylo provedeno ještě jedno měření, kdy byla do souboru záměrně vkládána proměnlivá hlavička. V tomto případě je vidět, že růst spotřeby diskového prostoru se sice projevil, ale podstatně menší, než v případě jednotlivých částí bible. Je tedy vidět, že úspor dosáhnout lze.

4.3 Měření počtu relací

V této kapitole jsou znázorněny další výsledky měření vlastností Pile. Jak bylo řečeno výše, z logovacích souborů je k dispozici několik parametrů, z nichž byly použity počet vytvořených relací, poměr počtu relací ku velikosti vstupních dat a její převrácená hodnota.

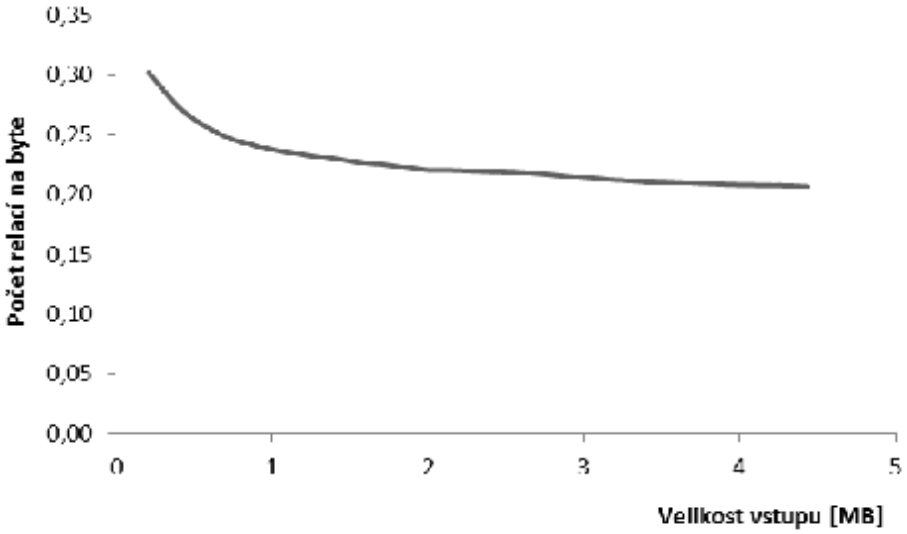
Jako vstupní data byly použity jednotlivé knihy Bible.

Získaná data znázorňuje Obr. 11, Obr. 12 a Obr. 13. Čerchovanou čarou je zobrazen běžný soubor (osa), přerušovanou čarou teoretický předpoklad a plnou čarou naměřená skutečnost.



Obr. 11: množství asimilovaných dat v závislosti na počtu relací

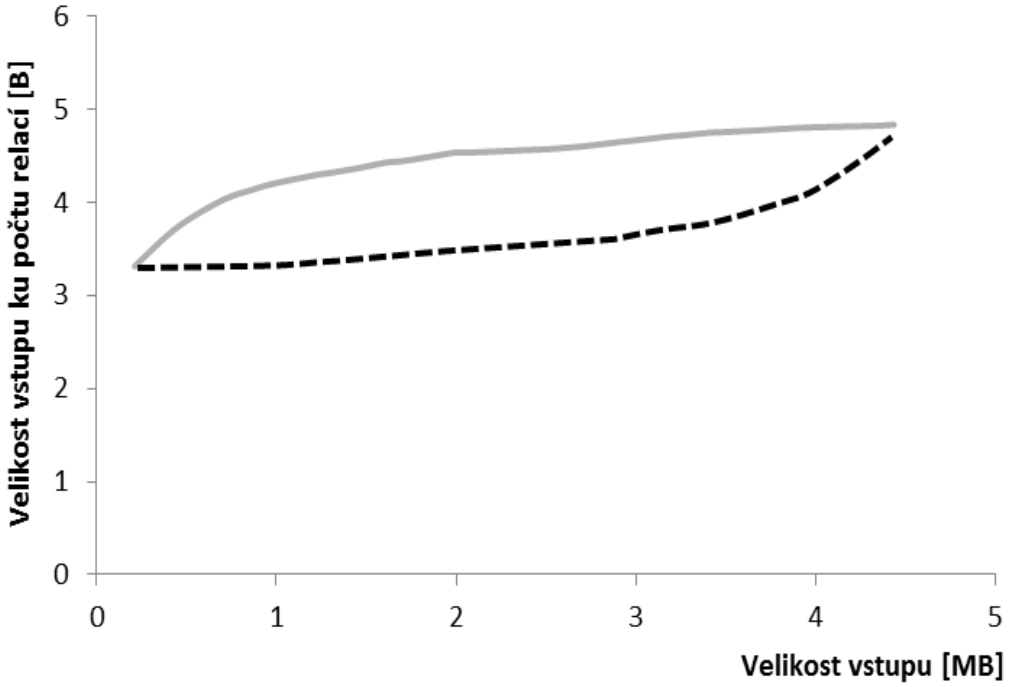
Zdroj: vlastní



Obr. 12: Počet relací vztažený na jeden byte v závislosti na velikosti vstupu

Zdroj: vlastní

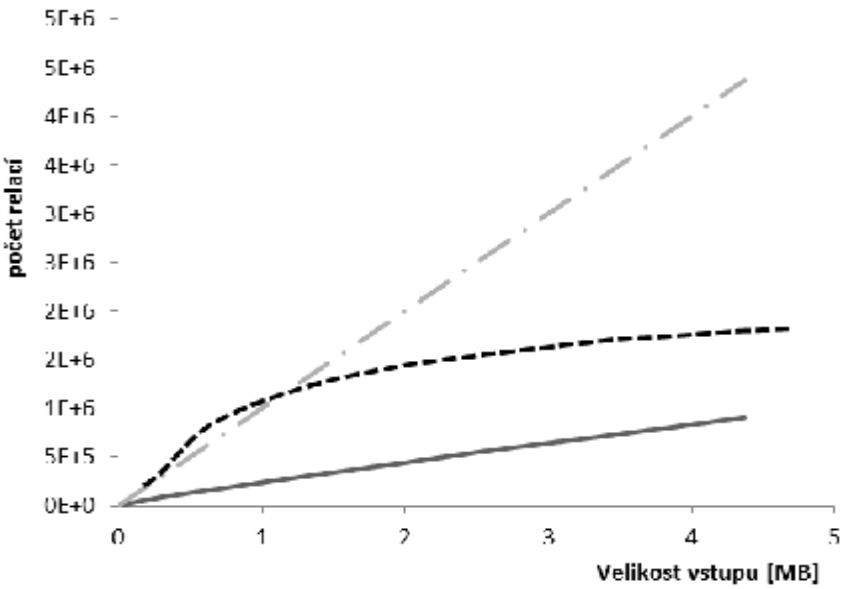
Z uvedeného Obr. 11 vyplývá, že množství asimilovaných dat roste rychleji, než počet relací, což je pozitivní a ověřuje tak růst kapacity Pile struktury. Toto zvyšování kapacity prokazuje i Obr. 12, kde je vidět, jak s růstem velikosti vstupu se počet relací nutných na asimilaci vstupních dat snižuje.



Obr. 13: Množství dat na jednu relaci

Zdroj: vlastní

Nicméně Obr. 13 znázorňuje negativní fakt, že množství dat připadajících na jednu relaci roste degresivně, ale předpokladem bylo, že poroste progresivně (viz Obr. 8) a prokáže tak, že s rostoucím množstvím asimilovaných dat bude větší kapacita každé nové relace.



Obr. 14: Množství vytvořených relací v závislosti na velikosti vstupu

Zdroj: vlastní

Obr. 14 znázorňuje to, že počet relací s růstem vstupu roste méně prudce, než samotná velikost diskového prostoru (což je však logické). Nicméně se zde potvrzuje negativní jev vyplývající již Obr. 10, tedy že zde počet relací neroste degresivně, ale lineárně, což není dobře.

5. Závěr:

Ze získaných dat vyplývají tyto závěry:

- Záleží na datech, která jsou asimilována, pro různé autory se tvoří různě veliká struktura.
- Jednoznačně vyplývá, že původní teoretický předpoklad dle Obr. 7 a Obr. 8 se nepodařilo ověřit.
- Získané výsledky neodpovídají tomu, co uvádí [5], viz srovnání Obr. 8 a Obr. 13.
- Testovaná Pile implementace je velice neefektivní v práci s pamětí. Protože se jedná o hodnocení výkonnosti dvojice engine-agent, je otázkou, která část má závažnější vliv a toto bude předmětem dalšího zkoumání..
- Je otázka, proč počet relací a tudíž i velikost výsledné struktury roste lineárně a ne logaritmicky, jak se předpokládalo. Toto bude taktéž předmětem dalšího zkoumání.

- Je zde jistý rozpor mezi Obr. 9 + Obr. 14 versus Obr. 11 + Obr. 12. Na jedné straně je vidět lineární růst spotřeby prostoru a počtu relací, na druhé straně nelineárně se snižující relativní velikost struktury vzhledem k počtu relací.

I přesto, že předpokládané teoretické předpoklady se nepodařilo potvrdit, cíl práce byl splněn, neboť práce dospěla ke konkrétním závěrům. Dále je třeba si uvědomit, že testovaná sw implementace je pouze demoverzí a není tak optimalizovaná na tento typ experimentů. Při dalších měřeních, která budou na této implementaci prováděna, je třeba se zaměřit zejména na funkcionality, které technologie Pile nabízí. Další možností je vytvoření nové implementace, ale to se v současné době nepředpokládá, je třeba nejprve důkladně zanalyzovat tu stávající.

Použité zdroje:

[1] TOMEŠ, M., The Pile system, Scientific papers of the University of Pardubice – Series D Faculty of Economic and Administration 12 (2007), 200--208,(2007), ISBN 978-80-7395-040-8

[2] WESTPHAL, R., *Storing Relations Instead of Data - Just a Cool Idea or a Revolutionary New Data Storage Paradigm?,* [online], 2006, [cit. 2009-11-30]. Dostupné z WWW: <http://weblogs.asp.net/ralfw/archive/2005/12/08/432665.aspx>

[3] TOMEŠ, M., *Application of Pile technology in substrings search*, Scientific papers of the University of Pardubice – Series D Faculty of Economic and Administration 13 (2008), 180—187, (2008), ISBN 978-80-7395-149-8

[4] TOMEŠ, M., *Pile systém a jeho implementace*. In *The 10th Internationational Conference of Postgraduate Students and Young Scientists in Informatics, Management, Economics and Administration IMEA 2010*. Pardubice : [s.n.], 2010. s. 101. ISBN 978-80-7395-254-9.

[5] REUTER, D.: *Processing Data by Assimilating Pure Relations - Benchmarking the Pile System*, [online], 2006, [cit. 2009-11-30]. Dostupné z WWW <http://www.pilesys.com /new/Documents/Pile%20Benchmark.pdf >

[6] *Free eBooks by Project Gutenberg*, [online], 2008-11-03, [cit. 2010-09-12]. Dostupné z WWW <http://www.gutenberg.org>

Kontaktní adresa:

Ing. Milan Tomeš
Univerzita Pardubice
Fakulta ekonomicko-správní, Ústav systémového inženýrství a informatiky
Studentská 84
53002, Pardubice
Email: milan.tomes@upce.cz
Tel. č.: +420466036147