

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
KATEDRA INFORMATIKY



Vizualizácia súborového systému

VIZUALIZÁCIA SÚBOROVÉHO SYSTÉMU

BAKALÁRSKA PRÁCA

MARTIN ŠKORUPA

UNIVERZITA KOMENSKÉHO V BRATISLAVE

Fakulta matematiky, fyziky a informatiky

Katedra informatiky

Vedúci bakalárskej práce

Mgr. Matej Novotný

Stupeň kvalifikácie: bakalár

Bratislava 2007

Čestne prehlasujem, že som túto prácu vypracoval samostatne s použitím uvedenej literatúry a zdrojov pod odborným vedením Mgr. Mateja Novotného

V Bratislave 15. júna 2007

.....
Martin Škorupa

Predovšetkým by som sa chcel poďakovať môjmu školiťovi Mgr. Matejovi Novotnému za jeho cenné rady a nápady pri tomto projekte. Ďalej Jaroslavovi Šoltýsovi a Martinovi Štefčekovi za ich pomoc pri implementácii. A v neposlednom rade svojim rodičom a starým rodičom, ktorí mi umožnili študovať a počas štúdia ma podporovali.

Martin Škorupa

Abstrakt

ŠKORUPA, Martin: Vizualizácia súborového systému. [Bakalárska práca] / Martin Škorupa. Univerzita Komenského v Bratislave. Fakulta matematiky, fyziky a informatiky; Katedra informatiky. Školiteľ: Mgr. Matej Novotný. Stupeň odbornej kvalifikácie: Bakalár štúdia informatiky. Bratislava: FMFI UK, 2007.

Predkladaná práca predstavuje rôzne spôsoby vizualizácie stromov, špeciálne na príklade súborového systému. Hodnotí a porovnáva jednotlivé vizualizácie z hľadiska poskytovania vizuálnej informácie. Rozoberá rôzne problémy vznikajúce pri práci so súborovým systémom a navrhuje ich riešenia pomocou primeranej vizualizácie.

Vytvorená aplikácia na vizualizáciu súborového systému pomocou metódy interring v praxi implementuje získané poznatky. Umožňuje rôzne spôsoby vizualizácie, ktoré sa využívajú na riešenie vyhľadávacích problémov. Kladie dôraz na jednoduché ovládanie a prehľadnosť. Na prevenciu change blindness využíva animáciu zmeny stavu. Využíva aj focus+context prístupy.

Kľúčové slová: Vizualizácia. Súborový systém. Interring.

Obsah

Abstrakt	4
Obsah.....	5
Zoznam ilustrácií.....	5
Zoznam skratiek	6
Úvod.....	7
Zadanie.....	7
Cieľ.....	7
1 Základné pojmy.....	8
2 Vizualizácia stromov	10
2.1 Systém zobrazovania stromových štruktúr	10
2.2 Prehľad zobrazovania stromových štruktúr	11
2.3 Vizualná informácia stromu.....	18
2.4 Porovnanie vizualizácií	18
3 Súborový systém.....	19
3.1 Význam súborového systému a praktické problémy	19
3.2 Vlastnosti	19
3.3 Účel vizualizácie	20
4 Popis aplikácie	21
5 Príklady.....	27
5.1 Príklady z vizualizácie pomocou aplikácie InterRing.....	27
5.2 Riešenie vyhľadávacích problémov pomocou aplikácie InterRing	29
6 Záver.....	32
Zoznam použitej literatúry.....	33
Obsah CD	34

Zoznam ilustrácií

Obr. 1 Schéma Rooted tree	11
Obr. 2 Príklad Rooted tree v Total Commanderi	12
Obr. 3 Schéma Radial tree	12
Obr. 4 Príklad Radial tree.....	13

Obr. 5 Schéma Balloon tree	13
Obr. 6 Príklad Balloon tree	14
Obr. 7 Schéma Cone tree	14
Obr. 8 Príklad Cone tree.....	15
Obr. 9 Schéma Treemap.....	15
Obr. 10 Príklad Treemap pri mapovaní trhu	16
Obr. 11 Príklad Treemap v StepTree	16
Obr. 12 Schéma Interring.....	17
Obr. 13 Príklad Interring v InterRing.....	17
Obr. 14 Načítaný adresárový strom	22
Obr. 15 Pohyb v zobrazení.....	23
Obr. 16 Zvýraznenie adresára / súboru	23
Obr. 17 Zmena farebnej schémy	24
Obr. 18 Zmena mapovania farby	25
Obr. 19 Zmena mapovania šírky.....	26
Obr. 20 Zobrazenie predchádzajúceho stromu.....	26
Obr. 21 Príklad 1	27
Obr. 22 Príklad 2	28
Obr. 23 Príklad 3	28
Obr. 24 Riešenie 1	29
Obr. 25 Riešenie 2.....	30
Obr. 26 Riešenie 3.....	31

Zoznam skratiek

BČZM – farebná schéma biela – červená – zelená – modrá

Úvod

Zadanie

Názov témy:

Vizualizácia súborového systému

Špecifikácia:

Súborový systém je hierarchická stromová štruktúra obsahujúca informácie o uzloch (adresáre) a listoch (súbory). Tieto informácie napomáhajú užívateľovi lokalizovať svoj cieľ v súborovom systéme, upravovať túto hierarchiu podľa svojich predstáv a analyzovať stav existujúcej hierarchie. Komplikovaná štruktúra súborového systému sťažuje riešenie týchto úloh a preto vizualizácia poskytuje okamžitý a prehľadný prístup k mnohým z týchto informácií. Navyše, pomocou agregácie umožňuje pristupovať k rôznym úrovniam detailu podľa požiadaviek užívateľa.

Cieľ

Cieľom bakalárskej práce je implementovať vizualizáciu stromových grafov pomocou metódy interring na zobrazenie súborového systému. Tá má umožňovať mapovanie rôznych vlastností adresárov a súborov na jednotlivé parametre vizualizácie. Výhodou takéhoto prístupu je jednoduchšie hľadanie adresárov a súborov z danou vlastnosťou. Cieľom nie je zostrojiť nástroj na správu súborového systému, ale nástroj na zobrazovanie a riešenie vyhľadávacích problémov.

Predmetom práce je tiež porovnať rôzne spôsoby vizualizácie súborového systému a popísať ich prínos pri riešení týchto problémov v zobrazovaní súborovej hierarchie.

1 Základné pojmy

V tejto sekcii zdefinujeme niektoré pojmy používané v práci. Definície sú v niektorých prípadoch menej rigorózne ako zaužívané formálne definície. Je to v záujme jednoduchosti a pochopiteľnosti práce.

Graf

- Graf $G = (V, E)$, kde V je konečná, neprázdna množina vrcholov a E je množina hrán medzi vrcholmi.

Strom

- Strom je neorientovaný, acyklický graf G .

Hĺbka vrcholu

- Počet hrán, ktoré najmenej treba prejsť od koreňa k vrcholu.

Hĺbka stromu

- Maximum z hĺbok všetkých vrcholov stromu

Level stromu

- Množina vrcholov s rovnakou hĺbkou.

Otec

- Pre daný vrchol je otcom susedný vrchol, ktorý má menšiu hĺbku.

Syn

- Pre daný vrchol je syn susedný vrchol, ktorý má väčšiu hĺbku.

Vetva stromu

- Pre daný vrchol je to podstrom, pre ktorý je daný vrchol koreň.

Kruhový odsek

- Majme dve polpriamky vychádzajúce z jedného bodu a kružnicu so stredom v danom bode. Kruhový odsek je doplnok ku kruhovému výseku, určenom týmito priamkami a kružnicou, v priestore medzi danými polpriamkami.

2 Vizualizácia stromov

Stromy, ako súčasť teórie grafov sa využívajú v rôznych oblastiach. Veľké využité majú napríklad pri počítačových sieťach, sociálnych sieťach, rozhodovacích stromoch, na reprezentáciu rôznych vzťahov (rodokmeň) a inde.

Pre tieto aplikácie je potrebná vizualizácia grafov a stromov, zobrazujúca ich štruktúru a vlastnosti, ktorá je oproti dátovému uloženiu v počítači pre človeka lepšie čitateľná.

2.1 Systém zobrazovania stromových štruktúr

K stromovej dátovej štruktúre uloženej v počítači pristupujeme cez koreň. Všetky vrcholy, ktoré majú hranu s koreňom sa považujú za jeho synov a koreň sa považuje za ich otca. Všeobecne sa pri dvojici vrcholov, ktoré spája hrana, považuje za otca vrchol, ktorý má menšiu hĺbku a za syna ten, ktorý má väčšiu hĺbku. Všeobecne platí, že pre každú hranu spájajúcu dva vrcholy vieme povedať, ktorý z vrcholov je otec a ktorý syn.

Pri zobrazovaní stromových štruktúr sa vychádza z dátovej štruktúry, v ktorej je graf uložený:

```
Class TGraph {
    TNode *root;                //pointer na koreň grafu
    Function draw();
}

Class TNode {
    TContent *value;            //informácie uložené vo vrchole
    TNode *son[];               //množina synov
    Function draw(TArea *area); //vykreslovacia funkcia
}
```

Na zobrazovanie sa teda využíva rekurzívne volanie vykresľovacej funkcie:

```
Fucntion draw(TArea *area) {
    vykresli vrchol do area;
    for i = 0 to son.Count -1 {
        son[i].draw(new TArea(getPartOfArea(area, son.Count)));
    }
}
```

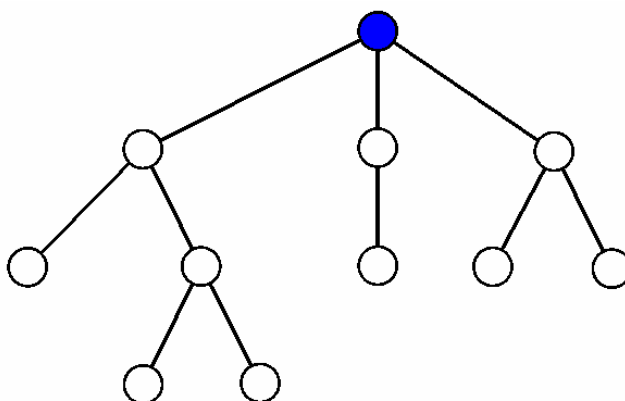
Pri vykresľovaní sa prenáša informácia o mieste, kam sa má daný vrchol vykresliť. Následne sa prepočítava miesto na vykreslenie synov.

2.2 Prehľad zobrazovania stromových štruktúr

Rooted tree

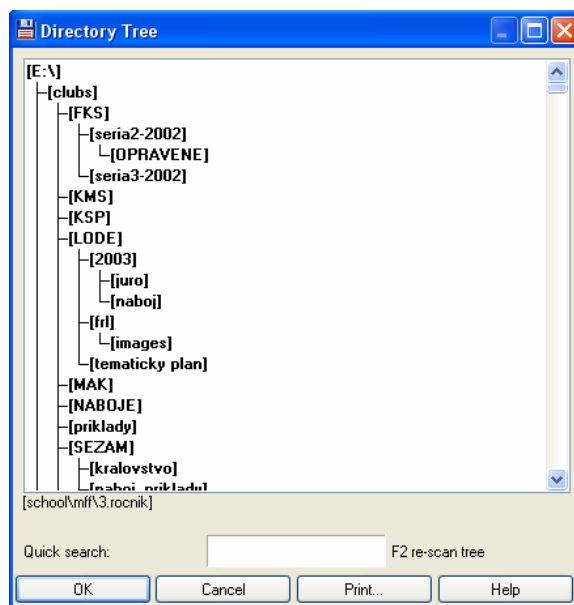
Rooted tree je základné zobrazenie stromovej štruktúry. Koreňový vrchol dostáva ako priestor na vykreslenie pre seba a svoju vetvu, teda celý strom, polrovinu určenú úsečkou a smerom v ktorom strom rastie. Vrchol sa vykreslí na danú úsečku, zvyčajne na jej stred. Synovia a ich vetvy dostávajú úsečku rovnobežnú s otcovou úsečkou a s rovnakou dĺžkou. Tá je rozdelená na rovnako častí ako je počet synov, pričom každá časť patrí jednému synovi. Jednotlivé rovnobežné úsečky predstavujú level grafu.

Nevýhodou tohto zobrazenia je slabé využitie priestoru. Maximálne sa môže pokryť polrovina, pri binárnom grafe len štvrtina priestoru.



Obr. 1 Schéma Rooted tree

Vizualizáciu súborového systému pomocou Rooted tree podporuje veľa programov. Jedným z nich je napríklad Total Commander.

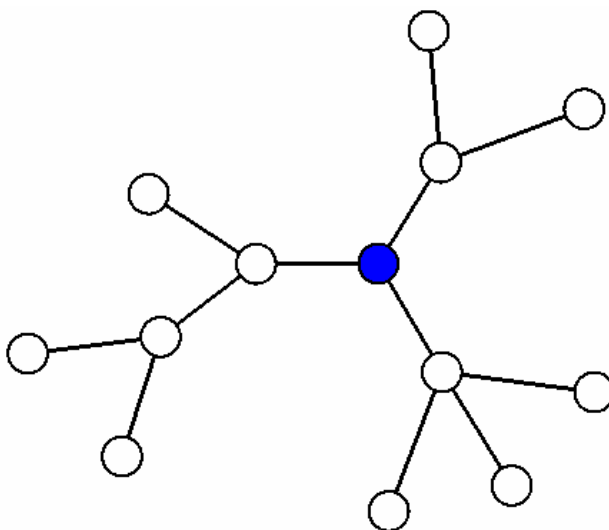


Obr. 2 Príklad Rooted tree v Total Commanderi

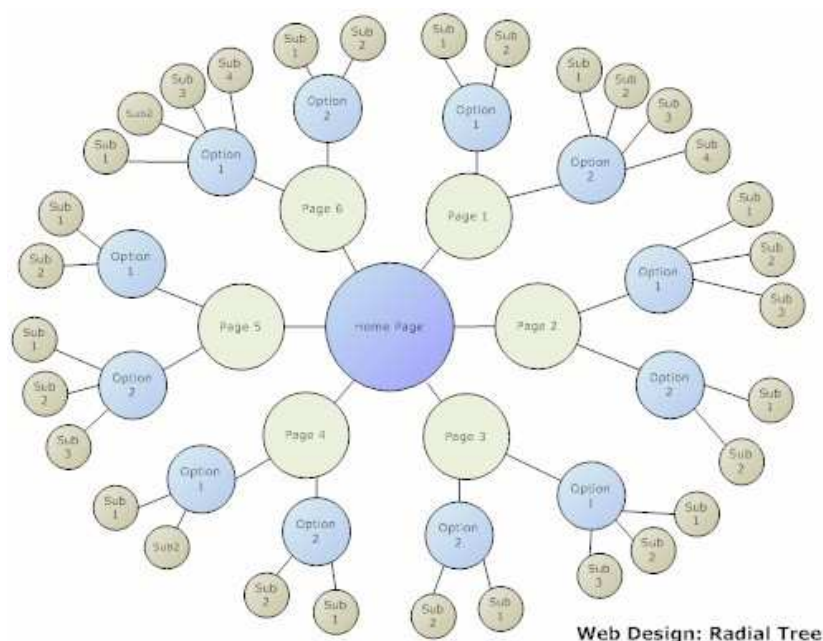
Radial tree

Vrchol dostáva ako priestor na vykreslenie seba a svojej vetvy priestor vymedzený dvoma polpriamkami vychádzajúcimi z jedného bodu a kružnicou so stredom v tomto bode, kruhový odsek. Vrchol sa vykresľuje na kružnicu so stredom v koreni. Synom predáva sústrednú kružnicu s polomerom zväčšeným o nejakú konštantu a kruhový odsek, vymedzený touto kružnicou a predchádzajúcim kruhovým odsekom, ktorý sa rozdelí pre jednotlivých synov. Jednotlivé sústredné kružnice predstavujú level grafu.

Radial tree oproti Rooted tree zaberá celý priestor.



Obr. 3 Schéma Radial tree

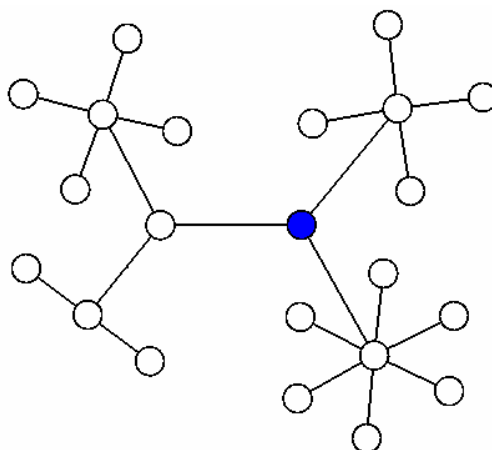


Obr. 4 Príklad Radial tree¹

Balloon tree

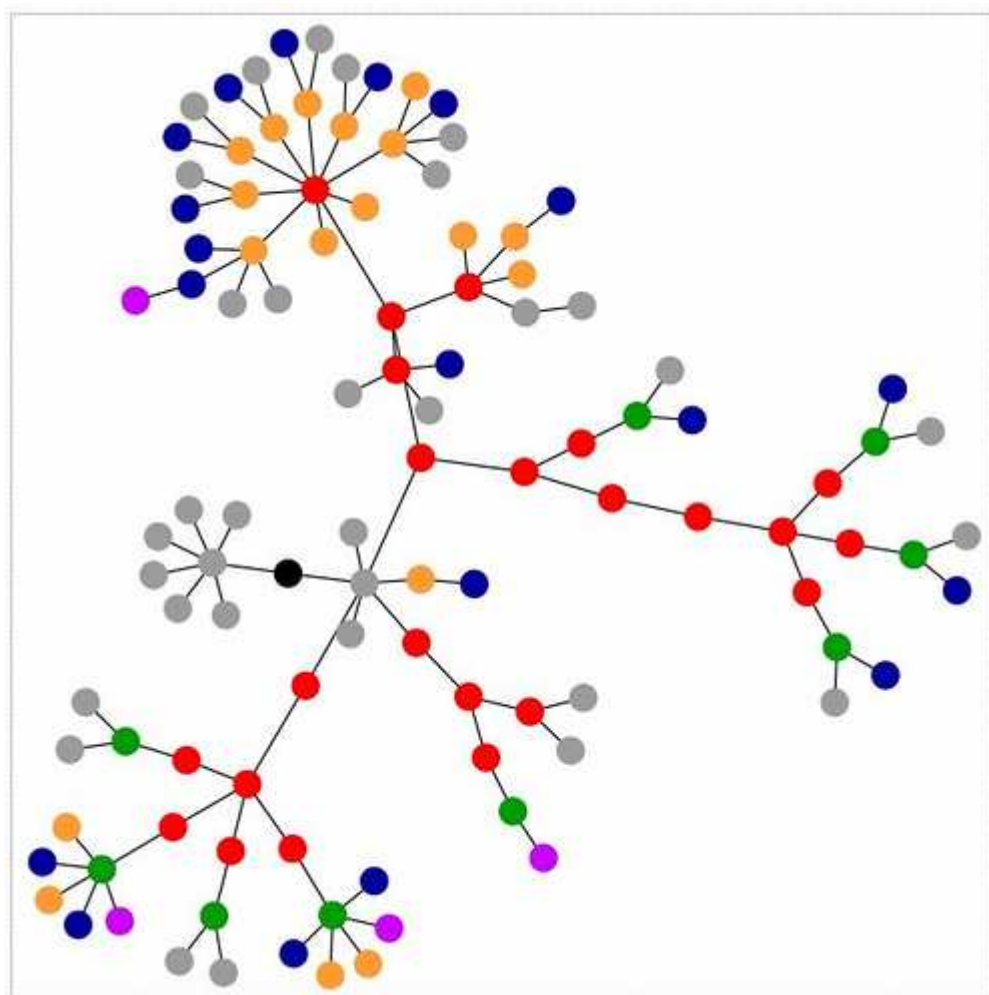
Vrchol pri Balloon tree oproti Radial tree vždy dostáva celý priestor a kružnicu so stredom v otcovi. Teda aj synom predáva kružnicu so stredom v ňom a uhol, ktorý má zvierat' ich spojnica so súradnicovou osou. Polomer kružnice sa pri predávaní zmenšuje, aby nedochádzalo k prekryvaniu hrán a vrcholov.

Balloon tree ešte viac využíva priestor ako Radial tree. Je to ale na úkor prehľadnosti, ktorá sa pri hlbších grafoch výrazne znižuje. Stráca sa aj informácia o leveli grafu.



Obr. 5 Schéma Balloon tree

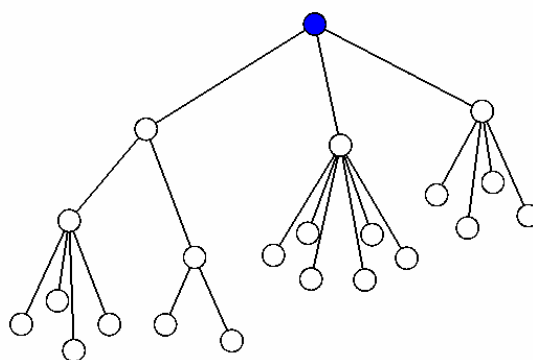
¹ http://www.smartdraw.com/examples/preview/index.aspx?example=Web_Design_-_Radial_Tree



Obr. 6 Príklad Balloon tree²

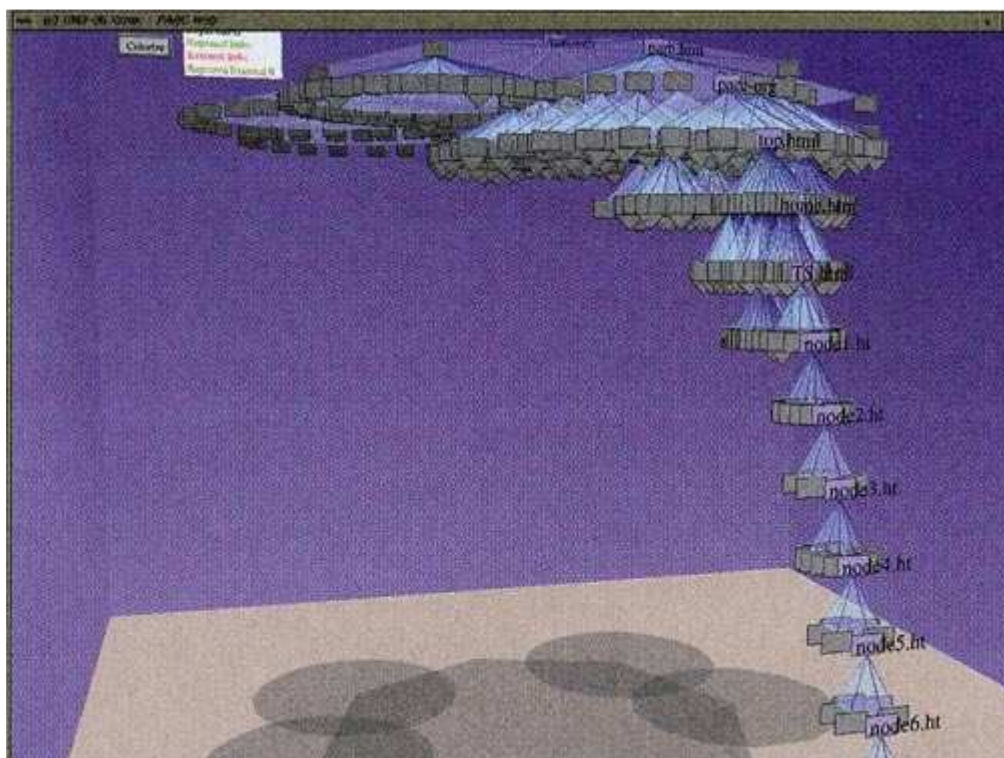
Cone tree

Cone tree sa zobrazuje do trojrozmerného priestoru. Je podobný s Balloon tree, kde sa každý ďalší level posúva do rovnobežnej roviny. Získava sa tak oproti Balloon tree prehľadnosť a informácia o leveli, ktorá je uložená v jednotlivých rovinách.



Obr. 7 Schéma Cone tree

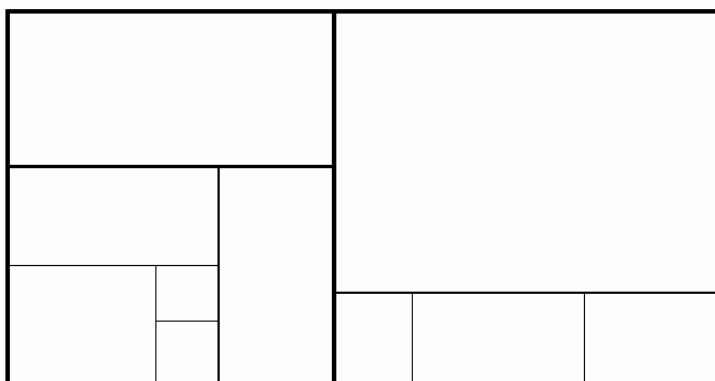
² <http://flickr.com/photos/ajc1/490892726/>



Obr. 8 Príklad Cone tree³

Treemap

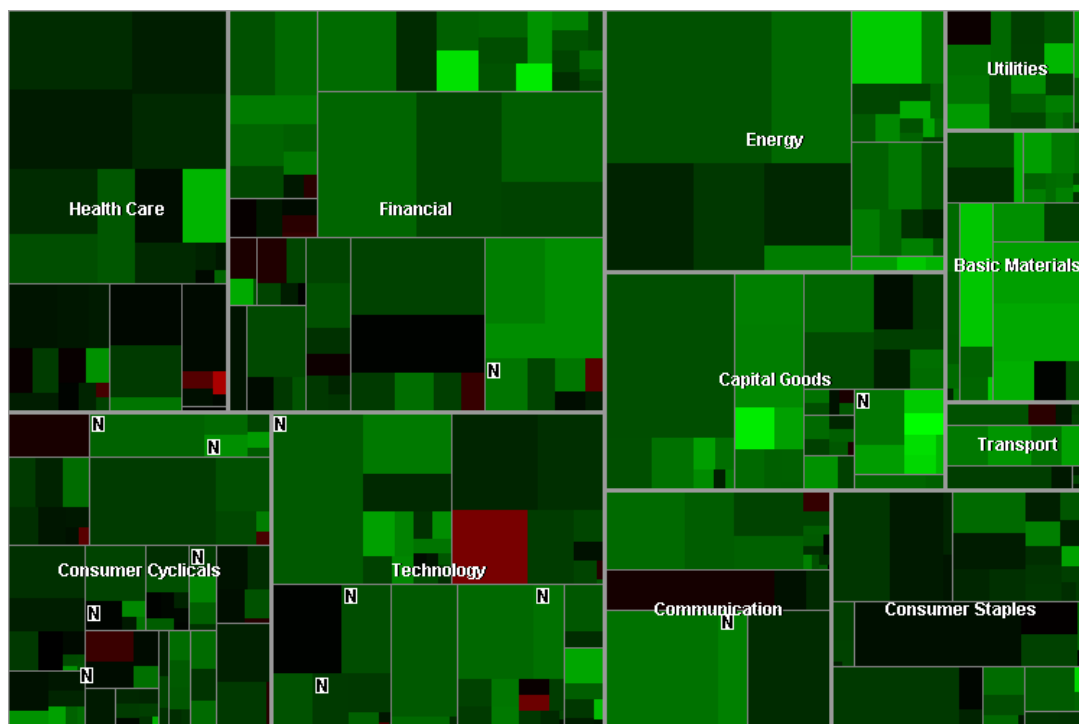
Vrchol dostáva ako priestor na vykreslenie pravouholník. Celý tento pravouholník predstavuje daný vrchol. Synovia predstavujú malé pravouholníky, ktoré vyskladávajú otca. Treemap zachytáva hlavne listy stromu. Ostatné vrcholy sú oproti listom nevýrazné. Preto sa aj stráca prehľad o leveloch v grafe.



Obr. 9 Schéma Treemap

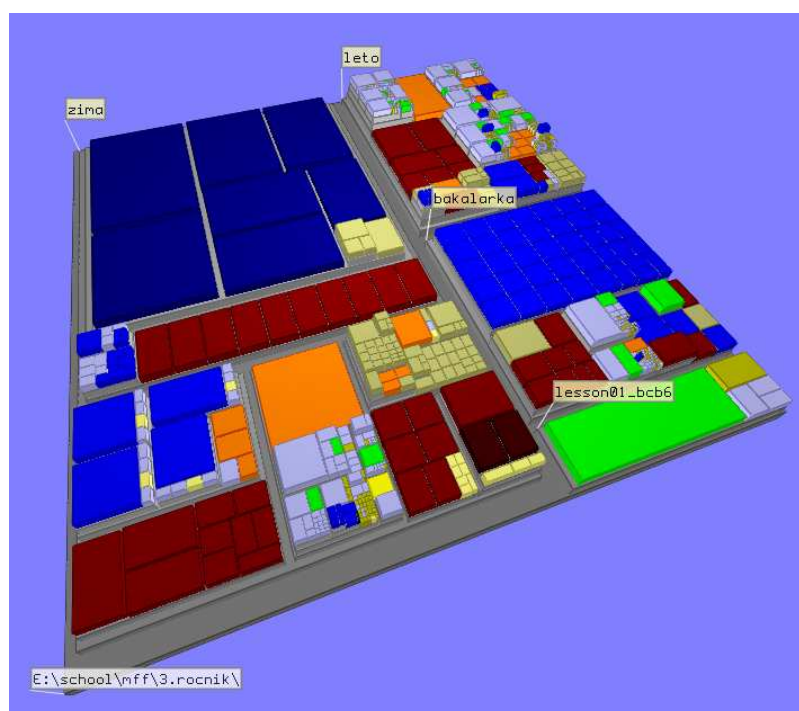
Túto metódu zobrazovania využíva napríklad stránka smartmoney.com na zobrazovanie stavu akcií na trhu.

³ <http://www.saul.unisi.it/parco/sigdoc/index.html>



Obr. 10 Príklad Treemap pri mapovaní trhu⁴

Na vizualizáciu súborového systému používa Treemap program StepTree. Využíva však 3D priestor na lepšie zobrazenie aj nižších vrcholov. Umožňuje pohyb v priestore na priblíženie vlastností, mapovanie farieb podľa prípon súborov a rôzne filtre.

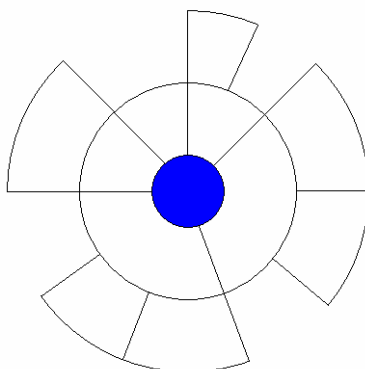


Obr. 11 Príklad Treemap v StepTree

⁴ <http://www.smartmoney.com/marketmap/> 14.6.2007 7:30

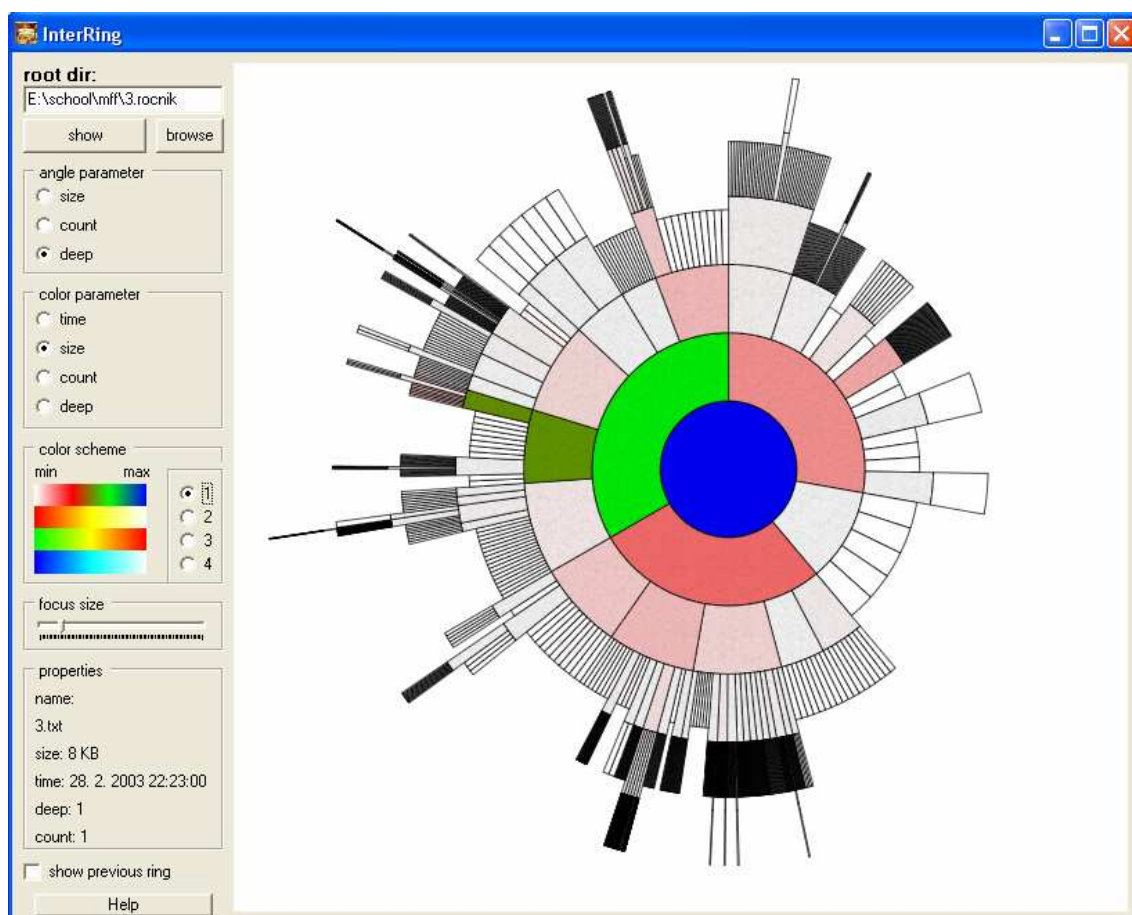
Interring

Interring je založený na Radial tree. Vrcholy sa zobrazujú ako oblasti ohraničené kruhovým výsekom a kružnicou, ktoré dostali od otca a kružnicou, ktorú predávajú synom.



Obr. 12 Schéma Interring

Na vizualizáciu súborového systému ho využíva tento projekt, ktorý je popísaný v sekcii 4.



Obr. 13 Príklad Interring v InterRing

2.3 Vizuálna informácia stromu

Každá vizualizácia stromu podáva určité informácie o štruktúre stromu. Pri prehľadných vizualizáciách je táto informácia čitateľnejšia. Okrem štruktúry stromu, môže vizualizácia zobrazovať aj doplnkové informácie. Tie väčšinou závisia od informácií uložených vo vrchoch. Ak nepočítame textovú informáciu, ktorú môžeme zobraziť do daného vrcholu a ktorá je schopná popísať všetky údaje o vrchole, môžeme na zobrazenie údajov využiť napríklad farbu jednotlivých vrcholov. Na správne ofarbenie vrcholu najprv potrebujeme zistiť maximálnu a minimálnu hodnotu zobrazovanej vlastnosti v strome. Dostávame interval, ktorý môžeme normalizovať. Potom už len stačí previesť číslo z intervalu do RGB podoby. Napr. RGB budeme pokladať za číslo z intervalu $\langle 0, 255 \cdot 255 \cdot 255 \rangle$. Podobne sa dá využiť aj veľkosť plochy ktorá sa prenáša pri vykresľovaní (šírka úsečky pri Rooted tree, uhol kruhového výseku pri Radial tree, veľkosť pravouholníka pri Treemap, ...). Ďalšie čo sa dá využiť je samotný tvar vrcholov, pri vizualizáciách, ktoré ho nemajú priamo zadany. Rôzne prípony súborov môžu reprezentovať rôzne tvary vrcholov napr. pri Rooted tree (poznáme to ako napr. ikonky v Total Commanderi).

2.4 Porovnanie vizualizácií

Pri vizualizácii veľkého počtu dát, ako napríklad súborový systém, dochádza k pomerne veľkej neprehľadnosti. Hlavne pri kruhových reprezentáciách ako Radial tree, Balloon tree, Interring, u ktorých sú listy pomerne husto vedľa seba a zle sa rozpoznávajú. Najlepšie tento problém rieši Rooted tree, čo je ale na úkor veľmi veľkej priestorovej náročnosti. Pomerne dobre vzhľadom na veľkosť využitého priestoru tento problém zobrazuje Treemap. Uňho však zanikajú údaje o vnútorných uzloch. Listy nezanikajú ani u Cone tree, čo je spôsobené prechodom do 3D.

Mapovanie vizuálnych informácií na farbu a veľkosť vrcholov podporujú všetky spôsoby zobrazovania. Rooted tree, Radial tree, Cone tree, Balloon tree umožňujú aj mapovanie na tvar vrcholov.

3 Súborový systém

3.1 Význam súborového systému a praktické problémy

Súborový systém je dátová štruktúra, v ktorej sa uchovávajú dáta na diskových jednotkách. Je to stromová štruktúra, ktorá nesie informácie o uzloch, ktoré predstavujú adresáre a listoch, ktoré predstavujú súbory. Slúži teda na prístup k údajom a ich správu.

Disk slúži na archiváciu dát a teda jedna z primárnych funkcií súborového systému je organizácia týchto dát. Bežný užívateľ ho využíva buď na služobné, alebo súkromné účely. Pri služobných pracuje hlavne s textovými dokumentmi, prezentáciami, štatistickými údajmi, ... V súkromí hlavne pri zábave a oddychu, teda uschovávanie filmov, hudby, obrázkov, hier, ... Pri správe týchto dát vznikajú nepotrebné adresáre a súbory. Či už duplikáty pri premiestňovaní dát, alebo prázdne adresáre pri mazaní dát a ponechaní adresárovej štruktúry.

Do súborového systému zasahujú aj rôzne programy, ktoré vytvárajú nové adresáre a súbory ako napríklad takzvané „dočasné“ alebo konfiguračné súbory. Po odinštalovaní často ostávajú tieto adresáre a súbory na disku bez toho, aby boli zmazané, prípadne sa pri odinštalovaní program zmaže, ale nechá zálohy konfiguračných súborov pre prípadnú opätovnú inštaláciu. Takto vznikajú nepotrebné adresáre na disku. Väčšinou majú malú veľkosť a veľkú hĺbku.

3.2 Vlastnosti

Na rozpoznávanie adresárov a súborov nám slúžia ich jednotlivé vlastnosti. Samozrejme najdôležitejší je názov, ktorý dáva prvotnú informáciu o adresári či súbore. Pri názve súbore je aj prípona, ktorá určuje typ súboru. Pre jednoduchosť môžeme adresár považovať za osobitný typ súboru.

Pre zálohovanie má dôležitý význam dátum vzniku, dátum poslednej zmeny a dátum posledného prístupu.

Pri súboroch zisťujeme aj veľkosť - koľko bytov zaberajú na disku. Adresáre reálne nemajú veľkosť, ale pre praktické potreby a pre efektívne využitie vizualizácie budeme veľkosť adresára brať ako súčet veľkostí všetkých súborov, ktoré obsahuje.

Ďalšie vlastnosti vychádzajú zo stromovej štruktúry súborového systému. Početnosť adresára, súboru môžeme zdefinovať ako počet súborov, ktoré sa nachádzajú v jeho vetve. Teda pre súbor je to jedna.

Hĺbkou adresára, súboru môžeme zdefinovať ako hĺbkou vetvy pre daný vrchol. Teda pre súbor to je jedna a pre adresár je to vzdialenosť k najvzdialenejšiemu súboru, alebo adresáru, ktorý je v jeho vetve.

Ďalšie známe vlastnosti, ktoré majú adresáre a súbory sú atribúty, práva. Tie sú zapísané v dvojkovej sústave podľa toho či má užívateľ pre daný súbor, či adresár dané právo. V tejto práci sa im nevenujem, predstavujú ale zaujímavú možnosť rozšírenia prezentovanej vizualizácie.

3.3 Účel vizualizácie

Vizualizácia súborového systému by teda okrem samotnej správy mala slúžiť na lepšie rozpoznávanie týchto „nepotrebných“ adresárov a súborov.

Napríklad internetové prehliadače si ukladajú stránky na disk, na zrýchlenie ich opätovného načítania. Adresáre, ktoré obsahujú tieto zálohy majú preto väčšinou veľkú početnosť a súbory, ktoré obsahujú sú pomerne malé. Na objavenie takýchto adresárov sa preto hodí vizualizácia adresárov podľa početnosti.

Na hľadanie adresárov po odinštalovaných programoch, ktoré majú väčšinou veľkú hĺbkou s pomerne malým obsahom dát, sa dá využiť vizualizácia podľa hĺbky, pri ktorej sa zvýraznia adresáre s veľkou hĺbkou.

Pri hľadaní duplicitných adresárov sa využívajú všetky vlastnosti adresárov a súborov, vzhľadom nato, že by ich mali mať oba adresáre rovnaké.

4 Popis aplikácie

Aplikácia InterRing zobrazuje súborový systém pomocou metódy Interring. Bola písaná v C++ v prostredí Borland C++ Builder 6. Na zobrazovanie využíva OpenGL. Umožňuje mapovať rôzne vlastnosti adresárov a súborov na jednotlivé parametre vizualizácie. A to:

- Šírka kruhového výseku
 - Veľkosť adresára / súboru
 - Hĺbka podstromu v danom adresári / súbore
 - Početnosť adresára / súboru
- Farba adresára / súboru
 - Dátum vytvorenia adresára / súboru
 - Veľkosť adresára / súboru
 - Hĺbka podstromu v danom adresári / súbore
 - Početnosť adresára / súboru

Aplikácia umožňuje:

- Interaktívne meniť farebnú škálu
- Otvárať a zatvárať adresáre
- Zvýrazňovať adresáre / súbory
- Interaktívne meniť uhol zvýraznenia
- Interaktívne meniť koreňový adresár
- Možnosť náhľadu predchádzajúceho zobrazenia
- Prehľad vlastností adresára / súboru

Všetky vizuálne zmeny prebiehajú v animáciách. Všetky prechody prebiehajú plynule, čo minimalizuje riziko, že užívateľ sa v zobrazovaní „stratí“.

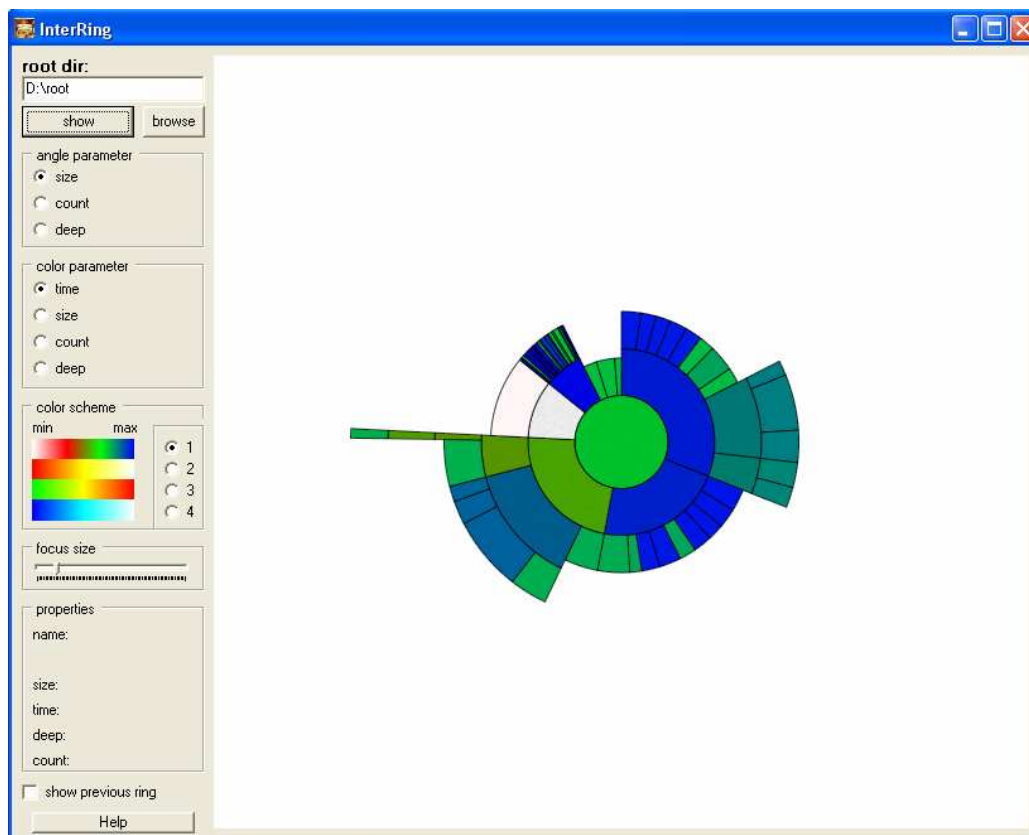
Aplikácia je určená na zobrazovanie súborového systému, neumožňuje jeho editáciu. Rieši niektoré problémy pri hľadaní určitých adresárov za pomoci rôzneho mapovania vlastností.

Po spustení aplikácie sa otvorí základná obrazovka bez načítanej adresárovej štruktúry.

Užívateľ si zvolí pomocou dialógového okna koreňový adresár a nechá ho načítať. Aplikácia prechádza všetky podadresáre pre daný koreň a načítava ich vlastnosti. Pre početné adresáre môže byť táto operácia pomerne zdĺhavá. Cieľom bolo

vizualizovať súborový systém, preto sa na efektívnosť algoritmov nekládol až taký dôraz.

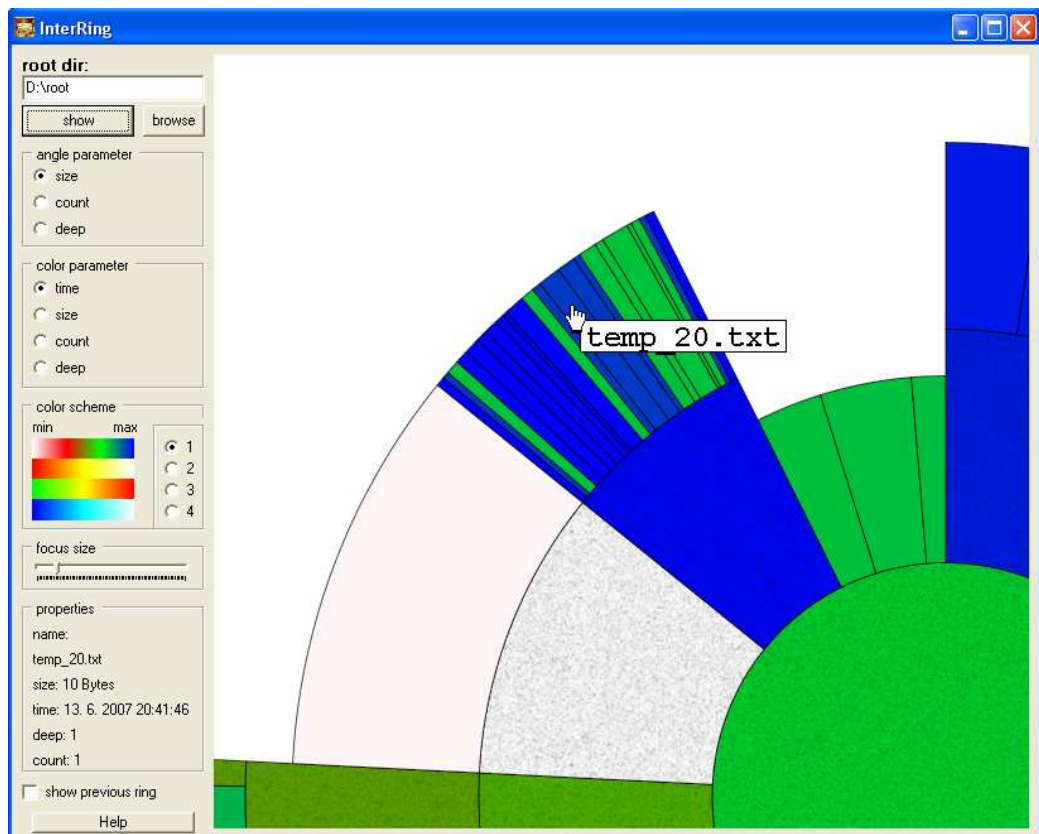
Po načítaní adresárovej štruktúry sa v animácii zobrazí vizualizácia. Adresáre sú od súborov odlíšené pomocou textúry.



Obr. 14 Načítaný adresárový strom

Užívateľ sa môže za pomoci myši v zobrazení pohybovať. Pomocou kolieska na myši sa dá zobrazenie priblížiť, alebo oddialiť.

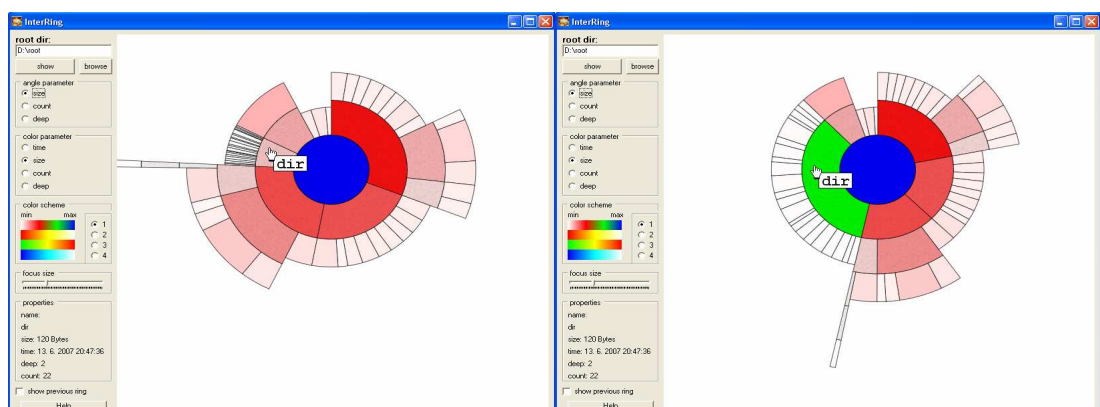
Pomocou držania pravého tlačítka sa môže pohybovať v zobrazení.



Obr. 15 Pohyb v zobrazení

Pri prejdenní kurzorom myši ponad adresár / súbor sa zobrazí jeho názov a v tabuľke v ľavo dole aj jeho ďalšie vlastnosti ako veľkosť, dátum vytvorenia, početnosť, hĺbka.

Po kliknutí na adresár / súbor sa tento v animácii zvýrazní a to farebne aj zväčšením kruhového výseku. Veľkosť zvýraznenia sa dá interaktívne regulovať pomocou trackbaru focus size. Ak nás zaujíma špeciálny adresár / súbor, môžeme si ho takto zvýrazniť, teda v porovnaní s ostatnými adresármi / súborami zväčšíme jeho uhol. Zároveň, ale nestrácame informácie o ostatných adresároch / súboroch. Táto technika sa nazýva focus + context.

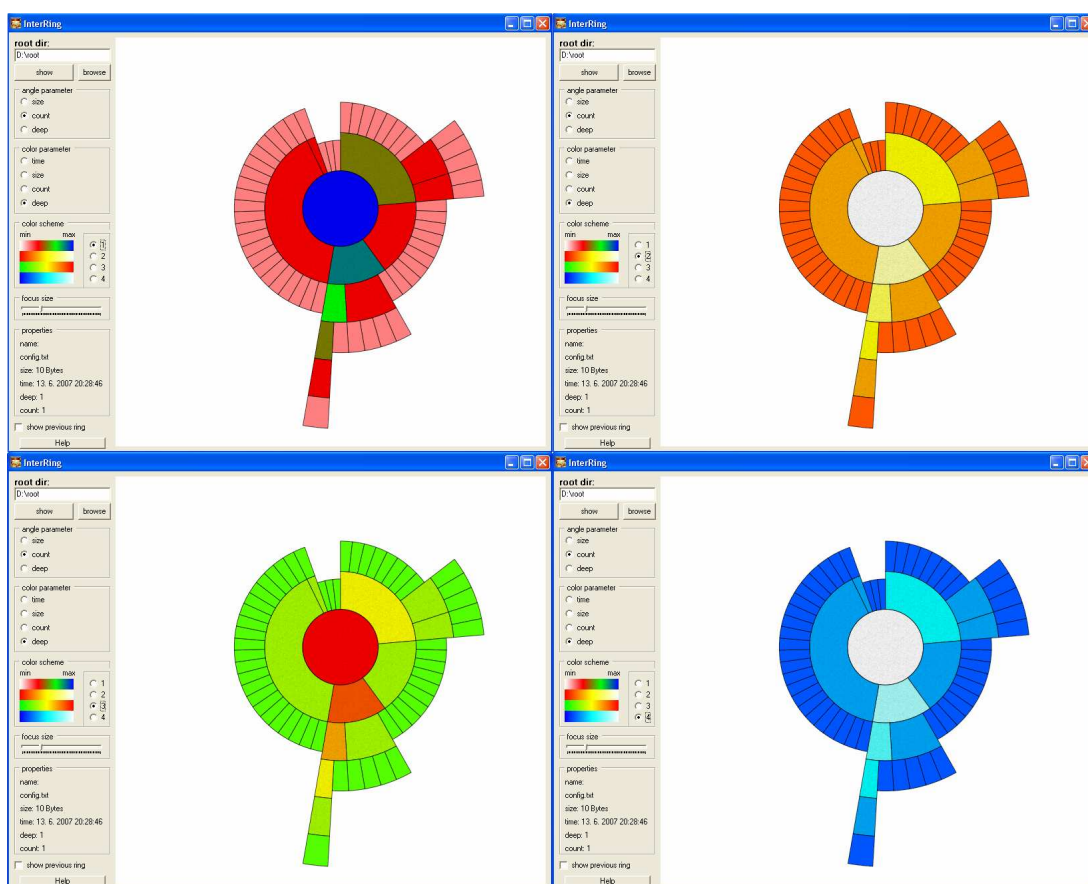


Obr. 16 Zvýraznenie adresára / súboru

Pri dvojkliku na adresár sa tento v animácii zatvorí / otvorí.

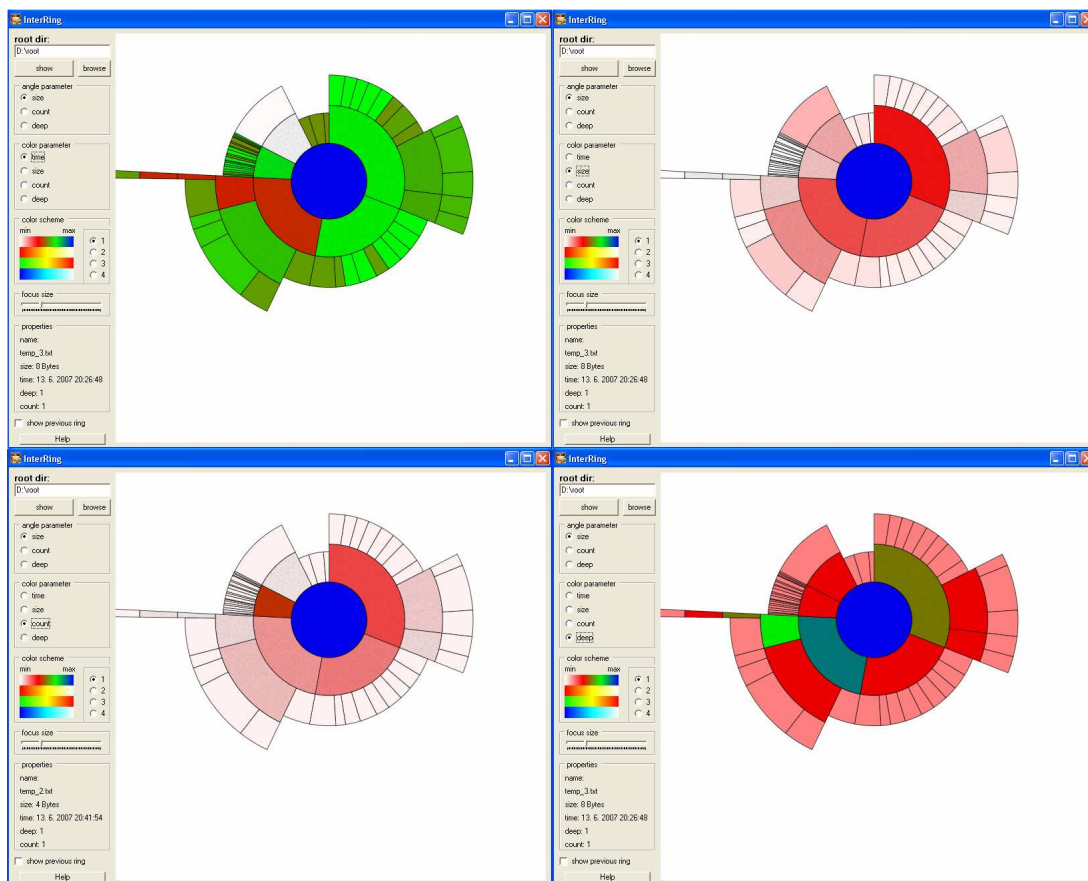
Stredným kliknutím na adresár sa tento adresár stane koreňovým adresárom a začne sa jeho načítavanie. Pri strednom kliku na koreňový adresár sa koreňovým adresárom stane otec terajšieho adresára.

V ľavom panely sú selekt boxy na určovanie vlastností, ktoré sa mapujú na jednotlivé grafické prvky vizualizácie. Ako aj voľba farebnej schémy, ktorá sa má použiť. Pri zmene farebnej schémy sa zobrazenie automaticky prekreslí s novou schémou.



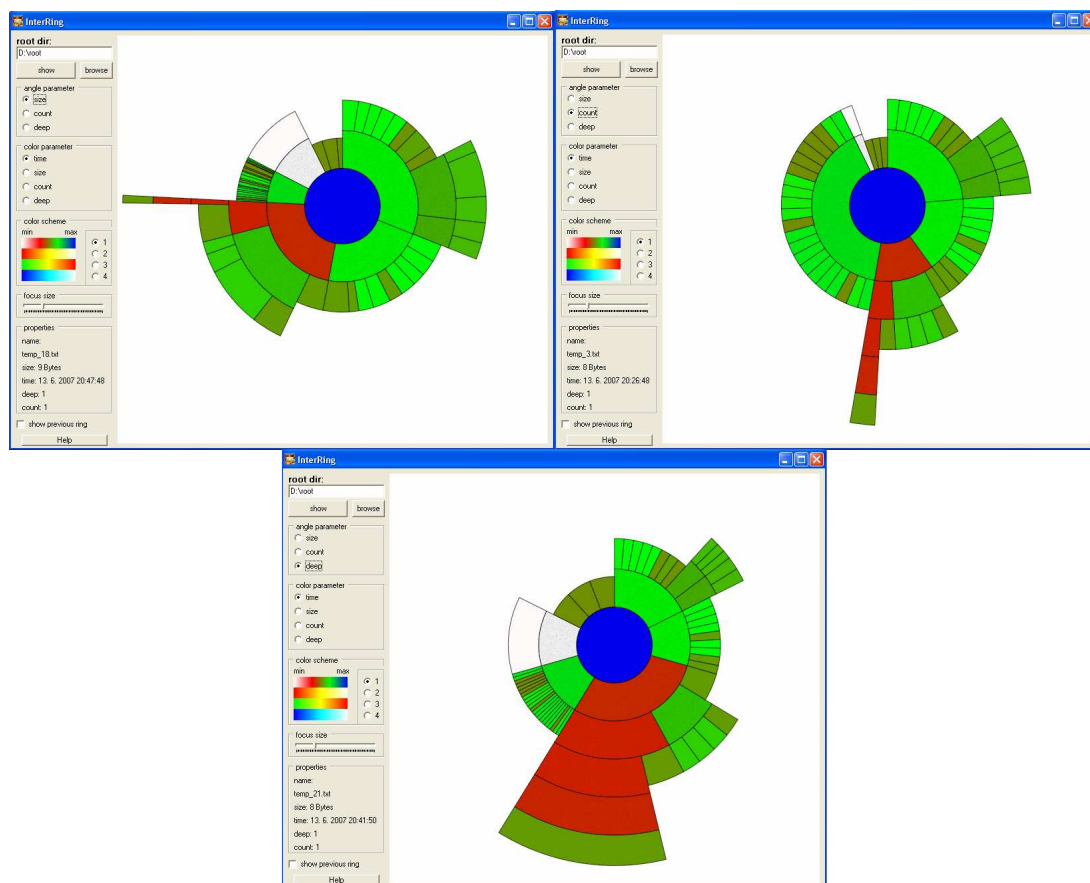
Obr. 17 Zmena farebnej schémy

Pri zmene mapovania farby sa zobrazenie automaticky prekreslí za použitia nového mapovania.



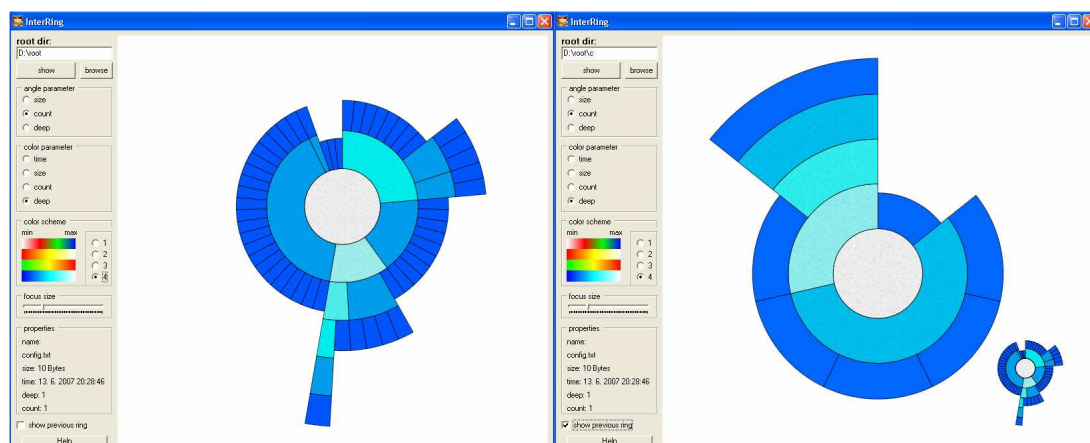
Obr. 18 Zmena mapovania farby

Pri zmene mapovania šírky kruhového výseku sa v animácii zmení zobrazenie.



Obr. 19 Zmena mapovania šírky

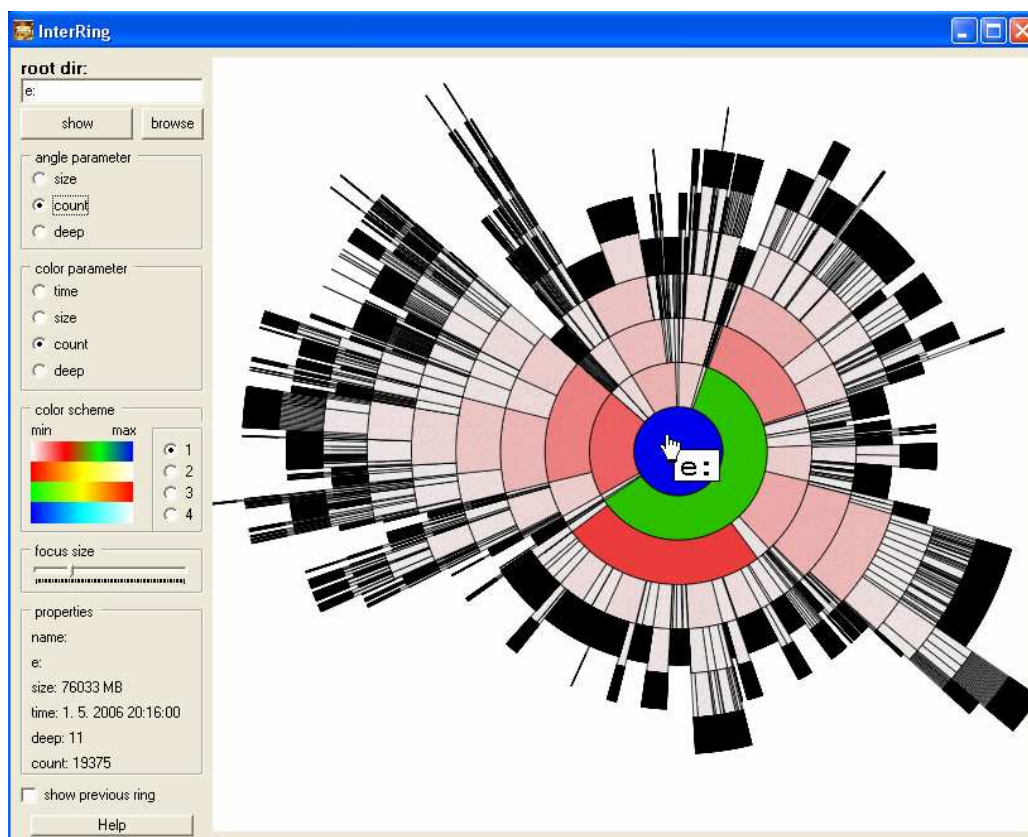
Na spodku ľavého panelu je checkbox na zobrazenie predchádzajúceho adresárového stromu. Je to ďalší z prístupov focus+context. Umožňuje vidieť užívateľovi adresárový strom, z ktorého vošiel do terajšieho.



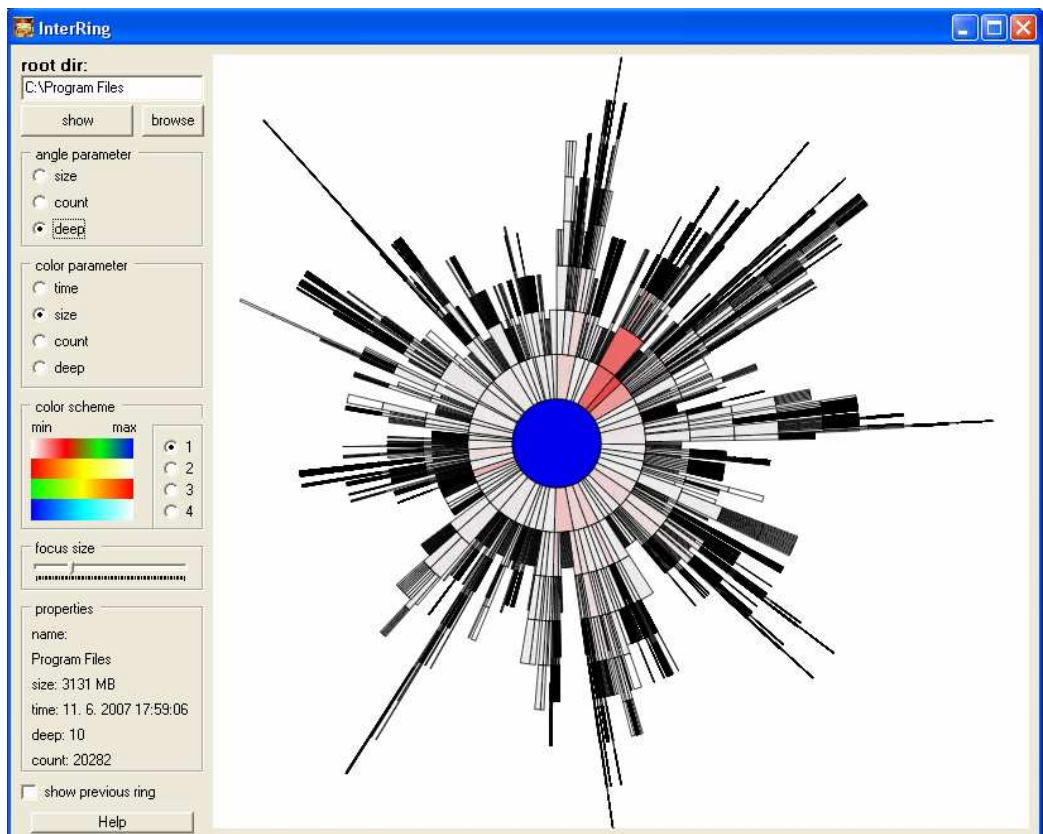
Obr. 20 Zobrazenie predchádzajúceho stromu

5 Príklady

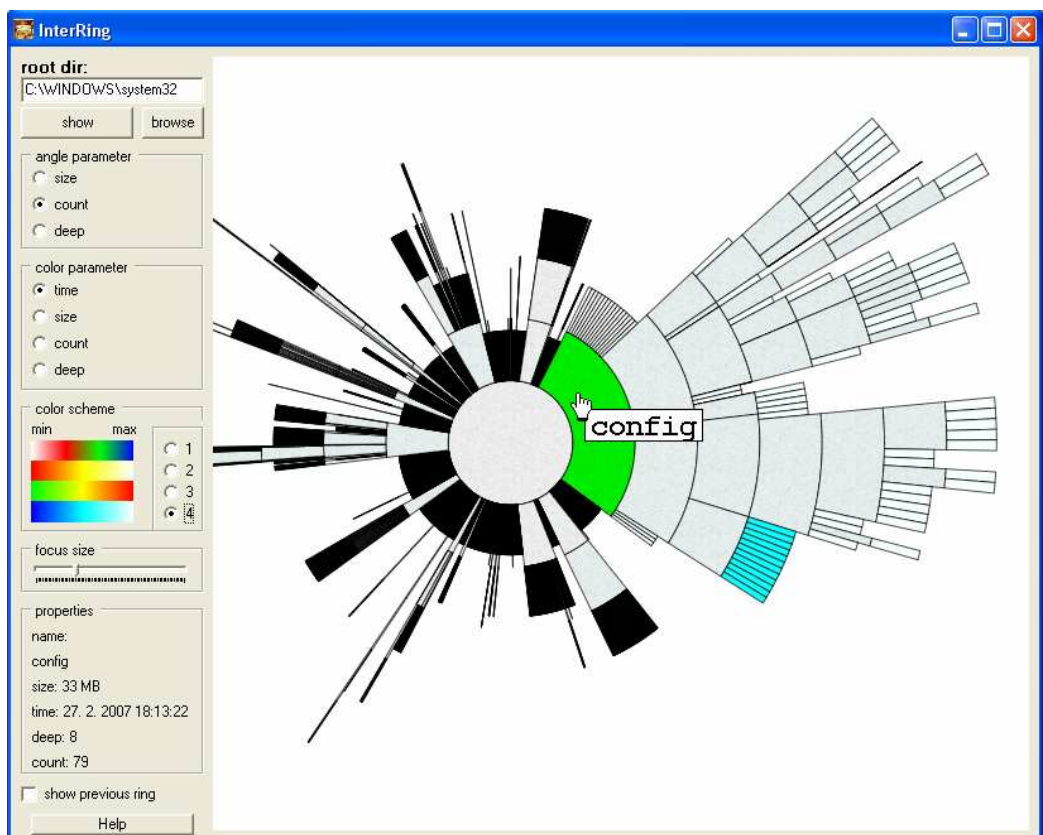
5.1 Príklady z vizualizácie pomocou aplikácie InterRing



Obr. 21 Príklad 1



Obr. 22 Příklad 2



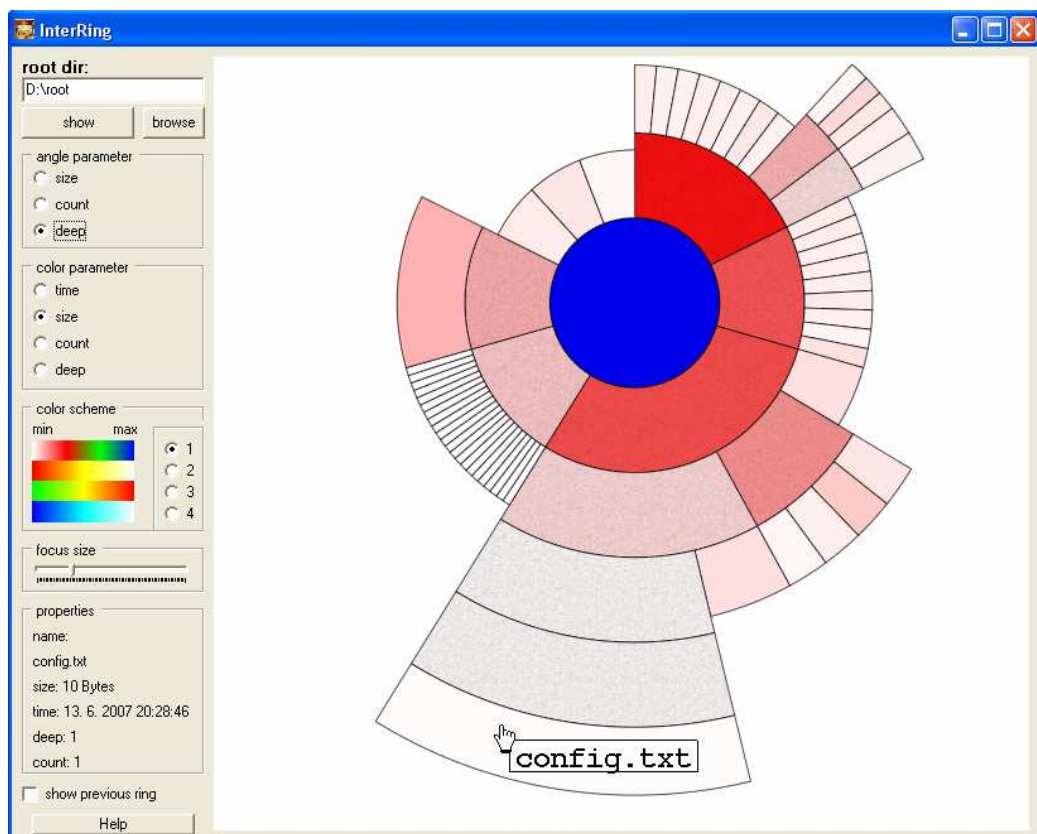
Obr. 23 Příklad 3

5.2 Riešenie vyhľadávacích problémov pomocou aplikácie InterRing

Hľadanie hlbokého adresára s malou veľkosťou

Daný prípad nastáva napríklad pri odinštalovaní programu, ktorý necháva zálohy konfiguračných súborov.

Potrebuje zvýrazniť hlboké adresáre. Na to využijeme mapovanie podľa hĺbky na šírku kruhového výseku. Na zistenie veľkosti použijeme mapovanie veľkosti na farbu so škálou BČZM. Hľadaný adresár bude vo vyšších leveloch so širším kruhovým výsekom a bude mať bielu farbu.

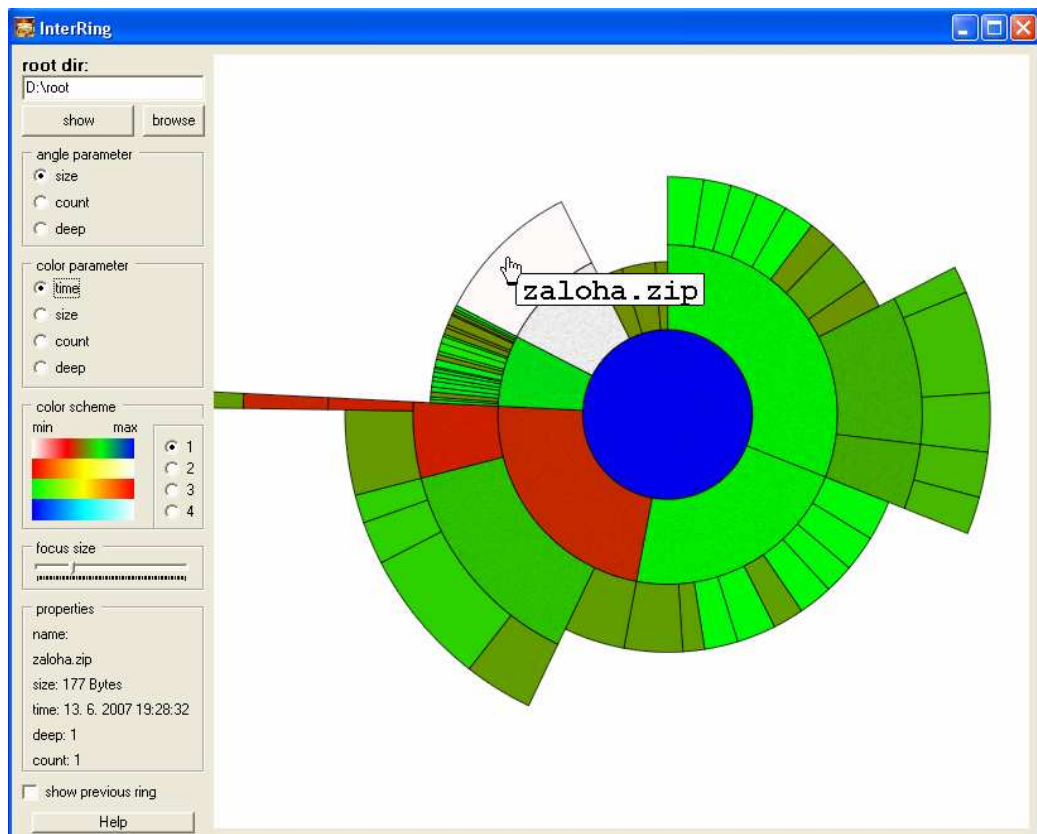


Obr. 24 Riešenie 1

Hľadanie starého veľkého samotného súboru

Daný prípad nastáva napríklad pri hľadaní zálohy v komprimovanej podobe.

Zvýrazníme veľké súbory pomocou mapovania veľkosti na šírku kruhového výseku. Farebnú škálu zvolíme BČZM. Na mapovanie farby použijeme čas. Hľadaný súbor bude mať širší kruhový výsek, bude biely a bude zaberáť celú šírku adresára.

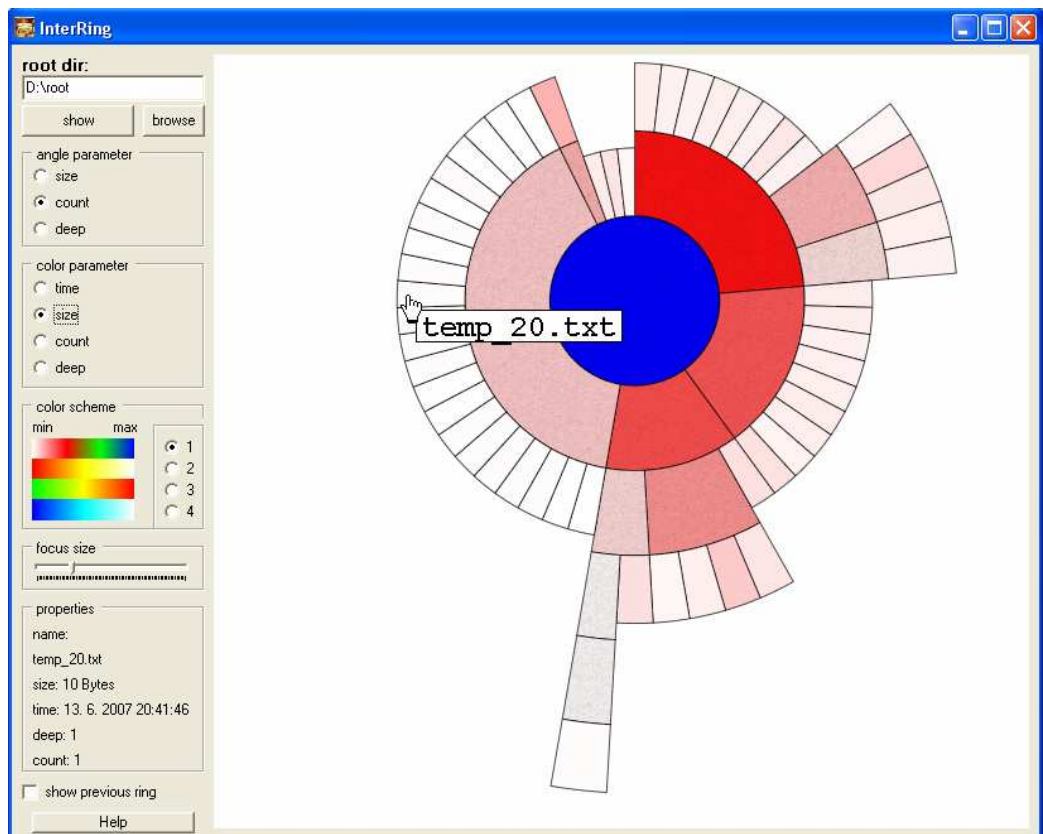


Obr. 25 Riešenie 2

Hľadanie početného adresára s malými súbormi

Daný prípad nastáva napríklad pri hľadaní „dočasných“ súborov vytvorených prehliadačom.

Na zvýraznenie početnosti adresára mapujeme početnosť na šírku kruhového výseku. Na farbu so schémou BČZM mapujeme veľkosť. Hľadaný adresár bude mať širší kruhový výsek a jeho súbory budú biele.



Obr. 26 Riešenie 3

6 Záver

Primárnym cieľom mojej práce bolo implementovať zobrazovanie súborového systému metódou interrering. Tento cieľ som splnil, čo dokazuje priložená aplikácia a obrázky z nej. Vytvorená aplikácia je graficky príjemná a jednoduchá na ovládanie. Umožňuje rôzne spôsoby zobrazovania a využíva animácie pri jednotlivých prechodoch zobrazovaní. Aplikácia implementuje aj zaužívané postupy pre focus+context vizualizáciu a redukuje riziko change blindness a straty orientácie vo vizualizácii pomocou plynulej animácie.

Aplikácia by sa dala rozšíriť o možnosť editácie súborového systému čím by sa stala omnoho viac využiteľná aj v bežnej praxi. Na zobrazovanie by sa dali doprogramovať rôzne filtre na súbory podľa prípon, veľkostí a iných parametrov, čím by sa z nej stal veľmi dobrý vyhľadávací nástroj.

Druhý cieľ bol popísať rôzne spôsoby vizualizácie stromu. Porovnal som jednotlivé vizualizácie a na príklade InterRing som ukázal riešenia rôznych problémov hľadania adresárov v súborovej štruktúre.

Táto práca mi dala veľa znalostí z programovania v C++ s využitím OpenGL a prehľad vo formách vizualizácie stromových štruktúr.

Zoznam použitej literatúry

- [1] Jing Yang, Matthew O. Ward and Elke A. Rundensteiner. InterRing: An Interactive Tool for Visually Navigating and Manipulating Hierarchical Structures. <http://davis.wpi.edu/~xmdv/docs/ring.pdf> (15-06-2007)
- [2] Alfred Kobsa. User Experiments with Tree Visualization Systems. <http://www.ics.uci.edu/~kobsa/papers/2004-InfoVis-kobsa.pdf> (15-06-2007)
- [3] Daisuke MIZUKOSHI, Yukio HORI, Tomonori GOTOH. Extension models of Cone Tree Visualizations to Large scale Knowledge base with Semantic Relations. http://wscg.zcu.cz/wscg2006/Papers_2006/Poster/C89-full.pdf (15-06-2007)
- [4] Mark Bruls, Kees Huizing, and Jarke J. vanWijk. Squarified Treemaps. <http://www.win.tue.nl/~vanwijk/stm.pdf> (15-06-2007)
- [5] B. Shneiderman and M. Wattenberg. Ordered treemap layouts. Proc. of Information. Visualization 2001, p. 73-78, 2001.
- [6] J. Stasko, R. Catrambone, M. Guzdial, and K. McDonald. An evaluation of space-filling information visualizations for depicting hierarchical structures. Int. J. Human-Computer Studies, Vol. 53, p. 663-694, 2000.

Obsah CD

- Aplikácia InterRing
- Zdrojové kódy aplikácie