

VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
PROGRAMŲ SISTEMŲ KATEDRA

Tiesioginis operacijų apdorojimas

Online transaction processing

Kursinis darbas

Atliko: 3 kurso 3 grupės studentas
Dainius Jocas (parašas)

Darbo vadovas: Asistentas Stasys Peldžius (parašas)

VILNIUS - 2011

Turinys

IVADAS	3
1. Tiesioginio transakcijų apdorojimo sistemos	4
1.1. OLTP sistemų savybės	4
1.2. Realaus laiko veika OLTP sistemose	6
2. OLTP palyginimas su OLAP sistemomis	6
2.1. OLAP sistemos	7
2.2. Kas yra duomenų sandėliai?	7
2.3. OLTP ir duomenų sandėlių skirtumai	8
2.4. Skirtumų apibendrinimas	9
3. OLTP sistemų tendencijos	11
4. Senesnės technologijos internetinių programų duomenų saugojimui kliento kompiuteryje	12
4.1. Slapukai	12
4.2. JavaScript ir failai vartotojo kompiuteryje	14
4.3. Papildiniais grįstos technologijos	15
4.3.1. „Flash“	15
4.3.2. Programėlės	16
4.3.3. „Google Gears“	17
4.3.4. Papildinių naudojimo ypatybės	17
4.4. Specifinės naršyklių savybės	18
5. HTML5 standarto naujovės susijusios su tiesioginiu transakcijų apdorojimu	20
5.1. Programų podėlis	21
5.2. Vidinė atmintinė	21
5.3. Saityno SQL duomenų bazė	22
5.4. Indeksuota duomenų bazė	24
5.5. Failų API	25

5.6. Duomenų kiekiai	26
5.7. Naujų technologijų palaikymas naršyklėse	26
REZULTATAI IR IŠVADOS	27
LITERATŪRA	28
SAVOKŲ APIBRĖŽIMAI	29

IVADAS

Šiandien reta veikla (mokslas, verslas, pramonės ir t.t.) gali išsiversti ir pilnai funkcionuoti be vienkokių ar kitokių informacinių sistemų pagalbos. Viena iš informacinių sistemų rūšių yra tiesioginio transakcijų apdorojimo sistemos. Tokių sistemų svarba, o kartu ir paklausa, nuolat auga, todėl jų kuriama vis daugiau ir vis sudėtingesnių. Šio darbo tikslas yra apžvelgti tiesioginio transakcijų apdorojimo sistemas, jų skirtumus su panašaus pobūdžio informacinėmis sistemomis bei pažvelgti į tokių sistemų kūrimo tendencijas.

Informacinių sistemų rinka yra labai aktyvi. Joje yra nemažai didelių žaidėjų, kurie investuoja milžiniškas pinigų sumas tam, kad būtų sukurtos naudotojų poreikius vis geriau tenkinančios sistemos. Šios sistemos yra iš dalies tarpusavyje panašios, tačiau kai kurie jų funkcionalumo aspektai gali gana esmingai skirtis. Todėl yra aktualu žinoti pagrindinius tokių sistemų skirtumus tam, kad būtų naudojama geriausiai poreikius tenkinanti informacinė sistema.

Šiuolaikinis verslas turi būti dinamiškas ir reaguoti į rinkos pokyčius tam, kad būtų galima konkuruoti bei kurti vis geresnę produkciją. Tokiam verslui jau dabar svarbu žinoti kaip atrodys ateities informacinės sistemos. Taip yra, nes informacinės technologijos sensta, jas keičia naujos, kurių atsiranda vis daugiau ir vis įvairesnių. Žinoti svarbiausias su informacinėmis sistemomis susijusias naujienas yra aktualu, nes šiandien priimti sprendimai įtakos verslo veiklą rytoj. Dėl to informacinių sistemų technologijų ir jų naudojimo formų tendencijų apžvalga yra nuolat aktuali tema.

Vienas aktualiausių tiesioginio transakcijų apdorojimo sistemų ypatumų - realaus laiko darbas interneto naršyklėse. Tokių sistemų svarba yra didelė, nes tobulėjant internetinėms technologijoms, naršyklėse veikiančių programų galimybės ir funkcionalumas sparčiai artėja prie į galutinio kliento kompiuterį įrašytų programų. Kad ir kokios geros šiandieninės internetinės programos bebūtų, internetinių technologijų kūrėjai nesėdi sudėję rankų. Jie dirba tam, kad internetinės programos būtų dar funkcionalesnės, patogesnės ir stengiasi padaryti jų kūrimą lengvesniu. Informacinių technologijų indust-

rija nestovi vietoje - ji siūlo vis geresnius sprendimus. Todėl mūsų pareiga yra išanalizuoti informacinių sistemų vystymosi tendencijas.

Šiuo darbu siekiama susipažinti su tiesioginio transakcijų apdorojimo sistemomis. Darbo metu atliktos tiesioginio transakcijų apdorojimo sistemų analizės rezultatus bus galima panaudoti kuriant naujas informacines sistemas - sprendžiant kokios funkcijos turėtų būti sistemoje bei kokias technologinius įrankius reikėtų rinktis.

1 Tiesioginio transakcijų apdorojimo sistemos

Tiesioginio transakcijų apdorojimo (toliau OLTP¹) (angl. Online Transactional Processing) sistemos - sistemos, kurios palaiko operacinio lygio veiksmus t.y. registruoja kasdienes rutinines verslo operacijas, būtinas verslo sistemai funkcionuoti, o jų esminis bruožas yra realaus laiko² veikla.

Pagrindinės OLTP sistemos dalys yra:

- Grafinė vartotojo sąsaja;
- Duomenų bazė;
- Ataskaitų generavimo įrankis.

1.1 OLTP sistemų savybės

OLTP sistemos projektuojamos tam, kad sugebėtų efektyviai apdoroti didelius transakcijų kiekius - realiu laiku apdorotų verslo įvykius (pvz. užregistruotų pirkimus) ir nuolat dirbtų su naujausiais duomenimis. Trumpai apžvelgsiu pagrindines OLTP sistemos savybes.

¹OLTP - trumpinys angl. Online Transaction Processing - tiesioginis transakcijų apdorojimas.

²Realaus laiko arba tikralaikis(-ė) (angl. real time) - vykstantis kompiuteryje ir sąveikaujantis su tuo pat metu vykstančiais procesais kompiuterio išorėje.[DGJ08]

1 lentelė: OLTP sistemų savybės[UPI07]

Savybė	Paaškinimas
Greitas atsako laikas	Visos OLTP sistemos yra kuriamos tam, kad jos būtų interaktyvios bei su jomis būtų galima dirbti tada, kada reikia. OLTP sistemoms reikia trumpų atsako laikų, kad sistemos naudotojai išliktų produktyvūs.
Smulkios transakcijos	OLTP sistemos paprastai dirba tik su tam tikrais nedideliais duomenų kiekiais; duomenų apdorojimas dažniausiai yra paprastas, o sudėtingi DB lentelių sujungimai (angl. join) yra naudojami palyginti retai.
Duomenų palaikymo (angl. maintenance) operacijos	Nėra neįprasta turėti ataskaitų rengimo ir duomenų atnaujinimo programas, kurios turi dirbti arba reguliariai, arba pagal poreikį. Tos programos, kurios veikia foniniu (angl. Background) režimu, kol kiti sistemos klientai normaliai dirba su sistema, gali bandyti dirbti su dideliais kiekiais duomenų, kurie yra toje pačioje DB.
Daugelio naudotojų sistema	OLTP sistemos gali turėti labai daug naudotojų, kuriems gali reikėti tuo pačiu metu dirbti su tais pačiais duomenimis.
Intensyvus vienalaikiškumas	Dėl to, kad OLTP sistemos gali turėti daug klientų; joms reikia apdoroti didelius transakcijų kiekius, kuriuos reikia apdoroti per trumpą laiką, intensyvaus vienalaikiškumo palaikymas yra labai svarbus.
Dideli duomenų kiekiai	Priklausomai nuo programos tipo, klientų kiekio ir duomenų saugojimo laiko, OLTP sistemų dydis gali labai išaugti. Pavyzdžiui, elektroninės bankininkystės, klientas nori pamatyti visas savo paskutinių metų sąskaitas.

Lentelės tęsinys kitame puslapyje...

lentelė 1 – Tęsinys

Savybė	Paaškinimas
Pasiekiamumas	Pasiekiamumas OLTP sistemose turi būti nuolatinis. Laikina nepasiekiamą sistema gali paveikti didelį klientų skaičių, dėl kurio gali stipriai nukentėti visa kompanija.
Periodiniai naudojimo ypatumai	Gali nutikti taip, kad OLTP sistemų naudojamumas turi kažkokį ciklišumą. Pavyzdžiui, kiekvieno mėnesio gale yra apskaičiuojama visų banko sąskaitų suvestinė.

1.2 Realus laiko veika OLTP sistemose

Realus laiko veika reiškia, kad su OLTP sistema sąveikaujantys agentai tikisi sinchronizuoto atsako į savo veiksmus. Kitaip tariant, operacijos sistemoje turi būti atliekamos pakankamai sparčiai, kad jų rezultatai būtų pateikti laiku, kol dar yra aktualūs. Pavyzdžiui tarkime, kad mūsų OLTP sistema yra skirta pardavinėti bilietus: kasininkė negali laukti valandos ar dviejų, kol sistema atliks bilieto pirkimo operacijas apie ką tik atliktą pardavimą, nes kasininkei reikia aptarnauti kitus bilieto laukiančius klientus, tačiau sistema turi nuolat dirbti su nujausia duomenų būsena, gauname išvadą, kad OLTP sistema turi atlikti reikalingas operacijas per griežtai apibrėžtą laiką. Realus laiko veikos užtikrinimas yra vienas svarbiausių OLTP sistemų bruožų.

2 OLTP palyginimas su OLAP sistemomis

Labai dažnai IT srities žmonės painioja arba tiesiog nežino koks yra skirtumas tarp OLTP ir OLAP³ sistemų. Šiame skyriuje stengsimosi apžvelgti esminius šių sistemų skirtumus.

³OLAP - trūpinys angl. OnLine Analytical Processing - tiesioginis analitinis apdorojimas.

2.1 OLAP sistemos

Anglų kalboje trumpinys OLAP siejamas su OnLine Analytical Processing sąvoka. Šis terminas naudojamas norint apibūdinti programinius produktus, kurie leidžia visapusiškai analizuoti verslo informaciją realiuoju laiku. Sąveika su tokiomis sistemomis vyksta interaktyviai, atsakymai net į daug skaičiavimų reikalaujančias užklausas gaunami per kelias sekundes. Galutinė informacija gali būti pateikta ne tik skaičiais, bet ir lengviau vartotojui suvokiamu grafiniu pavidalu.

Dauguma OLAP produktų pasižymi draugiška vartotojui aplinka, o kreipiantis į duomenų šaltinius reikiamą verslo informaciją galima gauti net ir nežinant, kaip rašyti sudėtingas užklausas.[IDA03]

Viena svarbiausių ir OLTP, ir OLAP sistemų dalių yra vieno ar kito tipo duomenų bazės. OLTP sistemose paprastai yra reliacinė duomenų bazė, o OLAP yra naudojami duomenų sandėliai.

2.2 Kas yra duomenų sandėliai?

Duomenų sandėlis (angl. Data Warehouse) yra tiesiog viena, užbaigta (angl. complete) ir suderinta saugykla duomenų, surinktiems iš keletos skirtingų šaltinių, pateikimui galutiniams tos informacijos vartotojams tokiu būdu, kad ji jiems būtų suprantama ir naudojama jų veiklos kontekste. (Barry Devlin, IBM konsultantas)[EDW06].

Duomenų sandėlis yra

- Specifiškos paskirties (angl. subject-oriented);
- Integruotas, suderintas (angl. integrated);
- Nuo laiko priklausantis (angl. time-varying) (sistemos darbo rezultatai priklausomi nuo to, koku laiko momentu sistema dirba. Šiuo atveju nuo sandėlyje esančių duomenų, nes jie laikas nuo laiko yra papildomi);
- Ne laikinas (angl. non-volatile).

duomenų rinkinys, kurio pirminis tikslas yra **organizuotas sprendimų priėmimas** (paremtas verslo sukauptais duomenimis).

Pagrindinės duomenų sandėliu besinaudojančių programų klasės:

- OLAP programos;
- Duomenų gavybos (angl. data mining) programos;
- Vizualizavimo (angl. visualization) programos.

Duomenų sandėliai turi tris veiklos lygius:

1. Neapdorotų duomenų lygis (angl. staging);
2. Integracijos lygis (angl. integration);
3. Duomenų pasiekimo lygis (angl. access).

Alternatyvus duomenų sandėlio apibrėžimas – duomenų bazė skirta ruošti ataskaitas ir duomenų analizę bei ilgalaikiam duomenų saugojimui. Duomenys į duomenų sandėlį yra pakraunami iš sistemų naudojamų kasdienėms transakcijoms, t.y. OLTP sistemų.

2.3 OLTP ir duomenų sandėlių skirtumai

2 lentelė: OLTP sistemų ir duomenų sandėlių savybių palyginimas

Savybė	OLTP	Duomenų sandėlis
Paskirtis (angl. Workload)	Iš anksto apibrėžtos operacijos.	Operacijos pagal poreikį.

Lentelės tęsinys kitame puslapyje...

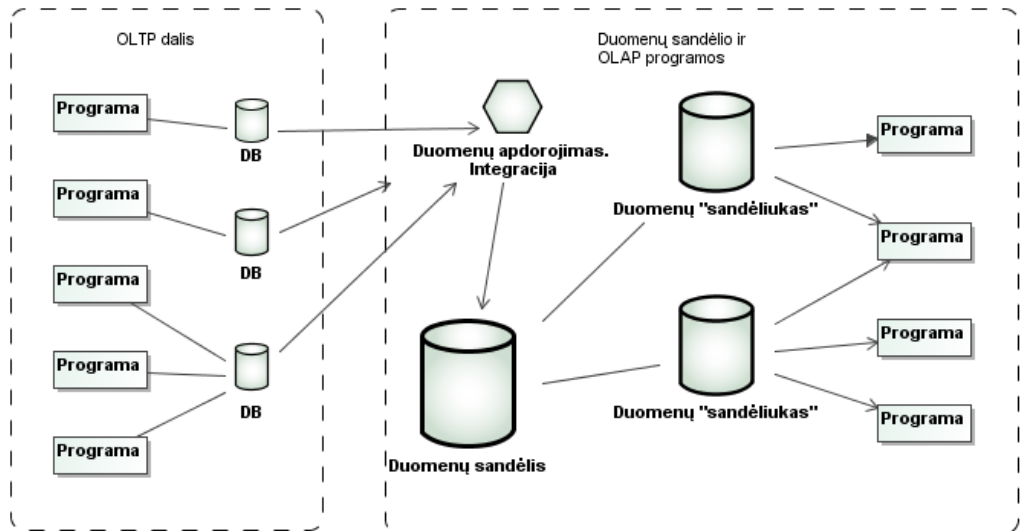
lentelė 2 – Tęsinys

Savybė	OLTP	Duomenų sandėlis
Duomenų redagavimas	Visada turi naujausius duomenis ir duomenys dažniausiai keičiami su kiekviena transakcija. Galutiniai vartotojai nuolat modifikuoja DB.	Reguliarus atnaujinimas pagal ETL (angl. Extract, Transform, Load) procesus. Galutinis naudotojas tiesiogiai nekeičia DB būsenos.
DB schema	Pilnai normalizuota, garantuojanti duomenų vientisumą, bei optimizuota duomenų atnaujinimo, įterpimo ištrynimo operacijoms.	Denormalizuota – optimizuota keletui svarbiausių operacijų.
Užklausų (angl. Queries) tipai	Užklausos naudoja nedaug ir tik reikalingus DB įrašus. Tipinė užklausa: „Gauti paskutinio kliento užsakymo duomenis“	Užklausos skenuoja tūkstančius arba milijonus eilučių. Tipinė užklausa: „Rasti sumą visų praėjusio mėnesio pasdavimų“
Istoriniai duomenys	Tipinis duomenų saugojimo laikas yra kelios savaitės ar mėnesiai. Saugoma tokia informacija, kurios reikia einamosios transakcijos įvykdymui.	Daug istorinių duomenų (mėnesiai, metai). To reikia, kad būtų galima atlikti išsamią duomenų analizę.

2.4 Skirtumų apibendrinimas

Duomenų sandėliuose yra saugomi duomenys sugeneruoti OLTP sistemoje, kuriuos prieš sukraunant į duomenų sandėlį reikia surinkti iš daugelio šaltinių, apdoroti ir pakrauti. Duomenų sandėliai yra skirti verslo duomenų analizės atlikimui, tendencijų ieškojimui ir kitoms darbui su istoriniais duomenimis.

menimis susijusioms operacijoms, o OLTP sistemos skirtos naujausių verslo duomenų apdorojimui.



1 pav.: OLTP ir duomenų sandėlio struktūrinė schema.

3 OLTP sistemų tendencijos

Vis daugiau ir daugiau programų šiandien veikia tiesiog interneto naršyklėje:

- Internetinės parduotuvės (pvz. „iTunes“);
- Elektroninio pašto klientai (pvz. „Gmail“);
- Socialinės programos (pvz. „Facebook“);
- Duomenų apdorojimo programos (pvz. „Google Docs“);
- Žaidimai (pvz. „Angry Birds“);
- Verslo sistemos (pvz. „erpnext“) ir t.t.

Tačiau, kad ir kokį gerą interneto ryšį turėtume, mes negalima garantuoti, kad mes juo visada galėsime naudotis, o jei ir galėsime, galbūt jis tiesiog bus nepakankamas, kad galėtume naudotis patogiai, pvz. naudojames mobiliuoju internetu, tačiau nuvažiavus į kalnus ryšio signalas tampa labai silpnas, smarkiai sumažėja mobilaus interneto greitis ir mes nebegalime išnaudoti kelių leisvų minučių žaisdami savo pamėgtojo „Angry Birds“ žaidimo internetinės versijos. Žinoma, kad dažniausiai galime turėti panašios paskirties programą pilnai įrašytą į kompiuterio atmintį, bet pasaulinė tendencija yra tokia, kad stipriai populiarėja taip vadinama „debesų kompiuterija“ (angl. Cloud computing) ir su ja susiję programiniai sprendimai. Atsiranda realus poreikis turėti mišraus veikimo programas tokias, kurių prigimtis yra internetinė, tačiau jos gali veikti ir nesant arba esant tik labai silpnam interneto ryšiui.

Pradinė interneto naršyklių idėja buvo peržiūrėti duomenis esančius kažkur nutolusiame interneto serveryje, o tam būtinai reikia interneto ryšio. Kaip jau minėjome anksčiau, yra poreikis peržiūrėti tuos pačius duomenis ir kai nėra interneto ryšio. Kyla problema, kaip pasiekti nepasiekiamus duomenis? Sprendimas yra gana paprastas – reikia padaryti duomenis visada pasiekiamus - įrašyti būtinus internetinės programos duomenis vartotojo

kompiuteryje. (Bet tada interneto naršyklė tampa jau nebe interneto naršykle, o tiesiog naršykle, nes gali naršyti, ne tik internetą, bet ir (kažkokiam lygyje) kliento kompiuterį.) Technologijų leidžiančių tokias programas kurti jau yra, kaip jau minėta, yra ir poreikis tokioms programoms. Kita darbo dalis bus susijusi su istorine apžvalga bei raidos perspektyvomis technologijų, leidusių įrašyti duomenis į kliento kietąjį diską ir taip leidusių dalį internetinės programos veikimo perkelti į kliento kompiuterį.

4 Senesnės technologijos internetinių programų duomenų saugojimui kliento kompiuteryje

4.1 Slapukai

Interneto svetainės naudoja slapukus (angl. cookies), kad galėtų pasiūlyti personalizuotą naudojimo patirtį vartotojams ir surinkti informaciją apie svetainės naudojimą. Daugelis svetainių taip pat naudoja slapukus, kad galėtų saugoti informaciją, kuri padeda nuosekliai naudoti svetainės dalis, pvz., parduotuvės krepšelį arba tinkintus puslapius. Patikimoje svetainėje slapukas gali pagerinti darbą leisdamas svetainei sužinoti kliento prioritetus arba praleisti prisijungimo etapą kaskart, kai einate į jau lankytą interneto svetainę. Tačiau kai kurie slapukai, pvz., įrašyti reklaminių antraščių, gali kelti grėsmę privatumui ir sekti lankomas svetaines.

Slapukų idėja yra gana sena – ji kilo Lou Montulli, o pirma slapukų specifikacija išleista 1994 metais spalio 13. Technologijos tikslas – kliento kompiuteryje saugoti informaciją susijusią su naršymu tinklalapyje. Tuo metu, kada atsirado slapukai, daugelis internetinių programų juos naudodavo tik su identifikacija susijusių duomenų saugojimui, kai tuo tarpu visi kiti duomenys susiję su klientu buvo saugomi serveryje. Kadangi slapukai yra tekstiniai jie nėra vykdomieji failai, todėl jie negali daugintis ir negali būti virusais. Naršyklė serveriui siunčia tik „rakto -> reikšmės“ (angl. „name-

>value“) porą, o visi kiti atributai yra naudojami slapuko ištrynimo datai apskaičiuoti. Duomenų, saugomų slapukų pavidalu, kiekis yra labai mažas, tačiau tai vis tiek yra „atjungtos atmintinės“ (angl. offline storage) klasės technologija.

3 lentelė: Slapukų rūšys

Rūšis	Veikimo pobūdis
Sesijos arba seanso (angl. Session)	Tokio tipo slapukas yra aktyvus tol kol, klientas prisijungęs prie portalo arba, jei nustatyta, gali gyvuoti dar tam tikrą laiko tarpą iki kito puslapio aplankymo, bet, jei apsilankymas neįvyksta per tam tikrą laiką, jis automatiškai yra ištrinamas.
Ilgaliakiai (angl. Persistent)	Kiekvieną kartą klientui prisijugus prie serverio slapukas yra siunčiamas klientui, taip galima skaičiuoti, kiek kartų vartotojas apsilankė portale.
Saugusis (angl. Secure)	Naudojamas tik jungiantis per „https“ ⁴ protokolą ir slapuko turinys yra užšifruotas tam, kad būtų mažesnė tikimybė nukentėti nuo duomenų vagystės.
Tik http (angl. HttpOnly)	Šiuo slapuku gali naudotis tik HTTP (ar HTTPS) tipo jungtis ir slapuko turinys yra nepasiekiamas kliento skriptams, taip slapukas yra apsaugotas nuo skriptinių atakų.
Trečiosios šalies (angl. Third-Party)	Šio tipo slapukai gaunami iš kitų svetainių reklaminių skelbimų (pvz., iššokančių reklamų arba reklaminių antraščių), esančių rodomoje svetainėje. Svetainės gali naudoti šiuos slapukus rinkodaros tikslais ir žiniatinklio naudojimui sekti.

Lentelės tęsinys kitame puslapyje...

⁴HTTPS - angl. HyperText Transfer Protocol Secure

lentelė 3 – Tęsinys

Rūšis	Veikimo pobūdis
Super (angl. Super)	Įrašo tik aukšto lygio domenus „.com“ ir tokiu būdu gali trikdyti naudojimąsi internetu. Naršyklės turi sąrašus neleistinių super slapukų, kuris yra nuolat atnaujinamas.
„Zombis“ (angl. Zombie)	Šio tipo slapukai atsikuria save po to kai buvo ištrinti (atkuriantis skriptas gali būti laikomas „Flash“ lokaliaje bendro naudojimo duomenų saugykloje).

4.2 JavaScript ir failai vartotojo kompiuteryje

JavaScript – objektiškai orientuota skriptų programavimo kalba. Ji dažniausiai yra naudojama internetinių puslapių interaktyvumo realizacijai. Javascript kodas negali pasiekti kliento kompiuterio failų sistemos. JavaScript kodas gali manipuluoti failais kurie buvo įtempti (angl. Drag and Drop) į naršyklę. Taip pat yra galimybė įrašyti failą į kompiuterio atmintį, tačiau klientas turi užbaigti išsaugojimo (angl. Save As..) dialogą. Taip yra dėl to, kad JavaScript kodas veikia savotiškame smėlio dėžės (angl. sandbox) režime. Smėlio dėžės režimas naudojamas kaip būdas apsaugoti kliento kompiuterį nuo kenksmingų internetinių atakų.

Nors JavaScript kodas veikia smėlio dėžės režimu, tačiau su tokiu režimu derinasi ir su internetiniu domenu susijusios lokalsios failinės sistemos koncepcija[FIA10]. Ateityje turėtų atsirasti galimybė pilnai dirbti su failais susietais su konkrečiu domenu. Ši koncepcija leis kliento kompiuteryje sukurti failinę struktūrą, kuri bus pasiekama per interneto naršyklę tik tada kai bus prisijungta prie domeno, kurio vardu toji failinė sistema buvo sukurta.

4.3 Papildiniais grįstos technologijos

Šiame skyriuje nagrinėsime papildiniais (angl. plugin) grįstas technologijas leidžiančias internetinėms programoms pasiekti duomenis kliento kompiuteryje. Jų yra trys:

- „Flash“;
- Programėlės;
- „Google Gears“.

Taip pat aptarsiu bendrus papildinių naudojimo trūkumus, bei aprašysiu atskirų naršyklių bandymus dirbti su kliento kompiuteryje esančių duomenų pasiekiamumu.

4.3.1 „Flash“

Nuo 6 Flash versijos atsirado vietinio bendro naudojimo objekto” (angl. local shared object)[LSO11] koncepsija. Vietinio bendro naudojimo objektas dar yra vadinamas “Flash slapuku”, taip yra dėl to, kad jų paskirtis kaip ir paprastų slapukų yra saugoti duomenis susijusius su internetiniu puslapiu kliento kompiuteryje. Didžiausias skirtumas yra tas, kad vietinio bendro naudojimo objektas yra sukuriamas tik puslapių naudojančių „Flash“ technologijas ir lokaliai gali būti naudojamas tik „Flash“ papildinius turinčiose sistemose. Taip pat jų naudojimo nustatymai yra nepriklausomi nuo paprastųjų slapukų naudojimo nustatymų, tai reiškia, kad net, jei paprastieji slapukai yra griežtai draudžiami, vietiniai bendro naudojimo objektai gali būti naudojami.

Saugomų duomenų pobūdis yra šiek tiek platesnis negu paprastų slapukų atveju. Dažniausiai yra saugomi vartotojo nustatymai. Jei yra naudojami standartiniai vietinio bendro naudojimo objekto nustatymai, tuomet kliento nėra klausama ar naudoti šio tipo atmintinę. Taip pat standartinis duomenų kiekis saugomas vietiniame bendro naudojimo objekte yra 100 kB, tačiau esant reikalui šis dydis gali būti keičiamas.

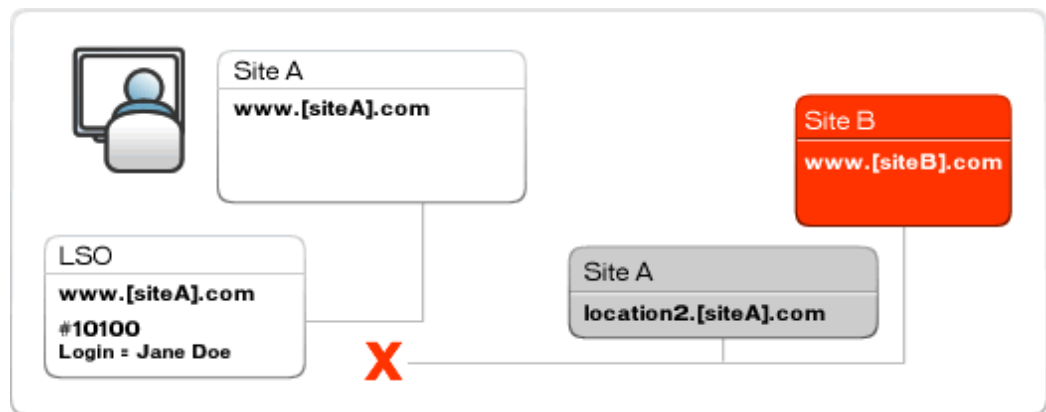
4 lentelė: Vietinio bendro naudojimo objekto privalumai ir trūkumai

Privalumai	Trūkumai
<ul style="list-style-type: none"> • Jis yra tinkamas saugoti duomenis susijusius su transakcijomis, nes gali saugoti ir sudėtingesnės struktūros negu paprastas tekstas duomenis; • Jo dydis gali būti plečiamas; • Technologija yra gana plačiai paplitusi, nes dauguma Windows OS naudotojų turi Flash įskiepius. 	<ul style="list-style-type: none"> • Mažai žinomas, todėl prastai kontroliuojamas. Tik Google Chrome naršyklė turi modulių leidžiančius pilnai kontroliuoti veikimą; • Įvairūs saugumo klausimai; • Gali būti naudojamas kenkėjiškai veiklai (Pavyzdžiui, vietinis bendro naudojimo objektas gali būti kaip atsarginė paprasto slapuko kopija skirta atkurti tą slapuką, kai jis yra ištrinamas); • Nėra globalus standartas;

Nors ši technologija yra gana dažnai kritikuojama dėl saugumo problemų, ją kuriančios kompanijos „Adobe Inc.“ vadovai nuolat pabrėžia, kad ši technologija nebuvo projektuojama kaip grėsmė galintis kelti įrankis. Tai galime matyti ir iš vietinio bendro naudojimo objekto naudojimo schemos.

4.3.2 Programėlės

Viena iš Java programavimo kalbos galimybių yra programėlių[?] rašymas. Java programėlės tai yra speciali Java programų grupė, kuri gali būti



2 pav.: Vietinio bendro naudojimo objekto veikimo schema.

atsisiųsta iš interneto ir įvykdyta aktyvų Java įskiepi turinčioje naršyklėje. Java kaip ir absoliuti dauguma programavimo kalbų turi galimybę skaityti ir rašyti failus į kliento kompiuterį. Programėlių naudojimas yra patrauklus tuo, kad jie yra daug spartesni už JavaScript kodą, bei gali geriau išnaudoti kliento kompiuterio resursus. Dėl ypatingos programėlių technologijos jos gali iškviesti JavaScript kodą, o savo rėštu JavaScript kodas gali iškviesti programėles. Kadangi programėlės yra galingi tiek pat kaip ir kitos Java programos, jų naudojimą reikia riboti, nes dėl netinkamo jų naudojimo gali iškilti daugybė problemų, todėl patartina leisti įvykdyti apletus atsisiųstus tik iš patikimų internetinių puslapių. Java technologijų naudojimas gali būti išeitis, kai norima turėti galimybę dirbti su duomenimis esančiais kliento kompiuteryje.

4.3.3 „Google Gears“

„Google Gears“ [STG11] interneto naršyklės papildinys, kuris įgalina kurti funkcionalines internetines programas, kurių dalis funkcionalumo yra pasiekama neprisijungus prie interneto ryšio. „Google Gears“ plėtojimas nutrūko, nes buvo nuspręsta turėtą įdirbį bei sukauptą patirtį investuoti į bendrų interneto standartų plėtojimą. Dauguma idėjų (vietinė SQL duomenų bazė, geolokacija ir t.t.), kurias plėtojant „Google Gears“ buvo pionierius,

perima arba ateityje perims HTML5 ir kiti interneto standartai.

4.3.4 Papildinių naudojimo ypatybės

Akivaizdžiausias trūkumas yra tai, kad internetinės programos kūrėjas turi tikėtis, kad klientas turėsis įdiegtą papildinį ir jis bus tinkamas naudoti versijos. Tačiau to tikėtis yra gana sudėtinga, nes pvz. kompanijos neretai turi ugniasienes, kurios blokuoja papildinius, arba dar blogiau – papildiniai gali būti nepritaikyti veikti su visomis operacinėmis sistemomis. Dar daugiau, programų kūrėjas turi tikėtis, kad papildinius kurianti kompanija nenutrauks jų kūrimo, bei sieks jį populiarinti. Taipogi, yra grėsmė, kad papildiniais grįstas technologijas pakeis su jais nesuderinami atviri standartai, kas dabar ir vyksta.

Tačiau gali būti ir tokių atvejų, kada papildinių naudojimas teiktų privalumus. Pavyzdžiui, įmonė sukūrė kažkokią paslaugą, kuri yra vertinga ir veikia kažkokio tai papildinio pagrindu. Jei klientams tikrai reikia tos paslaugos, tai jie gali įsidiegti reikalaujamus papildinius ir sėkmingai naudotis tomis paslaugomis. Tad papildinių naudojimas nors ir su akivaizdžiais savo trūkumais gali suteikti privalumų sunkiai pasiekiamų jų nenaudojant.

4.4 Specifinės naršyklių savybės

Darbo su kliento kompiuterio failine sistema specifinius modulius turi ir keletas naršyklių.

5 lentelė: Specifinės naršyklių savybės darbui su kliento failine sistema

Naršyklė	Technologija	Aprašymas
„Internet Explorer“	„UserData API“[UDA11]	Ši technologija leidžia saugoti ir naudoti internetinės programos duomenis tarp atskirų prisijungimų prie jos kliento duomenų atmintinėje (angl. UserData Store). Technologija įgalina saugoti ir naudoti sudėtingas ir dinamiškas duomenų struktūras, kurių dydis yra didesnis negu leidžia slapukai. Kliento duomenų atmintinės dydis priklauso nuo to, kokiame saugumo lygyje yra dirbama, tačiau vidutiniai atminties dydžiai yra 128 KB per failą ir 1024 KB bendros atminties.
„Firefox“	XPCOM (angl. Cross Platform Component Object Model)[FIO11]	Tai galingas karkasas (angl. framework) skirtas multiplatforminės ir modulinės architektūros programų kūrimui. Šios platformos pagrindu yra sukurta pati „Firefox“ naršyklė. Ši naršyklė yra gera tuo, kad jai galima rašyti įvairius papildinius (angl. add-ons), kurie gali išnaudoti XPCOM suteikiamus API, tame tarpe ir darbo su failais modulius. Gaunamos iš esmės neribotos programų kūrimo galimybės.

Tačiau visos šios specifinės naršyklių technologijos turi keletą esminių trūkumų:

- Programa pritaikyta dirbti su specifiniais moduliais, veiks geriausiai atveju tik keletoje naršyklių;

- Naršyklės kūrėjas gali nutraukti modulio palaikymą;
- Jei naršyklė bando pasinaudoti tais moduliais (tarkim naujesnė naršyklės versija, kuri jau nebepalaiko tų modulių), bet jiems nėra pritaikyta, tada reikia pasikliauti Java apletais, kurie irgi turi įvairių neigiamų jau aptartų savybių.
- Technologijos gali būti prastai dokumentuotos;
- Technologijos greičiausiai yra nepopuliarios.

5 HTML5 standarto naujovės susijusios su tiesioginiu transakcijų apdorojimu

HTML5 standartas turi visą klasę naujovių pavadintų „Darbas atsijungus ir atmintinė“ (angl. offline & storage). Šią klasę sudaro keletas naujų standartų:

- Programų podėlis (angl. application cache);
- Vidinė atmintinė (angl. local storage, web storage);
- Saityno SQL⁵ duomenų bazė (angl. web SQL database)
- Indeksuota duomenų bazė (angl. indexed database);
- Failų API⁶ sąsaja (angl. File API).

Visos šios technologinės naujovės įgalina kurti internetines programas, kurias būtų galima priskirti OLTP klasei. Dar daugiau, šios technologijos iš dalies išsprendžia klausimą, kaip naudoti internetines programas, kai nėra interneto ryšio. Tolesniuose skyriuose bus trumpa šių technologijų apžvalga.

⁵SQL - trumpinys angl. Structured Query Language

⁶API (angl. Application Program Interface) - programų sąsaja.

5.1 Programų podėlis

Vis didesnę svarbą turi internetinės programos, todėl darbas su jomis atsijungus prie interneto yra galimybė tas programas dar labiau pagerinti. Nors dauguma naršyklių turi padėjimo (angl. caching) mechanizmus, tačiau jie ne visada veikia taip, kaip tikimasi. Šiaip problemai spręsti HTML5 standarto kūrėjai siūlo naudoti programų podėlio suteikiamas galimybes. Pagrindiniai privalumai, kuriuos suteikia Programų podėlio technologija yra:

- Naršymas neprisijungus – galima naršyti internetinius puslapius neprisijungus prie interneto;
- Greitis – užkešuoti resursai yra pakraunami iš vietinės atmintinės, todėl tai vyksta greičiau;
- Sumažinta serverių apkrova – serveriui reikia siųsti tik tuos failus, kurie buvo modifikuoti arba, kurių nėra prasmės saugoti kliento kompiuteryje.

Pagrindinė šios technologijos savybė yra ta, kad programuotojas gali pasakyti naršyklei, kuriuos failus reikia padėti kliento kompiuteryje. Tai yra padaroma labai paprastu būdu:

1. Internetinio puslapio „html“ žymėje reikia nurodyti manifesto failą;
2. Manifesto faile reikia nurodyti failus, kuriuos reikia užkešuoti.

Šios technologijos pritaikymas daugusia naudos duoda vartotojo grafinei sąsajai sukurti, kol yra nepasiekiamas serveris, o pati technologija nieko nesako apie transakcijų palaikymą. Šiai problemai spręsti yra sukurtos kitos technologijos.

5.2 Vidinė atmintinė

Dar ši technologija vadinama „DOM⁷ Storage“. Vietinė atmintinė gali būti supaprastintai suprasta kaip slapukų išplėtimas pridėdant didesnės at-

⁷DOM - trumpinys angl. Document Object Model

minties palaikymą ir supaprastinant darbo su ta atmintine interfeisą. Kaip ir slapukų atveju vietinės atmintinės duomenys yra nestruktūrizuoti simboli-
nio tipo duomenys, kurie yra organizuojami atvaizdžio (ang. map) principu. Tačiau iš tikrųjų ši technologija skiriasi keletu esminių aspektų, apie kuriuos iš eilės.

Lokali saugykla yra skirstoma į dvi dalis: sesijos (ang. session) ir nuolatinės (angl. local) saugyklos. Sesijos atmintinė yra naudojama, kol klientas yra prijungtas prie tinklalapio arba kol tinklalapis (ar naršyklės kortelė, ar naršyklės langas) nėra uždarytas. Kai tinklalapis yra uždaromas visa atmin-
tis sukaupia sesijos atmintinėje yra ištrinama. Nuolatinės saugyklos atveju atmintinėje esantys duomenys nėra ištrinami uždarius naršytą tinklalapį – ji lieka kliento kompiuterio atmintinėje tarp naršyklės sesijų, todėl tinka duomenų saugojimui ilgą laiką.

Ši technologija gali gana stipriai prisidėti prie greitesnių programų kūri-
mo, nes ji gali pakeisti slapukų naudojimą. Slapukai lėtina programos veiki-
mą, nes jie turi būti perduoti serveriui kaskart kreipiantis į jį. Vienas iš būdų kaip išvengti šios problemos yra kaip galima labiau sumažinti slapukų dydį. Kitas – naudoti vietinės atmintinės galimybes. Ši technologija yra geresnė, nes jos visų duomenų nereikia kaskart perdavinėti serveriui. Dar daugiau, ši technologija turi elegantiškesnį darbo interfeisą. Vietinė atmintinė gali būti naudojama vietoj slapukų daugelyje atvejų.

Būtina paminėti, kad vietinės atmintinės technologija turi keletą prob-
lemų. Visų pirma su transakcijų palaikymu – pvz. programa atidaryta dvejo-
se kortelėse ir viena kortelė įrašinėja duomenis, o kita tuo pat metu pradeda
juos skaityti iš tos pačios atmintinės – šioje situacijoje neaišku ar rezultatai
bus tokie kokių tikimasi. Antra, jei duomenų kiekiai darosi dideli, paieška
jose stipriai letėja, nes nėra naudojami jokie indeksai.

5.3 Saityno SQL duomenų bazė

Visų pirma reikėtų paminėti, kad saityno SQL duomenų bazės standar-
to plėtojimas yra nutrauktas 2010 metų lapkričio mėnesį. Nors ši technologija

ir nėra oficiali HTML5 standarto dalis, tačiau ji yra įgyvendinta kai kurių didžiųjų naršyklių, todėl ji yra įdomi kaip karkasas, kuriuo remiantis būtų galima kurti visokeriopo funkcionalumo taikomąsias programas. Šioje darbo dalyje apžvelgsime svarbiausias šios technologijos savybes bei galimybes, remiantis šia technologija, sukurti OLTP aplikacijas.

Saityno SQL duomenų bazė yra kliento kompiuteryje esanti duomenų bazė, kuri suteikia pilną SQL funkcionalumą internetinėms aplikacijoms. Ši technologija pagal funkcionalumą ir sudėtingumą yra kaip ir tipinė reliacinė SQL DB. Ji veikia SQLite duomenų bazių valdymo sistemos pagrindu, kuri yra labai panaši į kitas populiariausias duomenų bazių valdymo sistemas.

OLTP sistemoms yra labai svarbus transakcijų palaikymas. Ši technologija yra kaip tik tai ko reikia, nes ji kaip ir visos kitos duomenų bazių valdymo sistemos pilnai palaiko transakcijas.

6 lentelė: Saityno SQL duomenų bazė

Privalumai	Trūkumai
Galima saugoti labai struktūrizuotus duomenis	Internetinės programos gali tapti neelegantiškomis, sudėtingomis, brangiomis, nestabiliomis ir t.t.
Galima kurti labai funkcionalius naršyklių papildinius	Palaikymo problema: naršyklės arba palaiko, arba palaiko dalinai, arba visai nepalaiko šio standarto
Tinka programų (pvz. elektroninio pašto klientai) dirbančių atsijungus duomenų saugojimui	Ganaribotas pritaikomumas, nes dalis naršyklių nepalaiko šios technologijos, todėl gali tekti kurti keletą tos pačios programos versijų, kad visa tai veiktų pas daugelį klientų.

Lentelės tęsinys kitame puslapyje...

lentelė 6 – Tęsinys

Privalumai	Trūkumai
Nelieka sinchronizavimo problemų, kai su ta pačia programa dirbama atskirose kortelėse ar languose, nes tas problemas sutvarko duomenų bazių valdymo sistema.	Kadangi yra panašias galimybes siūlančių technologijų (pvz. vietinė atmintinė), saityno SQL duomenų bazės turėtų būti skirtos daugiau stambiems duomenis saugoti kliento kompiuteryje, kas vargu ar yra gero stiliaus bruožas internetinėms programoms.
Duomenų vientisumas užtikrinamas, nes pilnai palaikomos transakcijos	Nepatogu naudoti su JavaScript, nes yra tik keletas metodų, o užklauskos yra tų metodų parametrai, kurie yra paprastos simbolių eilutės.

Vienas iš geresnių saityno SQL duomenų bazės naudojimo pavyzdžių būtų žinučių programa. Jei žinutės būtų saugomos kliento kompiuteryje, tai būtų galima sutaupyti kiekvieną kartą nesikreipiant į serverį, kai norima atlikti tokias operacijas kaip: žinučių paieška, filtravimas, rykiavimas. Dar daugiau, būtų galima atlikinėti paieškas ar kitas darbo su žinutėmis operacijas ties kiekvienu mygtuko paspaudimu, taip programas padarant interaktyvesnėmis.

5.4 Indeksuota duomenų bazė

Indeksuotos duomenų bazės technologija skirta duomenų saugojimui kliento kompiuteryje. Vienas pagrindinių tikslų yra saugoti didelius duomenų kiekius darbui atsijungus. Ši technologija yra kaip kompromisas tarp vidinės atmintinės paprastumo ir saityno duomenų bazės sudėtingumo. Ji suteikia galimybę saugoti sąlyginai didelius struktūrizuotų (nors neturinčių bendros schemos) duomenų kiekius - pradinis DB dydis yra 5MB, bet esant reikalui atmintinę galima padidinti. Taip pat ši technologija leidžia greitas

paieškas saugomuose duomenyse - to pasiekama naudojant indeksavimo mechanizmus. Indeksavimas atliekamas tik kažkokiai vienai objekto savybei (angl. property).

Kadangi šios technologijos kūrėjai kalba apie sąlyginai didelius duomenų kiekius, tad iškyla spartos klausimas, nes kaip žinia JavaScript naršyklėje veikia tik viena gija, todėl jai davus daug darbo (kitais tarant bus atliekama didelė transakcija) visi kiti procesai naršyklėje laukia kol tas darbas bus atliktas. Vienalaikiško veikimo užtikrinimas yra svarbus OLTP sistemoms. Šis klausimas yra išspręstas naudojant asinchroninius kvietimus. Tokiu būdu visas darbas yra atliekamas foniniu režimu, o programa yra pilnai interaktyvi.

Saugumo klausimai yra sprendžiami leidžiant klientui pačiam pasakyti ar galima sukurti naują indeksuotą DB - prieš sukuriant naują DB yra klausiama kliento ar jis duoda leidimą ją sukurti. Jei naršymas vyksta kokiam nors viešame kompiuteryje (pvz. bibliotekoje), tai bendru atveju siūloma dirbti privačiu režimu. Dirbant privačiu režimu indeksuotų duomenų bazių kūrimas yra uždraustas.

Ši technologija turi keletą trūkumų:

- Norint pridėti ar panaikinti indeksus, reikia būtinai pakeisti DB versiją, kurių ne visada patogiu administruoti;
- Nėra paprasto būdo atlikti duomenų rykiavimo, rūšiavimo operacijas - viską reikia pasidaryti „Rankutėmis“;
- Nepalaikomi lentelių (angl. object stores) sujungimai (angl. join);
- Norint atlikti paiešką pagal kažkokią saugomų objektų savybę, ta savybė būtinai turi būtų suindeksuota;
- Indeksai gali užimti daug atminties.

5.5 Failų API

[TODO]

5.6 Duomenų kiekiai

Visomis aukščiau minėtomis technologijomis kliento kompiuteryje esantys duomenys yra susiejami su vienu domenu ir tie duomenys nėra pasiekiami iš kitų domenų – tai lyg ir „smėlio dėžės“ (angl. Sandbox) principas.

Vienas domenas kliento kompiuteryje pavyzdžiui gali turėti:

- 5MB Web Storage
- 25MB Web SQL Database
- Negali turėti Indexed Database

Viso 30 MB, kas yra gana nemažai, tačiau šis dydis lengvai gali būti padidintas iki tiek kiek reikia.

Bendru atveju duomenų kiekiui pasiekus maksimalią ribą, naršyklė gali paklausti kliento, ar galima padidinti normas. Dabar standartinis maksimalų duomenų kiekį nusakantis skaičius yra 5MB - iki tokio duomenų kiekio naršyklė kliento gali ir neprašyti leidimo duomenų saugojimui.

5.7 Naujų technologijų palaikymas naršyklėse

Situacija naršyklių rinkoje dabar yra tokia, kad ne visos naršyklės viską palaiko ir net ir tos, kurios palaiko, palaiko ne viską vienodai. Bet visų šių technologijų tikslai yra du (įgalinti internetines programas dirbti atsijungus, ir padidinti darbo spartą) ir jie niekaip nesako, kad šių technologijų nepalaikymas reikš, kad jų pagalba pagamintos programos neveiks su visomis naršyklėmis – blogiausiu atveju viskas veiks tik prisijungus.

7 lentelė: Naujų technologijų palaikomumas

Technologija	Palaikomumas
Aplikacijų talpykla	Palaikoma visų naujausių ir populiariausių naršyklių.

Lentelės tęsinys kitame puslapyje...

lentelė 7 – Tęsinys

Technologija	Palaikomumas
Vietinė atmintinė	Palaikoma visų naujausių ir populiariausių naršyklių.
Saityno SQL duomenų bazė	Palaikoma „Google Chrome“, „Safari“ ir „Opera“ naršyklių. Oficialiai standarto plėtojimas yra nutrauktas. „Firefox“ ir „Microsoft“ naršyklių gamintojai pasisako prieš šį standartą ir neplanuoja įgyvendinti šios technologijos.
Indeksuota duomenų bazė	Palaikoma tik „Google Chrome“ ir „Firefox“ naršyklių. Kadangi šis standartas yra plėtojamas, tai laikui bėgant turėtų atsirasti ir kitų naršyklių palaikymas šiai technologijai.

Labiausiai palaikoma naujoji technologija, galinti rimtai prisidėti kuriant internetines OLTP sistemas, kurios galės veikti ir be interneto ryšio, yra vietinė atmintinė, nes ji turi puikų paprastumo ir funkcionalumo santykį. Ateityje svariausiai prie naujo tipo OLTP sistemų kūrimo svariausiai turėtų prisidėti indeksuotos duomenų bazės technologija.

REZULTATAI IR IŠVADOS

Šiame darbe buvo išnagrinėtos OLTP sistemos: apibrėžta tokių sistemų klasė bei peržvelgtos pagrindinės tokių sistemų savybės. Šios sistemos yra sudėtingos dėl griežtų joms keliamų reikalavimų.

Ruošiant medžiagą darbui pastebėta, jog trūksta aiškumo skiriant OLTP nuo OLAP sistemų. Todėl visas skyrius buvo skirtas išsiaiškinti pagrįn-

dinius tokių sistemų panašumus ir skirtumus.

Didžiausias dėmesys šiame darbe buvo skirtas technologijoms, kuriomis galima grįsti OLTP sistemų, veikiančių tiesiog interneto naršyklėse, veikimą atsijungus nuo saityno. Buvo apžvelgtos jau keletą ar keliolika metų egzistuojančios standartinės technologijos: slapukai bei papildiniais („Flash“, programėlės, „Google Gears“) grįsti sprendimai. Taip pat apžvelgti ir paskirų naršyklių gamintojų mėginimai sukurti sprendimus, leidžiančius internetinėms svetainėms dirbti su kliento kompiuteryje esančia atmintine.

Įvertinau naujuosius interneto standartus, kurie gali kokybiškai pagerinti internetinių OLTP sistemų veikimą ta prasme, kad internetinės programos gali veikti atsijungusios ir pasiekti kliento atmintinę. Įvertinus siūlomas naujoves (programų podėlį, vidinę atmintinę, saityno SQL duomenų bazę, indeksuotą duomenų bazę, failų API sąsaja), belieka sėsti prie klaviatūrų ir kurti naujo tipo programas, kurios bus greitesnės, mobilesnes ir naudingesnės.

Literatūra

- [DGJ08] V. Dagienė, G. Grigas, T. Jevsikova. Enciklopedinis kompiuterijos žodynas. 2-as papildytas leidimas. Vilnius: TEV, 2008, 654 p. Žiūrėta[2011-05-29]. Prieiga per internetą: <http://www.likit.lt/term/enc.html>
- [UPI07] Using Partitioning in an Online Transaction Processing Environment. 2007. [Žiūrėta: 2011-06-01]. Prieiga per internetą: http://download.oracle.com/docs/cd/B28359_01/server.111/b32024/part_oltp.htm
- [EDW06] Enterprise Data Warehouse (EDW). 2006. [Žiūrėta: 2011-06-01]. Prieiga per internetą: <http://edw.berkeley.edu/documents/Enterprise%20Data%20Warehouse%20brown%20bag%20-%202010-5-06.pdf>

- [IDA03] Interaktyvios duomenų analizės įrankiai šiuolaikinėje imonėje. OLAP duomenų bazės. [Žiūrėta: 2011-06-04]. Prieiga per internetą: <http://www.nk.lt/programine-iranga/interaktyvios-duomenu-analizes-irankiai-siuolaikineje-imoneje-olap-duomenu-baz>
- [DWC02] Data Warehousing Concepts. [žiūrėta: 2011-06-04]. Prieiga per internetą: http://download.oracle.com/docs/cd/B10500_01/server.920/a96520/concept.htm
- [FIA10] "Offline": What does it mean and why should I care? [Žiūrėta: 2011-06-04]. Prieiga per internetą: <http://www.html5rocks.com/tutorials/offline/whats-offline/#toc-fileAPI>
- [LSO11] What are local shared objects? [Žiūrėta: 2011-06-04]. Prieiga per internetą: <http://www.adobe.com/products/flashplayer/articles/lso/>
- [WJA01] Java Applet. [žiūrėta: 2011-06-04]. Prieiga per internetą: http://en.wikipedia.org/wiki/Java_applet
- [STG11] Stopping the Gears [Žiūrėta: 2011-06-04]. Prieiga per internetą: <http://gearsblog.blogspot.com/2011/03/stopping-gears.html>
- [UDA11] userData Behavior [Žiūrėta: 2011-06-05]. Prieiga per internetą: [http://msdn.microsoft.com/en-us/library/ms531424\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ms531424(v=vs.85).aspx)
- [FIO11] File I/O [Žiūrėta: 2011-06-05]. Prieiga per internetą: https://developer.mozilla.org/en/Code_snippets/File_I%2F%2FO

SAVOKŲ APIBRĖŽIMAI

Duomenų gavyba (angl. data mining) - duomenų išgavimas iš duomenų bazės, žiniatinklio arba kitokios didelės kompiuterinės duomenų saugyklos panaudojant asociacijas, statistikos, klasifikacijos, segmentavimo,

įvairius euristinius metodus. Naudojama kai operuojama dideliais duomenų kiekiais ir sunku apibrėžti paieškos kriterijus.[DGJ08]

Papildinys (angl. add on; add-in; plug-in; plugin - papildomas komponentas, įtaisytas į programą, kompiuterį arba jo įtaisą ir išplečiantis jo galimybes.[DGJ08]

Programėlės (angl. applet) - Nedidelė programa, kuri yra serveryje ir gali būti iškviesta iš tinklalapio. Vykdoma kliento naršyklėje. Programėlės dažniausiai rašomos Javos kalba. Tam, kad jos galėtų būti vykdomos, naršyklė turi būti suderinama su Java ir parinkta atitinkama programėlių vykdymo nuostata. Javos programėlės pasižymi tuo, kad gali pagyvinti tinklalapį, suteikti jam dinamiškumo ir gali būti vykdomos įvairiose operacinėse sistemose.[DGJ08]

Podėlis (angl. cache) - atmintis, skiriama laikinai padėti programos duomenims, kurių gali vėliau prireikti. Prisipilžius podėliui programa, prieš įrašydama naujus duomenis, pašalina dalį anksčiau įrašytų. Paprastai šalinami tie duomenys, kurių, kaip tikimasi, mažiausiai prireiks, pavyzdžiui, seniausiai įrašyti arba rečiausiai naudoti.[DGJ08]

Slapukas (angl. cookie) - duomenų rinkinys, kurį sukuria svetainė ir įrašo į lankytojo kompiuterį. Tai informacija, kuri paspartina arba supaprastina darbą vėl kreipiantis į tą pačią svetainę, pavyzdžiui, įsimenama anksčiau užpildyta anketa ir nebereikia jos pildyti iš naujo.[DGJ08]

Saitynas (angl. web) - žiniatinklio sinonimas - hipertekstinės informacijos visuotinis tinklas, svarbiausias interneto komponentas.

Foninė programa (angl. background application, background program) - Programa, nesanti veikiamajame lange. Gali atlikti veiksmus, signalizuoti apie jų eigą, bet negali priimti komandų.[DGJ08]