

UNIVERZA V LJUBLJANI
EKONOMSKA FAKULTETA

MAGISTRSKO DELO

ANALIZA UPORABE LAHKIH ODJEMALCEV NA PRIMERU
PROJEKTA FINANCIAL TIMES PALMS - ASIA

Ljubljana, september 2002

Goran Horvat

IZJAVA

Študent GORAN HORVAT izjavljam, da sem avtor tega magistrskega dela, ki sem ga napisal pod mentorstvom prof. dr. Jurija Jakliča in skladno s 1. odstavkom 21. člena Zakona o avtorskih in sorodnih pravicah dovolim objavo magistrskega dela na fakultetnih spletnih straneh.

V Ljubljani, 15.9.2002

Podpis:

KAZALO

1	UVOD.....	1
1.1	Področje proučevanja	1
1.2	Namen	2
1.3	Cilji	2
1.4	Metodološki pristop	2
1.5	Zasnova poglavij	2
2	RAZVOJ RAČUNALNIŠKIH ARHITEKTUR	3
2.1	Obdobje velikih računalnikov	3
2.2	Obdobje osebnih računalnikov	5
2.3	Arhitektura odjemalec / strežnik	6
2.4	Večslojne arhitekture odjemalec / strežnik	7
3	LAHKI ODJEMALCI	8
3.1	Koncept lahkih odjemalcev	8
3.1.1	Windows Terminali.....	8
3.1.2	Omrežni računalnik	9
3.1.3	Omrežni osebni računalnik	10
3.2	Vodilna podjetja na področju lahkih odjemalcev	11
3.3	Prednosti lahkih odjemalcev z vidika skupnih stroškov lastništva	12
4	CITRIX ICA IN MICROSOFT TSE TEHNOLOGIJA LAHKIH ODJEMALCEV	14
4.1	Citrix.....	14
4.1.1	Nastanek Citrix ICA / MetaFrame tehnologije	14
4.1.2	Predstavitev Citrix Metaframe	16
4.1.3	Independent Computing Architecture (ICA) protokol.....	21
4.1.3.1	ICA funkcionalnost	22
4.1.3.2	Delovanje ICA protokola	22
4.1.3.3	Sestava ICA paketa	23
4.1.3.4	ICA protokol in transportne plasti	25
4.1.3.5	ICA razširljivost	26
4.1.4	Orodja za administracijo in dodatki	26
4.1.4.1	Izenačevanje obremenitve	27
4.1.4.2	Objavljanje aplikacij.....	28
4.1.4.3	Senčenje	28
4.1.4.4	Vdelava aplikacij	29
4.1.4.5	NFuse	29
4.1.4.6	Storitve za upravljanje virov	31
4.1.4.7	Storitve za upravljanje namestitve	32
4.1.4.8	VideoFrame.....	33
4.1.4.9	MetaFrame za UNIX	33

4.2	Microsoft Terminal Services Edition (TSE).....	35
4.2.1	Predstavitev Microsoft TSE	35
4.2.2	Večuporabniško jedro Windows NT	36
4.2.3	Remote desktop Protocol (RDP).....	37
4.2.4	Odjemalec RDP	40
4.2.5	Orodja za administracijo.....	42
4.3	Kriteriji za izbiro med Microsoft Terminal Serverjem in Citrix MetaFrame strežnikom.....	45
5	PRIMER UPORABE LAHKIH ODJEMALCEV: FT PALMS ASIA	49
5.1	Opis informacijskega sistema FT PALMS	50
5.2	Opis in namen projekta FT – PALMS ASIA.....	56
5.3	Proces implementacije lahkih odjemalcev	58
5.3.1	Analiza	58
5.3.2	Načrtovanje strežnika.....	59
5.3.2.1	Strojna oprema	59
5.3.2.2	Domenska struktura.....	61
5.3.2.3	Namestitev programske opreme.....	63
5.3.2.4	Določitev strežniških farm	63
5.3.2.5	Nastavitev izenačevanje obremenitve	63
5.3.3	Namestitev aplikacij	65
5.3.3.1	Seznam aplikacij	65
5.3.3.2	Testiranje aplikacij	66
5.3.3.3	Testni sezname.....	68
5.3.4	Testiranje sistema z uporabniki.....	70
5.3.4.1	Namestitev odjemalcev.....	70
5.3.4.2	Izbira uporabnikov.....	71
5.3.4.3	Podpora uporabnikom.....	72
5.3.5	Vzporedna produkcija in produkcija	73
5.4	Pridobljene izkušnje in poslovne koristi	75
6	SKLEP	78
7	LITERATURA	81
8	VIRI	83
	Slovarček slovenskih prevodov tujih izrazov.....	84

1 UVOD

1.1 Področje proučevanja

Razvoj računalniške in komunikacijske tehnologije poteka z neverjetno hitrostjo. Če se ozremo v ne tako daljno preteklost, lahko ugotovimo, da so prevladovali veliki, centralni, t.i. mainframe računalniki, do katerih so uporabniki dostopali prek terminalov. Ves ta ustroj velikih sistemov pa je zamajala iznajdba osebnega računalnika, ki po hitrosti in zmogljivosti ni prav nič zaostajal za velikim bratom. Toda kmalu je bilo tudi te računalnike treba združevati med seboj in nastala so omrežja ter omrežni strežniki. Arhitektura teh omrežij se je sčasoma spreminjala. Na začetku so bili strežniki, povezani v tako omrežje, le neke vrste datotečni strežniki, kjer so uporabniki shranjevali skupne podatke in aplikacije.

Nadaljnji razvoj je pripeljal do t.i. modela odjemalec / strežnik, kjer se je, vsaj kar se tiče poslovnih aplikacij, procesiranje podatkov zopet začelo prenašati na stran strežnika. Strežnik ni več služil le kot datotečni strežnik, temveč je skrbel do določene mere za urejanje, procesiranje in posredovanje podatkov odjemalcem. Večina poslovne logike se je še vedno izvajala na odjemalcih. To je pomenilo, da je bilo potrebno imeti dovolj zmogljivo strojno opremo na strani odjemalca, da so bili odzivni časi uporabe poslovnih aplikacij sprejemljivi, kar je bil vzrok, da take odjemalce imenujemo debele odjemalce (angl. fat clients).

Kasneje so se pojavljale številne izvedbe te arhitekture. Najbolj se je uveljavila t.i. 3-slojna arhitektura odjemalec / strežnik, kjer se je med podatkovnim strežnikom in odjemalcem dodal še t.i. aplikacijski strežnik, ki je skrbel za izvajanje aplikacijske logike. To je model, kjer se urejanje, obdelava in posredovanje podatkov izvaja na podatkovnih strežnikih, aplikacijska logika pa se izvaja na enem ali večih aplikacijskih strežnikih, medtem ko odjemalec skrbi večinoma za predstavitveni del (zaslonske slike) in določeno (omejeno) izvajanje aplikacijske logike. Take odjemalce imenujemo lahke odjemalce (angl. thin clients). Ena izmed prednosti takega modela se kaže predvsem v zmanjšanju stroškov lastništva (angl. Total Cost of Ownership - TCO), kajti odjemalce ni potrebno nenehno nadgrajevati zaradi premajhne procesorske moči, ker se večina obdelav opravi na strežnikih. Prav tako je upravljanje določenega informacijskega sistema centralizirano, kar zmanjšuje potrebe po strokovnih virih pri uporabnikih.

Področje proučevanja se nanaša torej na različne tehnologije lahkih odjemalcev ter možnosti njihove uporabe v današnjih informacijskih sistemih.

1.2 Namen

Namen magistrskega dela je proučiti in opisati tehnologijo lahkih odjemalcev, ki očitno predstavlja naslednji korak v razvoju računalniških in komunikacijskih tehnologij. Osredotočil sem bom na tehnologijo, ki jo je razvilo podjetje Citrix in ki jo je licenciral tudi Microsoft. Z možnostjo uvedbe take tehnologije lahko večja podjetja in organizacije, ki imajo dislocirane enote, prihranijo ogromno sredstev in pridobijo poslovne koristi. Prav to bom prikazal z analizo projekta Financial Times Palms - Asia, katerega namen je bil implementacija informacijskega sistema za distribucijo časopisa s tehnologijo lahkih odjemalcev.

1.3 Cilji

Glavni cilji magistrskega dela so:

- predstavitev razvoja računalniških arhitektur,
- predstavitev definicije lahkih odjemalcev,
- predstavitev različnih arhitektur lahkih odjemalcev,
- proučitev in predstavitev Citrix ICA in Microsoft TSE tehnologije,
- proučitev in predstavitev procesa implementacije Citrix ICA tehnologije,
- analiza implementacije in uporabe lahkih odjemalcev na podlagi rezultatov projekta Financial Times Palms – Asia.

1.4 Metodološki pristop

Naloge sem se lotil z raziskovanjem in proučevanjem tuje strokovne literature, člankov in informacij, dostopnih na internetu ter z analizo izsledkov projekta Financial Times Palms - Asia.

1.5 Zasnova poglavij

Magistrsko delo ima šest poglavij. Po uvodu najprej predstavim razvoj različnih računalniških arhitektur, ki so potrebne za razumevanje razlogov in ki so pripeljali do tehnologije lahkih odjemalcev ter njihove možnosti in omejitve.

V tretjem poglavju podam definicijo arhitekture lahkih odjemalcev in predstavim nekaj najbolj pomembnih tehnologij lahkih odjemalcev ter številne prednosti tehnologije lahkih odjemalcev in njihove slabosti.

V četrtem poglavju se posvetim podrobnemu opisu Citrix ICA tehnologije in Microsoft Terminal Services tehnologije lahkih odjemalcev. Podam razloge, ki so vodili do nastanka in opišem ponujene storitve, prednosti in seveda tudi slabosti. Čeprav v tehnične detajle ne nameravam globlje zahajati, ker bi to močno

presevalo okvir in namen magistrskega dela, se bom dotaknil tudi internih sprememb operacijskega sistema Windows NT, na katerem temeljita opisani tehnologiji. Podal bom tudi primerjavo obeh tehnologij in kriterije za njihovo izbiro.

V petem poglavju opisujem proces implementacije Citrixove ICA tehnologije lahkih odjemalcev na primeru projekta Financial Times Print and Logistic Management System – Asia (v nadaljevanju Palms - Asia). Namen projekta je bil priključitev dislociranih enot podjetja Financial Times v Hong Kongu in Tokiu na centralni informacijski sistem za upravljanje distribucije časopisa Financial Times, ki se nahaja v Londonu. Pri projektu sem osebno sodeloval v vlogi konzultanta z nalogo lokalizacije informacijskega sistema azijskim uporabnikom in prilagajanja programskega okolja večuporabniškemu načinu dela Citrix MetaFrame strežnika. Podrobneje opisujem, kaj je potrebno za uspešno izvedbo ter kakšna oprema in infrastruktura je potrebna. Prikazan je proces implementacije informacijskega sistema in analiza uspešnosti projekta ter pridobljene izkušnje in poslovne koristi.

V zaključku povzamem sklepne ugotovitve in predstavim svoje izkušnje uporabe lahkih odjemalcev.

2 RAZVOJ RAČUNALNIŠKIH ARHITEKTUR

Za boljše razumevanje arhitekture lahkih odjemalcev je dobro poznati različne računalniške arhitekture, ki so se razvile skozi zgodovino. Nove tehnologije se ponavadi razvijajo na podlagi starejših tehnologij in tako tudi tehnologija lahkih odjemalcev povzema svoje temelje iz različnih starejših računalniških arhitektur.

V tem poglavju bom prikazal razvoj računalniških arhitektur skozi zgodovino in njihov vpliv in vlogo pri nastanku lahkih odjemalcev.

2.1 Obdobje velikih računalnikov

Tehnologija lahkih odjemalcev predstavlja tretje poglavje v računalniški zgodovini. Ta se je začela z velikimi računalniki (angl. mainframes). Veliki računalniki so bili za takratne čase izredno zmogljive in drage naprave. Nabavni stroški velikega računalnika so lahko dosegali več milijonov dolarjev in ponavadi je vzdrževanje teh 'velikanov' presegalo nabavne stroške. Veliki računalniki so bili tudi zares veliki. Zasedali so veliko prostora, bili so potratni, potrebovali so zelo velik pretok zraka za hlajenje in elektrike jim ni bilo nikoli dovolj. Pri uporabnikih pa so prevladovali terminali, ki so bili povezani s posebnimi protokoli. Povezave niso bile posebej hitre. S podporo uporabnikom in osrednjemu računalniškemu sistemu se je ukvarjala velika skupina ljudi. Najpomembnejši so bili operaterji, ki so skrbeli za računalnik, menjavali trakove, diske, kartice in izpisovali gore papirja.

Arhitektura velikih računalnikov je bila sestavljena iz centralnega računalnika, ki je bil ponavadi v zavarovanem in klimatiziranem prostoru. Uporabniki so se povezovali s temi računalniki preko t.i. "neumnih" terminalov (angl. dumb terminals). Ponavadi so taki terminali imeli le enobarvni tekstovni ekran in tipkovnico ter povezavo do centralnega računalnika. Njihova funkcija je bila posredovanje vhodnih signalov (tipkovnice) centralnemu računalniku in prikaz izhodnih signalov (tekst) centralnega računalnika. Zaradi tehnološke enostavnosti so bili takšni terminali razmeroma poceni, ponavadi manj kot 500 ameriških dolarjev (Cumhur, 2000, str.9). Danes veliko uporabnikov dostopa do velikih računalnikov s pomočjo osebnega računalnika, na katerem se izvaja emulator "neumnega" terminala.

Število velikih računalnikov, ki so v uporabi po celem svetu je še vedno visoko, kar je še posebej prišlo do izraza ob prehodu v novo tisočletje, ko je bilo potrebno odpraviti t.i. "tisočletni hrošč". Posledično je visoko tudi število "neumnih" terminalov in osebnih računalnikov s primarno vlogo emulatorja "neumnega" terminala.

Veliki računalniki še danes igrajo pomembno vlogo v velikih korporacijah predvsem zaradi naslednjih razlogov (Cumhur, 2000, str 8.) :

- Možno je doseči izredno veliko procesorsko moč s samo enim multiprocesnim sistemom.
- Zagotovljena je velika zanesljivost sistema s pomočjo povezovanja več računalnikov v gručice (angl. clustering).
- Primernost za izvajanje poslovno kritičnih transakcij (predvsem finančne transakcije).

Prednost arhitekture velikih računalnikov se kaže v centraliziranosti sistema, saj celoten sistem sloni na enem velikem računalniku, kar bistveno olajša upravljanje in vzdrževanje. Prav tako je varnost pri takšnem sistemu večja, kajti sam računalnik je fizično zavarovan, pa tudi uporabniki ne morejo ogroziti sistema kot npr. z virusi. V tabeli 1 so povzete prednosti in slabosti velikih računalnikov.

Tabela 1: Prednosti in slabosti velikih centralnih računalnikov

PREDNOSTI	SLABOSTI
Možnost razširitve	Tekstovni uporabniški vmesnik v aplikacijah
Visoka zanesljivost centralnega sistema	Visoki začetni stroški
Visoka zanesljivost terminalov	Visoka odvisnost od enega samega dobavitelja
Centralizirane varnostne kopije	Visoka odvisnost od enega samega računalnika (angl. Single point of failure)
Centralizirano upravljanje sistema	
Nizki stroški terminalov in s tem cena na uporabnika	
Visoka stopnja varnosti	

Vir: Cumhur, 2000, str.10

2.2 Obdobje osebnih računalnikov

V osemdesetih letih so osebni računalniki z naraščajočo procesorsko močjo in padajočimi cenami osvojili računalniški trg. Koncept grafičnega uporabniškega vmesnika je prinesel na tisoče novih aplikacij - tabelarni programi, tekstovni procesorji, rešitve za računalniško podprto načrtovanje (angl. CAD - Computer Aided Design) ipd. Takšni programi so močno dvignili standard uporabniških aplikacij glede prijaznosti in produktivnosti. Uporabniki niso več bili vezani na centralni računalnik in so lahko večino funkcij opravljali na svojem osebem računalniku.

Žal so osebni računalniki poleg večje prijaznosti in produktivnosti prinesli tudi večje stroške upravljanja in vzdrževanja, kajti vsak osebni računalnik je bilo potrebno vzdrževati in nadgrajevati posebej, kar je stroške upravljanja in vzdrževanja vedno bolj povečevalo. Velike korporacije lahko porabijo tudi več let za nadgrajevanje določenih različic programske opreme, saj imajo opravka z več tisoč računalniki. V tabeli 2 so prikazane glavne prednosti in slabosti osebnih računalnikov.

Tabela 2: Temeljne prednosti in slabosti osebnih računalnikov

PREDNOSTI	SLABOSTI
Standardna strojna oprema	Dražji od "neumnih" terminalov
Standarden operacijski sistem	Ni centraliziranih varnostnih kopij
Možnost razširljivosti	Ni centraliziranega upravljanja
Grafičen uporabniški vmesnik	Visoki stroški upravljanja in vzdrževanja
Razmeroma nizki nabavni stroški (v primerjavi z velikimi računalniki)	
Visoka produktivnost in prijaznost	

Vir: Cumhur, 2000, str.12

2.3 Arhitektura odjemalca / strežnika

Prihod interneta je v svetu osebnih računalnikov vzpodbudil nastanek lokalnih omrežij (angl. Local Area Network - LAN), pretežno z operacijskim sistemom Novell NetWare. Osebnih računalnikov niso bili več samostojne delovne postaje, ampak so se začeli povezovati v omrežja. Na začetku so bili strežniki, povezani v tako omrežje, le neke vrste datotečni strežniki, kjer so uporabniki shranjevali skupne podatke in aplikacije.

Kasneje se je tudi procesiranje podatkov začelo prenašati na strežnik, ampak večina poslovne logike se je še vedno odvijala na odjemalcih. Tako je nastal nov model v računalništvu: odjemalec / strežnik (angl. client / server).

Arhitekturo odjemalca / strežnika lahko definiramo kot (Norris, 1997, str. 273):

"Delitev določene aplikacije na (vsaj) dva dela, pri katerem en deluje kot odjemalec (zahteva storitve) in drugi kot strežnik (nudi storitve)."

Ker se je večina procesiranja še vedno izvajala na odjemalcih, je bilo potrebno imeti dovolj zmogljivo strojno opremo na strani odjemalca, da so bili odzivni časi sprejemljivi. To je bil tudi vzrok, da se je sčasoma v informacijski industriji teh odjemalcev in strežnikov oprijel opis debeli odjemalec (angl. fat clients) oz. lahki strežnik (angl. thin server) (Dewire, 1998, str 31).

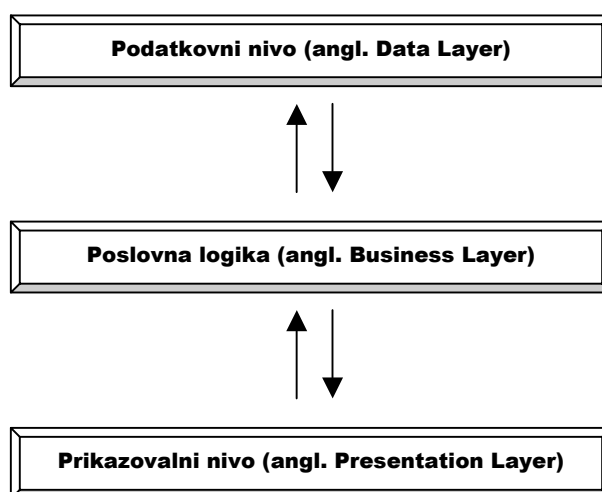
Model odjemalca / strežnika je bil korak naprej v svetu osebnih računalnikov, prinesel pa je tudi nove komplikacije. Na strani odjemalca ponavadi najdemo Intelx86 osebni računalnik z Windows operacijskim sistemom, ki sprva ni bil načrtovan za omrežja. V primerjavi z velikimi računalniki osebni računalniki niso bili tako zanesljivi, njihovo upravljanje in varovanje ni bilo možno centralizirati in

njihove uporabnike je bilo potrebno posebej izšolati za pravilno uporabo osebnega računalnika. Tudi distribucija novih aplikacij je bila časovno potratna, saj je bilo potrebno programske pakete namestiti na vsak odjemalec oz. osebni računalnik posebej.

2.4 Večslojne arhitekture odjemalec / strežnik

Prihod prostranih omrežij (angl. Wide Area NetWork - WAN), Interneta in Intraneta je tvoril kup med sabo povezanih nekompatibilnih sistemov. Z namenom, da bi lahko uporabniki z različnimi operacijskimi sistemi uporabljali iste aplikacije, se je razvila večslojna arhitektura odjemalec / strežnik. Pri tem se določena aplikacija deli na tri ali več sloje, kot jih prikazuje slika 1.

Slika 1: Večslojna arhitektura odjemalec / strežnik



Vir: Kanter, 1998, str. 32

Vsak nivo je neodvisen od drugega in lahko uporablja poljuben operacijski sistem. Podatkovni nivo skrbi za upravljanje podatkov, shranjevanje in povpraševanje po njih. V nivoju poslovne logike lahko najdemo celotno aplikacijsko logiko, ki definira, kako se aplikacija odziva na določene zahteve uporabnika. Prikazovalni nivo naj bi se nahajal na strani odjemalca in bi skrbel za interakcijo z uporabnikom in naj bi vseboval le najnujnejšo programsko logiko, da bi lahko to izpolnil. Namen te strukture je čim enostavnejša in hitrejša prenosljivost oz. implementacija prikazovalnega nivoja iz enega okolja v drugo.

Dober primer večslojne arhitekture odjemalec / strežnik so aplikacije na internetu oz. spletne aplikacije. Prikazovalni nivo je tukaj spletni pregledovalnik, nivo poslovne logike so lahko skripte na spletnih strežnikih ali pa bolj kompleksne komponente kot npr. javanska zrnca, podatkovni nivo pa je lahko relacijska podatkovna baza, s katero komunicirajo zgoraj omenjene skripte oz. komponente.

Kot lahko vidimo, je platforma prikazovalnega nivoja popolnoma neodvisna od platforme nivoja poslovne logike in prikazovalnega nivoja.

Čeprav ima takšen pristop veliko prednosti, pa na drugi strani zahteva popolnoma drugačen pristop pri razvoju aplikacij. To je seveda povezano z velikimi stroški, še posebej, če gre za velike sisteme, tako da še danes srečamo veliko aplikacij, ki so razvite po modelu debeli odjemalec / lahki strežnik.

3 LAHKI ODJEMALCI

3.1 Koncept lahkih odjemalcev

Koncept lahkih odjemalcev združuje prednosti osebnih računalnikov in velikih računalnikov. Omogoča centralizirano upravljanje in vzdrževanje, kot ga poznamo pri velikih računalnikih ter hkrati večjo produktivnost in prijaznost, ki ga prinašajo grafični uporabniški vmesniki osebnih računalnikov.

Arhitektura lahkih odjemalcev razdeli aplikacijsko logiko tako, da se poslovna logika aplikacije izvaja na zmogljivem strežniku, medtem ko bi se naj s prikazovanjem uporabniškega vmesnika ukvarjal samo odjemalec. Lahke odjemalce bi lahko definirali kot (Regel, 1999, str.5):

"Koncept lahkih odjemalcev je omrežna arhitektura, ki temelji na več slojni odjemalec / strežnik arhitekturi, kjer odjemalec nudi le najnujnejšo funkcionalnost za interakcijo s strežnikom."

V poslovnem svetu obstaja več različnih tehnologij lahkih odjemalcev, med katerimi so najbolj razširjene naslednje:

- Windows terminali (angl. Windows Based Terminals),
- omrežni računalniki (angl. Network computers),
- omrežni osebni računalnik (angl. Net PC).

Treba je poudariti, da razen Windows Terminalov ostale našteje različice ne ustrezajo zgoraj navedeni definiciji lahkih odjemalcev v popolnosti, kar bo razvidno v nadaljevanju.

3.1.1 Windows Terminali

Cilj velikih organizacij je, najti način, kako zamenjati starejše "neumne" terminale z novimi napravami, ki bi jim omogočale uporabo dosedanjega informacijskega sistema. Eden od načinov je uporaba emulatorja "neumnega" terminala na osebem računalniku. Ker pa je osebni računalnik dokaj zmogljiva naprava, bi to

bilo nepotrebno zapravljanje virov. Alternativa osebnemu računalniku je Windows Terminal (angl. Windows based Terminal).

Windows terminali predstavljajo grafičen naslednik tekstovnemu "neumnemu" terminalu. To so lahko X-Terminali ali pa Windows Terminali, ki temeljijo na Microsoft RDP oz. Citrix ICA protokolu, ki ga bom podrobno predstavil kasneje. Oba tipa terminalov se uporabljata le za vnos in prikaz podatkov, aplikacijska logika pa se izvaja na strežniku. Razlikuje pa se sama implementacija, saj gre za dva popolnoma različna operacijska sistema.

Protokol za X Windows, imenovan X oziroma X11, je bil razvit leta 1986 na Institute of Technology (MIT) v Massachusettsu. Postal je de facto standard za okenski sistem na vseh vodilnih Unix delovnih postajah. Arhitektura X Windows razdeli aplikacijsko logiko in prikazovalni nivo, tako da je možno uporabljati različne operacijske sisteme na strežniku in odjemalcu.

Študije so pokazale, da povprečen X-Windows Terminal porabi 624-736Kps pasovne širine omrežja (Mann, 1999, str. 51). Prav tako naj bi povprečen X-Windows terminal zahteval 10-20 megabajtov pomnilnika in procesorsko moč, enako najmanj Intel 486 procesorju.

Citrixov ICA (angl. Independent Computing Architecture) protokol je bil razvit za uporabo Microsoft Windows aplikacij na oddaljenih strežnikih. ICA ima zelo majhne zahteve glede procesorske moči odjemalca in omrežne pasovne širine, ki je potrebna do strežnika. Tako je velikim organizacijam omogočena uporaba najnovejših Windows aplikacij na starejših osebnih računalnikih, če imajo procesorsko moč enako najmanj Intel 286 procesorju in pasovno širino 20 kb/s do strežnika. Prav tako kot pri X protokolu je tudi pri ICA protokolu operacijski sistem strežnika neodvisen od odjemalca.

Medtem ko je potrebno aplikacije za X-Windows protokol posebej razvijati, Citrixov ICA omogoča uporabo vseh aplikacij, ki si so bile razvite za Microsoft Windows operacijski sistem, kar je seveda pglavitna prednost te tehnologije. X-Windows in ICA predstavljata pravo implementacijo lahkih odjemalcev, saj se izvajanje aplikacijske logike odvija 100 odstotno na strežniku.

3.1.2 Omrežni računalnik

Leta 1995 je konzorcij računalniških podjetij (Oracle, Sun, IBM) z izvršilnim direktorjem Oracla Larrijem Ellisonom na čelu, predstavil svojo vizijo prihodnosti računalništva v obliki omrežnega računalnika. Omrežni računalnik naj bi se prilagodil NCref1 standardu (NC-1). Standard NC-1 je predstavljal popolnoma novo arhitekturo, ki uporablja javanski operacijski sistem in prestavi vso

aplikacijsko logiko na strežnik. Takšni terminali naj bi operacijski sistem, brskalnik in programske rešitve naložili iz strežnika ob zagonu. Imeli naj bi lokalno navidezno javansko mašino (angl. Java Virtual Mashine), tako da bi se vse aplikacije izvajale v Javi, programskem jeziku ki ga je razvilo podjetje Sun Microsystems. Imeli bi lokalni pomnilnik RAM in nobenega trajnega pomnilnika. Komunikacija do strežnika pa bi potekala po splošno razširjenem transportnem protokolu TCP/IP.

Prednost omrežnega računalnika je v nižjih stroških strojne opreme, kajti terminali naj ne bi imeli lokalnega diska, procesorji pa bi lahko bili iz Risc družine, ki predstavlja cenejšo alternativo procesorjem podjetja Intel.

Prav tako bi bila implementacija in vzdrževanje novih aplikacij enostavnejša, kajti omrežni računalnik ob vsakem zagonu ponovno naloži programsko opremo iz strežnika, tako da je potrebno spremembe izvesti le na strežniku.

Velika slabost koncepta omrežnih računalnikov je pa seveda pomanjkanje kakršnihkoli programskih rešitev, kajti vse programe je treba razviti ponovno v skladu s specifikacijami NCR¹ in v programskem jeziku Java. Druga slabost je zahteva po večji pasovni širini v omrežju, kajti omrežni računalniki bi že po svoji naravi vsekakor povečali promet na lokalnem omrežju.

Koncept omrežnega računalnika ni doživel posebej velikega razmaha, saj je podcenjeval priljubljenost operacijskega sistema Windows in predvideval, da bo programski jezik Java zamenjal aplikacije napisane za Windows okolje. Prav tako je razvidno, da omrežni računalnik že odstopa od koncepta lahkih odjemalcev, saj uvaja delno procesiranje na odjemalcu (Kanter, 1998, str 83).

3.1.3 Omrežni osebni računalnik

Koncept omrežnega osebnega računalnika (angl. Net PC) sta v sodelovanju razvili podjetji Microsoft in Intel. Omrežni osebni računalnik je bil odgovor Microsofta na Oraclov omrežni računalnik. Razlika med omrežnim in klasičnim osebnim računalnikom naj bi bila predvsem v administraciji, ki naj bi bila centralizirana. Operacijski sistem omrežnih osebnih računalnikov je Microsoft Windows, kar zmanjšuje stroške šolanja uporabnikov in omogoča uporabo široke palete že razvitih programskih rešitev.

Da bi bila administracija osebnih računalnikov čim lažja, je Microsoft razvil t.i. ZAW (Zero Administration for Windows) tehnologijo. Ključne komponente te tehnologije, ki bi naj centralizirala administracijo, so naslednje (Cole, 1999, 168 str.):

- **Samodejna konfiguracija** (angl. Auto Configuration). Odjemalec samodejno posodobi nastavitve iz strežnika.

- **Samodejno namizje** (angl. Automatic Desktop). Samodejna namestitvev aplikacij na odjemalcih.
- **Samodejno shranjevanje sistemskega stanja** (angl. Automatic System State Storage). Vsi podatki o stanju odjemalca se samodejno shranjujejo na strežniku.
- **Centralizirano upravljanje** (angl. Central Administration Lockdown). Dostop do programske in strojne opreme odjemalcev je centraliziran in omogočen le administratorju omrežja.
- **Poljubnost aplikacij** (angl. Application flexibility). Podpora Windows in javanskim aplikacijam.

Kot je bilo omenjeno pri centraliziranem upravljanju, naj omrežni osebni računalnik ne bi imel disketne enote. Prav tako naj bi bil fizično zaprt, tako da uporabniki ne bi imeli nobene možnosti konfiguriranja oz. nadgrajevanja računalnika brez vednosti administratorja.

Stroški vzdrževanja klasičnega omrežja osebnih računalnikov naj bi se z uporabo ZAW tehnologije zmanjšali do 15% (Halfhill, 1997, str. 80). Razvidno je, da omrežni osebni računalnik še vedno deluje kot debeli odjemalci, vendar je njihovo upravljanje centralizirano, kar je ena izmed temeljnih prednosti koncepta lahkih odjemalcev.

3.2 Vodilna podjetja na področju lahkih odjemalcev

Trg lahkih odjemalcev je velik in zato mnoga podjetja poskušajo nuditi različne storitve na tem področju. Ta podjetja lahko razdelimo v dve skupini. Prva se ukvarja z izdelavo strojne opreme, druga pa z izdelavo programske opreme. Pri tem je jasno, da se proizvajalci strojne opreme morajo prilagoditi proizvajalcem programske opreme, ki postavljajo standarde in protokole. Med najbolj znane proizvajalce Windows terminalov in omrežnih osebnih računalnikov spadajo ameriška podjetja HP - Compaq, Network Computing Devices in Wyse Technology. Bistvene lastnosti teh terminalov so nizka cena in majhne zahteve po vzdrževanju. Njihovi terminali pa podpirajo protokol X-11 za X – Windows ter RDP in ICA protokol za Windows Terminal Server oz. Citrix MetaFrame strežnik.

Med ponudniki rešitev, ki zajemajo programsko opremo, sta najbolj razširjeni podjetji Citrix in Microsoft, omeniti pa velja še podjetje SCO z izdelkom Tarantela. Največ zanimanja je za izdelke Citrixa in Microsofta. Microsoft ponuja dva izdelka,

MS Terminal Server 4.0, ki temelji na tehnologiji NT, in Windows 2000 Advanced Server, v katerem je terminalska tehnologija že vgrajena in podprta z terminalskimi servisi. Microsoft je osnovno tehnologijo za terminalski strežnik licenciral od Citrixa, zato ne preseneča, da so Citrixovi izdelki sorodni Microsoftovim. Citrix ponuja strežnik Metaframe 1.8 za Microsoft Terminal Server 4.0 in MetaFrame 1.8 ter MetaFrame 2000/XP za operacijski sistem Windows 2000. Citrixovi izdelki uporabljajo učinkovitejši protokol za komunikacijo med strežnikom in odjemalcem, zato bolje izrabljajo pasovno širino kot Microsoftovi izdelki.

Citrixovi in Microsoftovi produkti lahkih odjemalcev so v poslovnem svetu uveljavljeni ter postali standard za tehnologijo lahkih odjemalcev, saj jih uporablja 80% podjetij iz seznama Fortune 500 (Citrix, 2002).

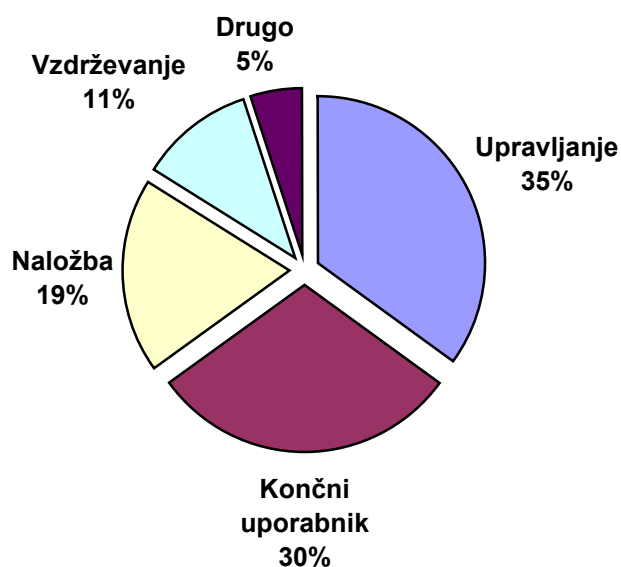
3.3 Prednosti lahkih odjemalcev z vidika skupnih stroškov lastništva

Široka uporaba lahkih odjemalcev v poslovnem svetu ima zelo preprost namen: prihraniti denar. Lahki odjemalci naj bi občutno zmanjšali skupne stroške lastništva (angl. TCO – Total Cost Ownership). V nadaljevanju bom opisal koncept skupnih stroškov lastništva in vpliv lahkih odjemalcev na njih.

Analizo skupnih stroškov lastništva je prvi predstavil Gartner Group leta 1987. Od takrat naprej se pojavlja v različnih metodologijah. Raziskave Gartner Groupa so pokazale, da v večini primerov stroški informacijskih sistemov ne zajemajo le nabavne stroške strojne in programske opreme. Skupni stroški lastništva informacijskega sistema zajemajo vlaganje v strojno in programsko opremo, upravljanje, strokovno, organizacijsko in upravljavsko osebje ter stroške, ki nam jih povzročajo končni uporabniki in poslovno škodo izpada sistema. Koliko zares stane določeno vlaganje, se pokaže šele čez daljši čas. Analize so pokazale, da pomeni naložba le slabo petino stroškov informacijskega sistema. Najdražje je upravljanje, saj obsega kar 35% omenjene vrednosti. V te stroške so všteti tudi stroški za strokovni kader, ki skrbi za sistem (Garnter Group, 2000).

Poleg načrtovanih stroškov so tu še skriti stroški, ki nastanejo med uporabo informacijskega sistema. Sem štejemo formalna in neformalna učenja, pomoč, ki si jo uporabniki posredujejo med seboj ter samopomoč in razvoj lastnih programov oziroma rešitev. Študije Gartner group so pokazale, da sestavljajo skriti stroški, ki jih povzročajo uporabniki skupaj s stroški načrtovanega in nenačrtovanega izpada sistema, kar 55% vseh stroškov lastništva in ne le 30% kot so predvidevali doslej.

Slika 2: Struktura stroškov v skupnih stroških lastništva



Vir: Gartner Group, 2000

Različne raziskave so skušale odgovoriti, kako uporaba različne tehnologije vpliva na skupne stroške informacijskega sistema. Raziskava skupine The Tolly Group je pokazala, da je treba pravzaprav razmišljati o skupnih stroških lastništva aplikacije (Tolly Group, Atur, 1999), kajti pomembno je, ali je aplikacija nameščena na strežniku ali na odjemalcu. Prav to ima velik vpliv na kompleksnost sistema in čas namestitve aplikacije v okolje. Pomembno je tudi, ali se aplikacija izvaja na strežniku, odjemalcu ali v večslojnim načinu. Če se izvaja na odjemalcu, mora biti odjemalec dovolj zmogljiv. Če se deli aplikacije dinamično pretakajo na odjemalce in izvajajo na njih, pa lahko naletimo na ozka grla v omrežju.

Tudi fizična lokacija podatkov vpliva na stroške. Če so vsi podatki shranjeni na enem mestu, so hitreje dosegljivi, veliko pa lahko prihranimo tudi pri lažjem, hitrejšem in manj zapletenem načinu izdelave varnostnih kopij naših podatkov. Velike težave so z zagotavljanjem dostopa do različnih aplikacij oddaljenim uporabnikom ali uporabnikom, ki so veliko na poti. Takim uporabnikom dostop s katerimkoli odjemalcem po različnih povezavah, najbolje kar po internetu, prihrani veliko časa in jim poveča produktivnost.

Iz navedenega je razvidno, da lahki odjemalci zmanjšajo skupne stroške lastništva informacijskega sistema. To se doseže v izredno povečani odzivnosti na zahteve po spremembah ali novi programski opre: Ker se programska oprema namešča samo enkrat, lahko merimo čas do razpoložljivosti nove programske opreme uporabnikom ne več v tednih, ampak v minutah, ne glede na to, koliko je potencialnih uporabnikov. Stroški administracije in vzdrževanja ter podpore programske opreme pri uporabnikih so precej nižji. V primeru Windows Terminalov

se programska oprema namešča in izvaja izključno na strežnikih, uporabnik pa k sebi dobiva le sliko. Prav tako je zmanjšana potreba po neprestanih nadgradnjah strojne opreme, kajti nadgrajevati je potrebno le strežnik.

4 CITRIX ICA IN MICROSOFT TSE TEHNOLOGIJA LAHKIH ODJEMALCEV

4.1 Citrix

V tem poglavju se posvetim podrobnemu opisu Citrix ICA tehnologije lahkih odjemalcev. Opisana bo ključna tehnološka rešitev, ki omogoča strežbo aplikacij lahkim odjemalcem, t.i. Independent Computing Architecture (ICA). Razvilo jo je podjetje Citrix Systems iz Fort Lauderdale na Floridi in je vodilno na področju rešitev lahkih odjemalcev.

4.1.1 Nastanek Citrix ICA / MetaFrame tehnologije

Vizija ustanovitelja in predsednika uprave podjetja Citrix, Eda Iacobucci, je bila izvajanje iste aplikacije na računalnikih z različnimi operacijskimi sistemi in viri. Ko je opravljal funkcijo vodje razvoja na skupnem projektu OS/2 Microsofta in IBMa, je poskušal svojo vizijo uresničiti, toda nobeno od omenjenih podjetij takšen razvoj ni zanimal.

Tako je Iacobucci leta 1989 ustanovil podjetje Citrix Systems in začel razvijati večuporabniško jedro MultiWin, ki je temeljilo na operacijskem sistemu OS/2. MultiWin je omogočal več hkratnih sej in namizij na operacijskem sistemu OS/2, ki so se izvajale v zaščitениh pomnilniških prostorih za vsakega uporabnika posebej. Leta 1996 je podjetje Citrix začelo prodajati WinWiew, prvi večuporabniški OS/2 operacijski sistem (Kaplan, 2000). Produkt WinWiew je temeljil na MultiWin tehnologiji in je vseboval prvo verzijo ICA (angl. Independent Computing Architecture) protokola. Čeprav so takrat prevladovali arhitekture debeli odjemalec / strežnik, so ljudje podjetja Citrix verjeli v arhitekturo lahki odjemalec / strežnik in večuporabniški operacijski sistem.

Po zatonu operacijskega sistema OS/2 je Iacobucci uspel prepričati Microsoft, da bo povpraševanje po večuporabniškem operacijskem sistemu Windows NT obstajalo. Tako je podjetje Citrix pridobilo licenco za izdelavo večuporabniškega jedra, temelječega na operacijskem sistemu Microsoft Windows NT. Licenčna pogodba je vsekakor bila "win-win" razmerje za podjetje Microsoft, kajti v primeru da je vizija Citrix-a prava, bo Microsoft imel rešitev za morebiten večuporabniški trg.

Podjetje Citrix je objavilo produkt WinFrame, ki je bil kombinacija operacijskega sistema Windows NT 3.51 Server in MultiWin tehnologije. WinFrame je predstavljal napredek v primerjavi s prejšnjim produktom WinWiew, ki je temeljil na operacijskem sistemu OS/2. Ob času objave je operacijski sistem Windows 3.1 in kasneje Windows 95 postajal standard z velikim naborom programskih rešitev in WinFrame je predstavljal rešitev za oddaljeno izvajanje aplikacij, pisanih za operacijski sistem Windows.

WinFrame strežniki so izvajali Windows aplikacije, uporabniki pa so jih uporabljali z različnimi odjemalci prek omrežnih povezav. Ključni element WinFrame tehnologije je bil prenosni protokol, ki je razdelil aplikacijsko logiko in uporabniški vmesnik. Preko omrežja so potovali le signali tipkovnice, miške in podatki za obnovitev ekrana. ICA protokol je uporabnikom dal občutek, da delajo z lokalno nameščeno aplikacijo, čeprav se je le-ta izvajala na oddaljenem strežniku.

Povečano zanimanje za Metaframe je navdušilo Microsoft, da licencira MultiWin tehnologijo od Citrixa. Leta 1997 je Microsoft integriral to tehnologijo v operacijski sistem NT 4.0 in produkt poimenoval Terminal Services Edition (TSE). Toda TSE je namesto prenosnega protokola ICA uporabljal Microsoftov protokol RDP, ki pa ne omogoča vseh funkcionalnosti, ki jih nudi Citrixov ICA protokol.

Citrixov odgovor na TSE se je imenoval MetaFrame. To je bil hkrati naslednik WinFrame-a. MetaFrame predstavlja dodatek (angl. add-on) k Microsoft TSE tehnologiji, medtem ko je WinFrame bil samostojen operacijski sistem. Po objavi leta 1999 je razvoj MetaFrame-a stalno tekel naprej, tako da deluje z vsemi Microsoftovimi produkti, ki vsebujejo TSE tehnologijo (Kaplan, 2000):

- Windows NT Server 4.0,
- Windows NT 4.0 Terminal Server Edition,
- Windows 2000 Advanced Server.

Osnovne administrativne funkcije TSE tehnologije so omejene na upravljanje omrežja in uporabniških sej. MetaFrame med drugim dodaja možnost upravljanja z resursi tekočih aplikacij, upravljanja zasedenosti strežnika in izboljšano upravljanje sej. Aplikacije se lahko objavljajo na internetu oz. intranetu, tako da se jih da uporabljati prek spletnega pregledovalnika. Več aplikacij skupaj pa lahko predstavimo tudi kot spletni aplikacijski portal. Dostop do MetaFrame je omogočen velikemu naboru odjemalcev iz različnih omrežij in operacijskih sistemov.

Microsoft TSE tehnologija omogoča omejeno uporabo prednosti tehnologije lahkih odjemalcev preko RDP protokola. Z dodatkom Citrix MetaFrame pa nam je omogočena uporaba ICA protokola in vseh prednosti povezanih z njim. Razlike in lastnosti teh dveh tehnologij bom opisal v tem poglavju.

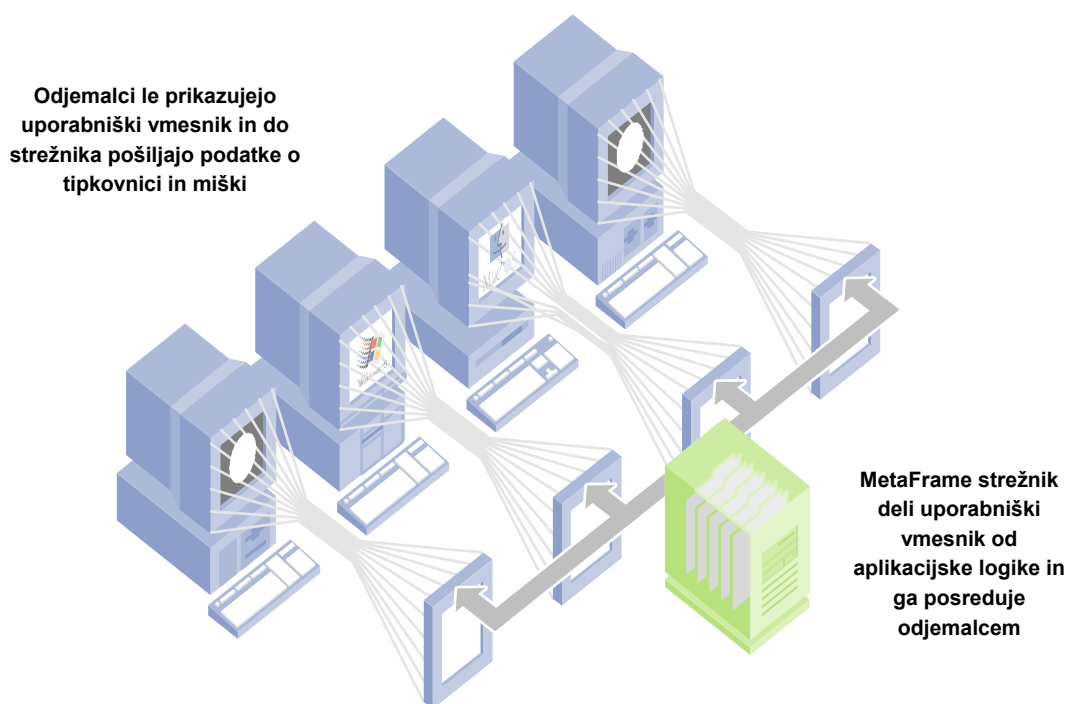
4.1.2 Predstavitev Citrix Metaframe

Citrix Metaframe bi lahko na kratko in malo ironično opisali kot tehnologijo velikih centralnih računalnikov z grafičnim uporabniškim vmesnikom.

Dejansko imamo strežnik (ali tudi več njih povezanih v gručo), na katerega se priklopijo odjemalci oziroma t.i. terminali. Celotno procesiranje oziroma izvajanje aplikacij poteka na strežniku, do odjemalcev prihajajo le ekranske slike, v nasprotno smer pa potujejo podatki o pritiskih na tipkovnico, premiki miške ali podatki iz kake druge vhodne naprave na odjemalcu. Odjemalci se lahko na strežnik povežejo preko kakršnegakoli LAN ali WAN omrežja ali pa celo preko direktnih klicnih linij.

Kot strežnik nam lahko služi že malo močnejši osebni računalnik z Intel Pentium procesorjem. Na strani odjemalca pa imamo lahko zelo različne naprave, od osebnih računalnikov z DOS ali katerikoli od Windows operacijskim sistemom, Windows terminalov, omrežnih računalnikov, Apple-vih Macintosh-ov, Unix delovnih postaj in katerikoli drugo napravo, ki lahko poganja spletni pregledovalnik.

Slika 3: Citrix tehnologija lahkega odjemalca



Vir: Kanter, 1998, str. 15

Kot je že omenjeno v uvodu, sloni celotna tehnologija na Windows NT operacijskem sistemu, ki so mu v podjetju Citrix dodali t.i. MultiWin jedro, ki bo podrobneje opisano kasneje.

Za implementacijo te tehnologije potrebujemo Windows NT 4.0 Terminal Server Edition oz. Windows 2000 Advanced Server Edition, na katerega dodatno namestimo še Citrixov Metaframe dodatek.

Citrix Metaframe omogoča, da se vsakemu uporabniku, ki se prijavi na strežnik, dodeli svojo, zaščiteno sejo na tem strežniku. Tako lahko navaden, na Intel Pentium II procesorju sloneč strežnik služi za hkratno delo nekaj 10 uporabnikov. To število je seveda odvisno od aplikacij, katere posamezni uporabniki izvajajo (Kanter, 1998).

Dostop do takega strežnika je mogoč preko kakršnegakoli LAN ali WAN omrežja, ter preko direktnih klicnih linij. Na strani odjemalca imamo lahko zelo različne naprave. Najobičajnejši so navadni osebni računalniki. Uporabimo lahko celo star Intel 286 računalnik, na katerem teče DOS operacijski sistem in ima vsaj VGA grafično kartico in 1MB dinamičnega pomnilnika (RAM). Nanj naložimo odjemalca (program ki se naloži z ene diskete) in že se lahko priključimo na strežnik, kjer lahko zaganjamo najnovejše Windows aplikacije.

Prav tako so podprti vsi drugi osebni računalniki z vsemi verzijami Windows operacijskega sistema, Applovi Macintoshi, računalniki, na katerih teče Unix operacijski sistem, Windows Terminali, omrežni računalniki, brezžične naprave in ostale informacijske naprave, ki imajo vgrajeno podporo za ICA tehnologijo.

Med podjetji, ki so že licencirala ICA tehnologijo in izdelujejo različne informacijske naprave z vgrajeno podporo za to tehnologijo, so tudi velikani kot IBM, Sun Microsystems, HP, Sharp, NCD, Wyse technology, Acron in drugi (Anderson, 2002).

Za uvedbo MetaFrame tehnologije lahkega odjemalca lahko torej uporabimo že obstoječo infrastrukturo in obstoječe delovne postaje (tudi tiste, ki so bile zrele za odpis). Če pa informacijsko infrastrukturo gradimo popolnoma na novo, je smiselno razmišljati o Windows terminalih. To so, kot je opisano v tretjem poglavju, enostavni terminali, ki ne skrbijo za nič drugega kot za prikaz slike in komunikacijo s strežnikom. Vso potrebno programsko opremo imajo že vgrajeno (navadno zapisano na Flash ROMu), poleg procesorja in grafične kartice imajo samo še mrežno kartico in serijska ter paralelna vrata ter priključek za tipkovnico in miško.

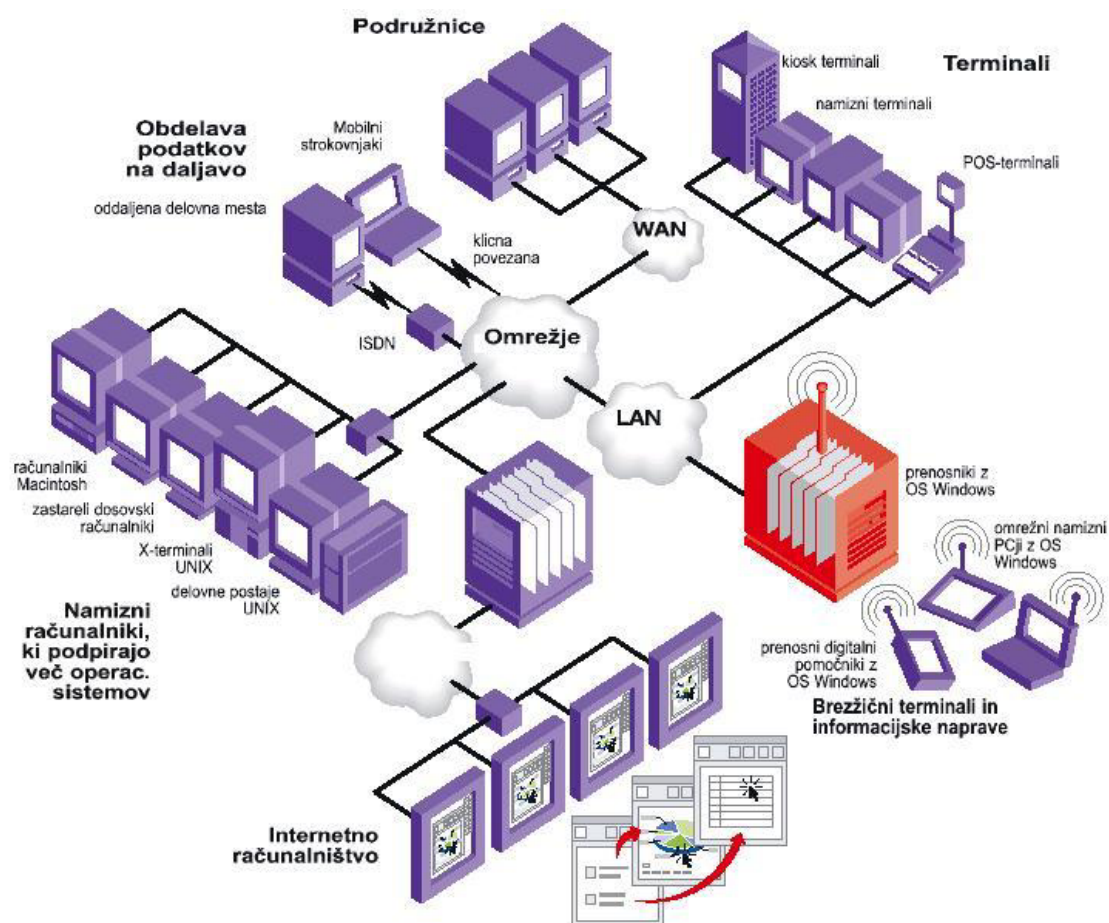
Značilnosti in hkrati prednosti, ki jih nudi Citrix Metaframe tehnologija, je kar nekaj. Med najbolj izstopajočimi lahko navedemo naslednje (Citrix, 2002):

- Centralizirano upravljanje in s tem povezano zmanjšanje celotnih stroškov. Kot sem že omenil, je ravno upravljanje in vzdrževanje velike množice osebnih

računalnikov zelo zahtevno in predvsem drago opravilo. S pomočjo tehnologije lahkega odjemalca, se vračamo k centraliziranemu upravljanju, kjer lahko vzdrževalec vse posege izvede na samo enem mestu in to celo iz oddaljene lokacije. Odjemalci pri tej tehnologiji so podobni terminalom (ali pa to celo so) in ne zahtevajo praktično nobenega vzdrževanja. Novo verzijo katerekoli aplikacije (npr. Word, Excel, poslovno aplikacijo ...) namestimo samo na enem mestu in takoj imajo vsi uporabniki dostop do te najnovejše verzije.

- Varnost. Kot je že omenjeno, pri tej tehnologiji po omrežju od strežnika do odjemalca potuje le ekranska slika, tako da podatki nikoli ne zapustijo strežnika in ne potujejo po omrežju, kot je primer pri standardnih odjemalcih / strežnikih tehnologijah. Poleg tega, da podatki nikoli ne zapustijo strežnika, je lahko tudi ekranska slika, ki se prenaša, zakodirana (lahko tudi s 128 bitnim ključem), tako da se možnost za prestrzanje kakršnihkoli podatkov s strani nepooblaščenih oseb zmanjša na minimum.
- Povezovanje heterogenih računalniških okolij. Omogočeno je dokaj enostavno povezovanje vseh obstoječih, velikokrat heterogenih računalniških sistemov in omrežij. Podprti so praktično vsi bolj razširjeni protokoli: TCP/IP, IPX, SPX, NetBIOS in direktne asinhronne povezave. Uporabniki se na strežnik lahko povežejo preko navadnih komutiranih telefonskih linij, ISDN-a, Frame relay, ATM omrežja, WAN povezav (T1, T3, X.25 ipd.), brezžičnih povezav ter seveda preko interneta in intranetov.

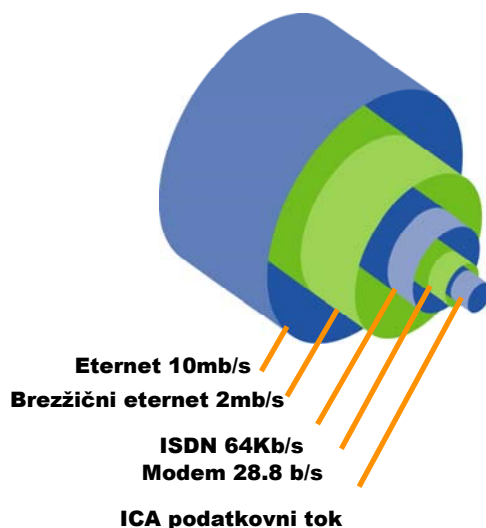
Slika 4: Povezovanje heterogenih sistemov



Vir: Citrix, 2002

- Dostop iz oddaljenih lokacij. Protokol, na katerem deluje MetaFrame tehnologija lahkih odjemalcev, se imenuje ICA (Independent Computing Architecture – opisan bo v naslednjem poglavju) in je visoko optimiziran za prenašanje grafičnih ekranskih slik, tako da za normalno delo zadostuje že navadna 28,8 kilobitna modemska povezava s strežnikom. Kljub ozkopasovni, le 28,8 kilobitni povezavi, so odzivni časi primerljivi tistim, ki smo jim deležni, če smo priključeni v LAN omrežje.

Slika 5: Podatkovni tok ICA protokola



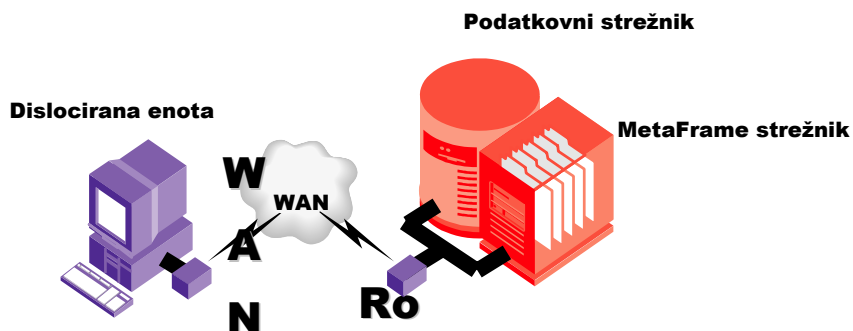
Vir: Citrix, 2002

Sam koncept je torej zelo podoben tistemu, ki ga poznamo od velikih centralnih računalnikov, kjer smo imeli centralni, t.i. mainframe računalnik, s katerim so bili povezani "neumni" terminali. MetaFrame tehnologija lahkega odjemalca ta sistem razširja in ga naredi bolj odprtega, saj lahko kot odjemalci nastopijo "neumni" terminali, osebni računalniki in še cela vrsta drugih informacijskih naprav.

V nadaljevanju je navedenih nekaj tipičnih primerov uporabe MetaFrame tehnologije (Kanter, 1998, 161-173) :

- Podjetja z dislociranimi poslovnimi enotami. Tu se pojavlja problem podvajanja podatkov, prenosa podatkov po WAN omrežju in s tem povezane varnosti in seveda problem odzivnih časov. V tem primeru uporaba tehnologije lahkega odjemalca omogoči interaktivno delo na centralni bazi podatkov z odzivnimi časi, ki so primerljivi oziroma enaki odzivnim časom na lokalnem omrežju.

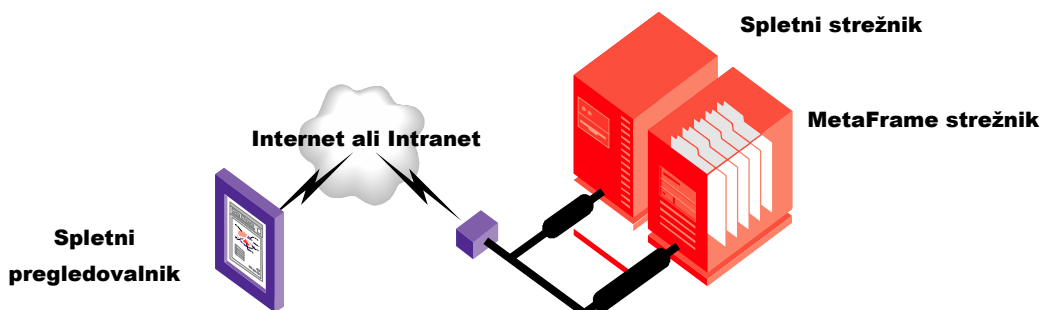
Slika 6: Primer dislocirane enote



Vir: Kanter, 1998

- Internet. Svojo aplikacijo lahko naredimo dostopno iz interneta ali intraneta, ne da bi bilo potrebno na novo napisati eno samo vrstico programske kode. Tudi starejše DOS aplikacije lahko izvajamo preko interneta povsem brez predelave.

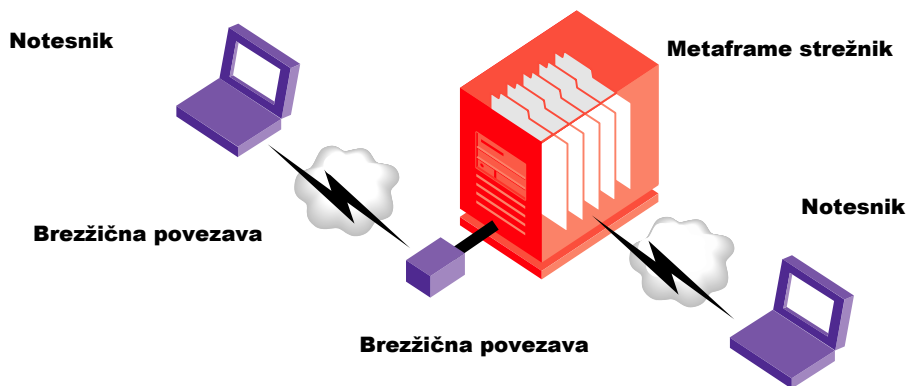
Slika 7: Primer povezave preko interneta



Vir: Kanter, 1998

- Potniki na terenu: Podjetja, ki imajo terenske potnike, ki na mestu prodaje izstavijo račun, dobavnico ali izpolnijo naročilo, se lahko s svojim prenosnikom in modemom neposredno (ali preko interneta) povežejo s osrednjim računalnikom in imajo dostop do operativnih podatkov o artiklih, njihovih zalogah in cenah.

Slika 8: Primer brezžične povezave



Vir: Kanter, 1998

4.1.3 Independent Computing Architecture (ICA) protokol

V tem poglavju bo opisan Independent Computing Architecture (ICA) protokol, ki skupaj z MultiWin jedrom predstavlja temelj Citrix MetaFrame tehnologije lahkega odjemalca. ICA je storitveni protokol za prikaz aplikacij v okolju Windows, ki omogoča strežbo aplikacij lahkim odjemalcem.

4.1.3.1 ICA funkcionalnost

ICA protokol je namenjen hitremu prikazovanju okenskega okolja preko LAN, WAN ali komutiranih omrežij in tako zmanjševanju omrežnega prometa. Omogoča prenos uporabniškega vmesnika Windows aplikacij preko ozkopasovnih povezav. ICA je robusten in nadgradljiv protokol, ki vsebuje definicije za naslednje zmožnosti (Citrix, 1998):

- prikaz besedila preko celotnega ekrana (Dos emulacija),
- grafično prikazovanje okenskih aplikacij,
- vhod tipkovnice in miške,
- nadzor seje,
- okvirjanje za asinhrono povezave,
- odkrivanje napak in odpravljanje napak,
- enkripcija,
- kompresijske zanke,
- preusmeritev datotečnega sistema,
- preusmeritev tiskanja,
- več splošnih navideznih kanalov,
- izreži in prilepi preko strežnikov (Copy – Paste),
- preusmeritev komunikacijskih vrat.

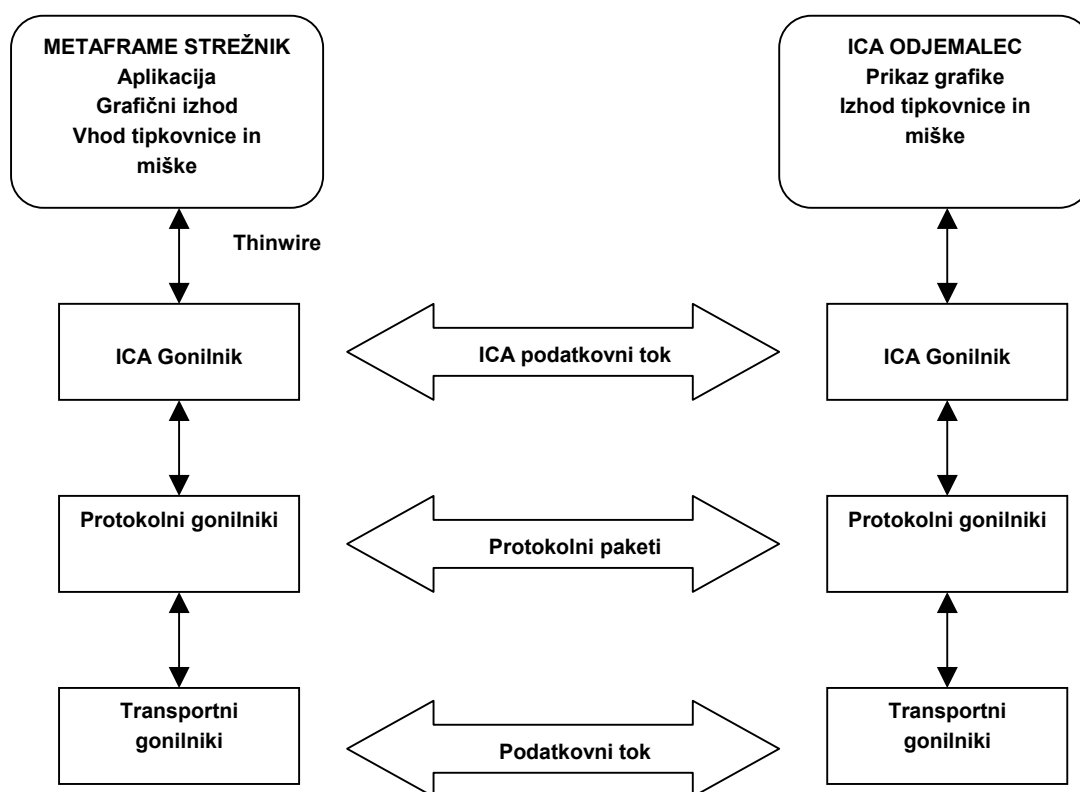
4.1.3.2 Delovanje ICA protokola

ICA je fizični protokol, ki se uporablja za komunikacijo med odjemalcem (osebni računalnik, Unix delovna postaja, Macintosh računalnik) in aplikacijskim strežnikom MetaFrame. ThinWire je ime podatkovnega protokola, ki eksportira grafično sliko določene aplikacije. ThinWire je logični tok podatkov, ki tečejo oviti v ICA paketu. ThinWire ni fizičen protokol. Fizičen protokol ICA mora zagotoviti, da ThinWire podatkovni tok prispe brez kakršnihkoli napak do odjemalca oz. strežnika.

Na aplikacijskem strežniku MetaFrame je ThinWire sestavni del GDI-ja (angl. Graphical Device Interface) in podsistema za video gonilnik. V sodelovanju z WinFrame-Win32 podsistemom skupaj generirata optimizirane ukaze za prikazovanje okenskih objektov. Vse skupaj se imenuje ThinWire protokol.

Izhod ThinWire protokol gonilnika je logični podatkovni tok, ki se pošlje preko navideznega kanala API-ja (angl. Application Programming Interface). Ta prevzame podatkovni tok in ga vstavi v ICA paket. Izoblikovan ICA paket gre skozi nekaj gonilnikov za protokole, kjer lahko dodamo morebitno kriptiranje, kompresiranje in okvirjanje. Potem se paket pošlje preko transportne plasti do odjemalca. Pri odjemalcu gre paket skozi iste plasti v obratnem vrstnem redu. Rezultat tega je grafični prikaz aplikacije (grafičnih objektov) pri uporabniškem vmesniku odjemalca.

Slika 9: Delovanje ICA protokola



Vir: Citrix, 1998

4.1.3.3 Sestava ICA paketa

ICA paket je sestavljen iz obvezne, en bajt velike ukazne kode, kateri sledijo morebitni podatki. Ta paket ima lahko opsijske glave za upravljanje prenosa paketa, ki se vključijo glede na pogovor med strežnikom in odjemalcem v času vzpostavitve povezave. Celotna struktura paketa ICA je odvisna od medija prenosa podatkov (LAN, asinhrono) in morebitnih opcij, ki jih zahteva uporabnik (kompresija, enkripcija).

ICA paketi imajo naslednjo strukturo (Xiaochun, 2002):

GLAVA OKVIRA (angl. frame head)
GLAVA ZA ODKRIVANJE NAPAK (angl. reliable preamble)
GLAVA ZA ENKRIPCijo (angl. encrypt preamble)
GLAVA ZA KOMPRESIJO (angl. compress preamble)
UKAZ (angl. command code)
UKAZNI PODATKI (angl. command data)
KONEC OKVIRJA (angl. frame trail)

- Glava okvirja (angl. Frame head). Opcijska glava za okvirjanje. Uporablja se za okvirjanje podatkovnega toka.
- Glava za odkrivanje napak (angl. Reliable preamble). Opcijska glava za zanesljivost prenosa podatkov. Uporablja se za odkrivanje in odpravljanje napak.
- Glava za enkripcijo (angl. Encrypt preamble). Opcionalna glava za enkripcijo. Uporablja se upravljanje kriptiranih podatkov.
- Glava za kompresijo (angl. Compression preamble). Opcionalna glava za kompresijo podatkov. Uporablja se za upravljanje kompresiranih podatkov.
- Ukaz (angl. Command). En bajt dolg ICA ukaz. Predstavlja začetek paketa ICA protokola.
- Ukazni podatki (angl. Command data). Opcionalni podatki v zvezi s predhodnim ukazom. Dolžina podatkov je odvisna od samega ukaza in je lahko tudi nič.
- Konec okvirja (angl. Frame trail). Uporablja se za okvirjanje pri asinhronih podatkovnih prenosih.

Samo komanda je vedno prisotna v paketu. Prisotnost določenih glav je odvisna od prenosa in se določi ob inicializaciji. Prisotnost ukaznih podatkov je odvisna od samega ukaza.

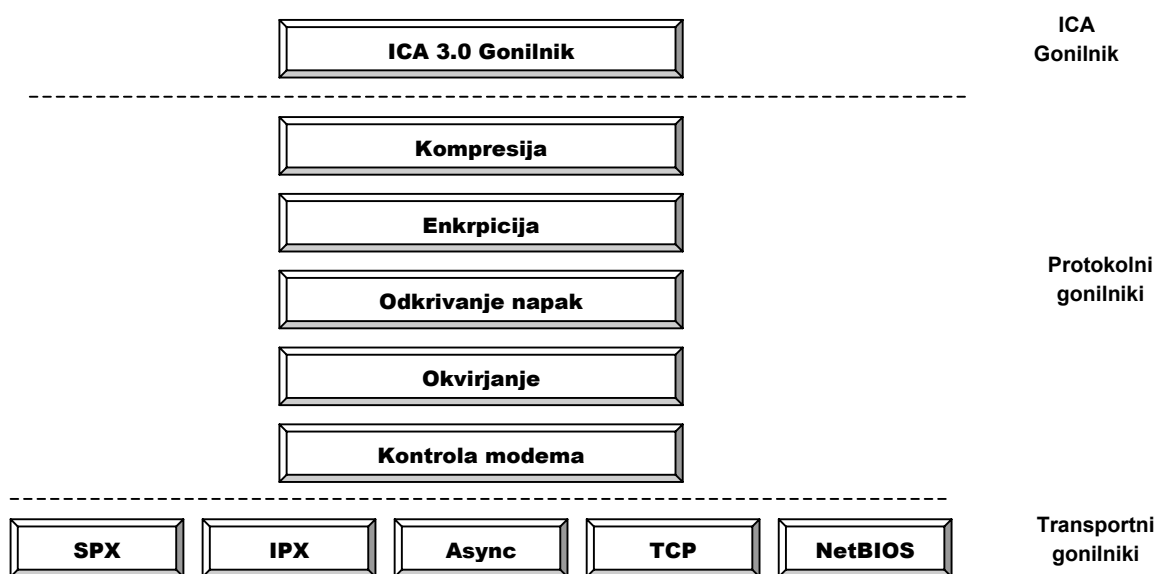
4.1.3.4 ICA protokol in transportne plasti

Pod podatkovnimi paketi ICA so lahko različni gonilniki protokolnih plasti. ICA protokol ne potrebuje teh plasti. Njihova prisotnost in uporaba se določi med rokovaljem, ki se zgodi na začetku določene seje. Ker so te plasti spodaj, se lahko odvzamejo ali pa zamenjajo. Po potrebi se lahko določene plasti tudi dodajo. Velika množica dodanih modulov za protokole omogoča delovanje z najbolj razširjenimi transportnimi tehnologijami, kot so npr. TCP/IP, NetBIOS, IPX/SPX in PPP/SLIP, in s tem pravo neodvisnost.

Sklad ICA protokola se lahko dinamično sestavlja, tako da ugotovi potrebam določenega transportnega protokola. Primer: Ker IPX ni zanesljiv, se lahko doda zanesljiv gonilnik protokola nad IPX transportnim gonilnikom. Ker pa IPX temelji na okvirjanju, lahko gonilnik za okvirjanje odvezamo iz sklada ICA protokola. V primeru TCP protokola, ki ne temelji na okvirjanju, pa moramo gonilnik za okvirjanje dodati. Ker pa je TCP zanesljiv, lahko zanesljiv gonilnik odvezamo iz sklada ICA protokola.

Slika 10 prikazuje podatkovni sklad ICA protokola.

Slika 10: Sklad ICA protokola



Vir: Xiaochun, 2002, str. 9

- **Kompresija.** Protokol za kompresijo lahko ovije kriptiran ICA paket. Kompresijski protokol se lahko zamenja in ni točno določen s strani ICA protokola.
- **Enkripcija.** Protokol za enkripcijo lahko ovije ICA paket. Enkripcijski protokol se lahko zamenja in ni natančno določen s strani ICA protokola.
- **Odkrivanje napak.** Prenosni protokol za odkrivanje napak se uporablja za odkrivanje napak pri prenosu podatkov in njihovo odpravljanje. Ta protokol se uporablja z vsakim protokolom, ki ne zagotavlja zanesljivo dobavo. Takšen protokol je npr. Novell-ov IPX in standardne asinhronne povezave.
- **Okvirjanje.** Prenosni protokol za okvirjanje se uporablja za upravljanje komunikacij, ki ne bazirajo na okvirjanju, kot npr. TCP. Skupaj s protokolom za odkrivanje napak se uporablja za prenos podatkov brez napak.
- **Kontrola modema.** Ta kontrolni protokol se uporablja za upravljanje modema, njegovo odkrivanje in inicializacijo pred vzpostavitvijo zveze.

4.1.3.5 ICA razširljivost

ICA je fleksibilen in razširljiv protokol. Protokol je tako nastavljen, da se prilagaja različnim zmogljivostim odjemalca. Med rokovaljem ICA odjemalec posreduje strežniku informacije o resoluciji ekrana, globine barv, velikost pomnilnika in ostale podatke, na podlagi katerih se protokol potem ustrezno prilagodi. Ta komunikacija omogoča veliko število različnih odjemalcev, od enostavnega terminala do visoko zmogljive delovne postaje.

Z arhitekturo navideznih kanalov je omogočena razširljivost z novimi podatkovnimi tipi, kot sta npr. zvok in slika. Navidezne kanale lahko tudi uporabljamo za podporo dodatnih naprav na strani odjemalca, kot npr. čitalec črtne kode ali pa skenerji.

Modularna arhitektura ICA protokola omogoča dodajanje novih plasti. Lahko bi obstoječi modul za enkripcijo zamenjali z RSA ali DES enkripcijo. Po potrebi bi lahko dodali konverteje, kot npr. ICA/X.11. Podpora novim transportnim protokolom, kot npr. ATM, je enostavno izvedljiva.

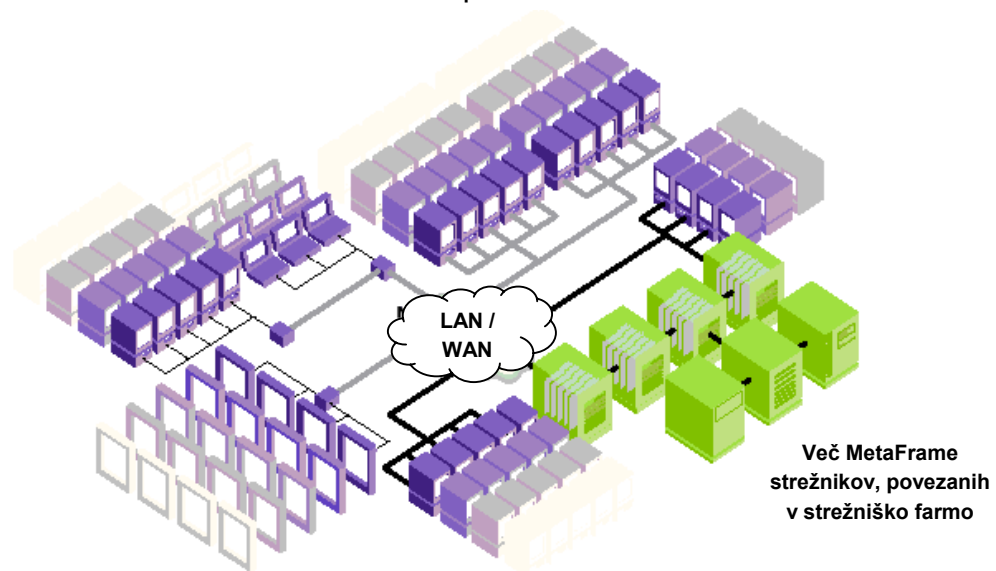
4.1.4 Orodja za administracijo in dodatki

Citrix MetaFrame strežnik nam nudi velik nabor orodij za administracijo. Poleg klasičnih Windows NT Server funkcij za upravljanje strežnika in pravic, so nam na voljo orodja s katerimi lahko upravljamo večuporabniško jedro. V nadaljevanju bodo predstavljene in opisane najbolj pomembna orodja in njihove funkcionalnosti.

4.1.4.1 Izenačevanje obremenitve

Funkcionalnost Izenačevanje obremenitve (angl. Load Balancing) nam omogoča postavitev strežniške farme (angl. Server Farm). Tako je več MetaFrame ali WinFrame strežnikov povezanih v farmo z namenom, da lahko podpremo večjo število uporabnikov.

Slika 11: MetaFrame strežniki povezani v strežniško farmo



Vir: Citrix, 2002

Aplikacije, ki jih želimo izvajati, morajo biti nameščene na vseh strežnikih, ki so del farme. Tako lahko isto aplikacijo 'objavimo' (publishing - objavljane aplikacij bo podrobneje opisano kasneje) na več oz. na vseh MetaFrame strežnikih, ki so del farme. Ko uporabniki zaženejo objavljeno aplikacijo oz. sejo, ki je bila konfigurirana na večih strežnikih, Load Balancing izbere strežnik, ki bo izvajal aplikacijo. Izbira temelji na množici nastavljivih parametrov, ki tvorijo kalkulacijo obremenitve določenega strežnika.

Administrator sistema uporablja Load Balancing Administrator Utility za določitev osnovnih parametrov obremenitve, kot npr. število uporabnikov, ki predstavlja polno obremenitev. Za bolj napredno definicijo obremenitve so na voljo naslednji parametri:

- **Uporaba odstranjevalne datoteke** (angl. Pagefile Usage). Razmerje med največjo dovoljeno velikostjo odstranjevalne datoteke in njeno trenutno velikostjo.

- **Aktivnost premeščanja** (angl. Swap Activity). Število dostopov do odstranjevalne datoteke na sekundo.
- **Obremenitev procesorja** (angl. Processor Usage). Odstotek časa, v katerem je procesor strežnika zaseden.
- **Zasedenost pomnilnika** (angl. Memory Load). Razmerje med prostim pomnilnikom in skupnim pomnilnikom strežnika.
- **Seje** (angl. Sessions). Razmerje med še razpoložljivimi in dovoljenimi sejami.

Administrator lahko vsakemu od zgoraj naštetih parametrov dodeli svojo težo, tako da je mogoče kalkulacijo obremenitve strežnika prilagoditi aplikacijam, ki se izvajajo na strežniku. Če imamo opravka z aplikacijo, ki ne zahteva velike računalniške moči (npr. POS – Point of Sale), bomo največjo težo dodelili številu uporabnikov, ki so lahko priključeni na strežnik. V primeru da, aplikacija izredno bremeni procesor, bomo parametru "Obremenitev procesorja" dodelili večjo težo.

4.1.4.2 Objavljanje aplikacij

Objavljanje aplikacij (angl. Application Publishing) pomeni namestitev in konfiguracijo aplikacij na večuporabniškem strežniku, tako da lahko ICA odjemalci neposredno dostopijo do njih. Administrator v te namene uporablja orodje "Published Application Manager". Cilj objavljanja aplikacij je doseči čim enostavnejši, način kako lahko uporabniki dostopajo do aplikacij na MetaFrame strežniku. Objavljena aplikacija poenostavi uporabnikom delo na naslednji način:

- **Naslavljanje aplikacij** (Application addressing). Namesto da se uporabniki povežejo z strežnikom preko IP naslova oz. imena, lahko uporabljajo ime objavljene aplikacije. Povezovanje z aplikacijo preko njenega imena olajša delo uporabnikom, kajti ni več treba pomniti imena strežnika, na katerem se nahaja.
- **Navigacija aplikacij** (Application navigation). Uporabnikom ni potrebno znanje o uporabi namizja na operacijskem sistemu Windows NT/2000, kajti aplikacija se samodejno prikaže znotraj ICA seje.
- **Preverjanje uporabnikove identitete** (User authentication). Uporabnikom se ni treba prijavljati na strežnik in nato še v aplikacijo.

4.1.4.3 Senčenje

Senčenje (angl. Shadowing) je orodje, ki je neposredno namenjeno zmanjševanju stroškov administracije in podpore uporabnikom. Gre za prevzemanje kontrole,

nadziranje ali pridružitve trenutne uporabnikove seje. Po aktiviranju senčenja določenega uporabnika se administratorju pokaže uporabnikov ekran. Pri tem ni nujno, da se uporabnik sploh zaveda tega, da se nad njegovo sejo izvaja senčenje. Po želji lahko administrator prevzame miško oz. tipkovnico uporabnikove seje. Na tak način lahko izvaja oddaljeno šolanje uporabnikov na določenih delih aplikacije, kajti v večini primerov se uporabnikom zatakne prav na določenih mestih aplikacije. Ko pride do problema, uporabnik pokliče center za pomoč (angl. Help desk), ki vklopi senčenje uporabnikove seje. Če je možno, bo center za pomoč poskusil problem rešiti, uporabnik pa lahko sledi postopkom in se hkrati nauči kako postopek izpeljati sam.

Ni treba poudarjati, da je z uporabo senčenja prihranek časa velik in posledično kvaliteta podpore uporabnikom zelo učinkovita.

4.1.4.4 Vdelava aplikacij

ALE (angl. Application Launching and Emedding) izvajanje aplikacij omogoča popolno funkcionalnost obstoječih Windows aplikacij, ki se izvedejo npr. vdelane (embedded in) v HTML spletne strani, brez sprememb v programski kodi. Do aplikacij je mogoče dostopati preko ActiveX kontrole za Microsoft Internet Explorer, dodatnega programa (angl. plug-in) za Netscape Navigator in Javanskega apleta za vsak Java usposobljen sistem.

Administratorji lahko hitro integrirajo Windows aplikacije na uporabnikovo namizje. Razvijalci lahko razširijo poslovne aplikacije preko spleta in tako prihranijo pri času in denarju. Vstavljena aplikacija je videti in se obnaša tako, kot če bi se izvajala lokalno, čeprav se izvaja na MetaFrame strežniku, ki lahko tiči za spletnim strežnikom in požarnim zidom.

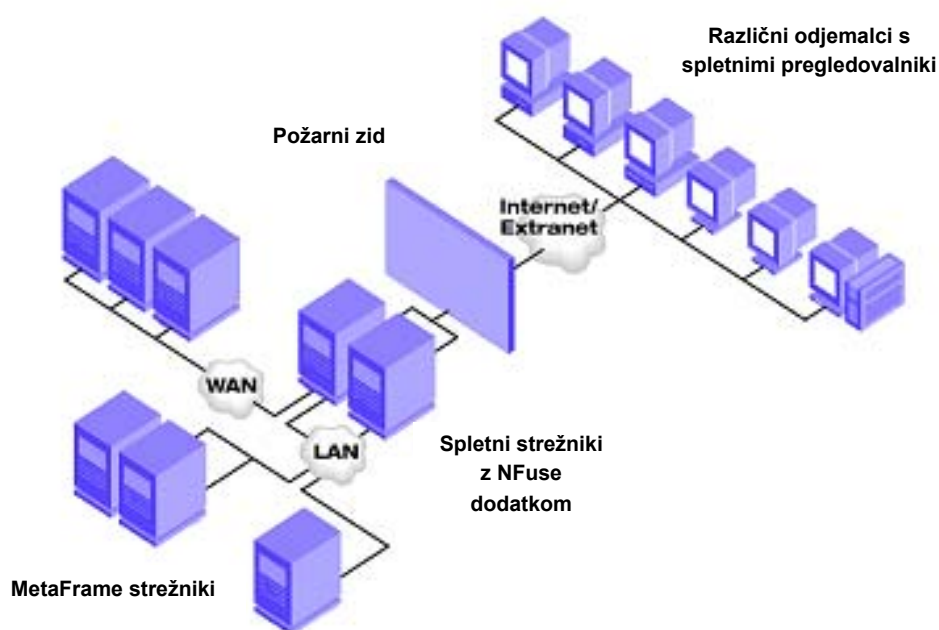
Za usposabljanje ALE aplikacij nam je v pomoč administrativno orodje ALE čarovnik, ki nam samodejno generira HTML izvorno kodo, ki je potrebna za prikaz aplikacij z uporabo ALE.

4.1.4.5 NFuse

Citrix NFuse, temelječ na ALE tehnologiji, omogoča integriranje aplikacij v korporacijske spletne portale, do katerih se uporabniki povezujejo s standardnimi spletnimi pregledovalniki. Na ta način lahko podjetja strežejo aplikacije preko vsake spletne povezave, žične ali brezžične, brez dragega časovno zamudnega in napornega prepisovanja izvirne kode aplikacije. NFuse omogoča uporabnikom povezovanje z aplikacijo na naraven način, s čimer je zagotovljena prava mobilnost na lokacijah kupcev, hotelih, doma – povsod, kjer to zahtevajo okoliščine poslovanja.

Ker NFuse zagotavlja orodja, potrebna za popolno izkoriščanje interneta kot kanala za strežbo aplikacij in vsebin, lahko uporabniki zadovoljijo trenutne in bodoče zahteve računalniškega poslovanja in prek interneta izkoriščajo obstoječe aplikacije.

Slika 12: Dostop do aplikacij z uporabo NFusa preko spletnih pregledovalnikov



Vir: Citrix, 2002

Podobno, kot aplikacije v portalih združujejo več virov informacij na enem mestu, združuje NFuse posamezne aplikacije v sklope, ki so potem na voljo uporabnikom v njihovih spletnih pregledovalnikih.

Z uporabo Management Console programa za upravljanje administratorji portalov definirajo po meri pripravljene sklope aplikacij za opravljanje specifičnih nalog v podjetju in hkrati zagotavljajo, da nepooblašчени uporabniki določenih aplikacij ne morejo uporabljati. Uporabniki portalov lahko svoje aplikacijske portale prilagajajo svojim potrebam s spreminjanjem različnih nastavitev.

Aplikacije, ki jih streže NFuse, se lahko izvajajo v ločenem pregledovalnikovem oknu ali v okviru samega portala. S pomočjo NFuse tehnologije lahko administratorji povečajo privlačnost spletnih portalov, tako da vanje neposredno vključijo vse aplikacije z drugačno spletno vsebino. Slika 13 prikazuje program Microsoft Excel, ki se izvaja v aplikacijskem portalu.

Slika 13: Microsoft Excel se izvaja v spletnem portalu



Vir: Citrix, 2002

Citrixovi aplikacijski portali se dopolnjujejo z aplikacijskimi portali drugih proizvajalcev, ker je sklop aplikacij vedno le objekt v obstoječem portalu. Pravzaprav so se številni vodilni ponudniki programskih rešitev za poslovne portale povezali s podjetjem Citrix, da bi skupnim kupcem ponudili aplikacijske portale z vsemi potrebnimi lastnostmi. Ti partnerji so razvili posebne objekte za NFuse, ki omogočajo interaktiven dostop do aplikacij, ki se izvajajo na strežnikih MetaFrame.

4.1.4.6 Storitve za upravljanje virov

Orodje RMS (angl. Resource Management Services) omogoča administratorjem analiziranje in nato optimiranje Citrix MetaFrame strežnikov. Najbolj ključne funkcionalnosti so opisane v nadaljevanju:

- **Ustvarjanje dnevnika aplikacij** (angl. Application and system audit trail creation). RMS deluje z vsemi podatkovnimi bazami, ki so dostopne preko ODBC vmesnika (angl. Open Database Connectivity Interface). V določeno bazo podatkov beleži prijave in odjave uporabnikov, dolžine uporabnikovih

sej, število in imena aplikacij, ki so se uporabljale v teku seje. Na podlagi zabeleženih podatkov lahko administrator preverja odstotek časa in pomnilnika, ki so ga potrošile določene aplikacije. Te ugotovitve nato uporabi za pravilno planiranje dodatnih resursov.

- **Natančno opazovanje sistema** (angl. Comprehensive system monitoring). RMS alarmi javijo administratorju potencialne probleme, preden le-ti dejansko nastopijo. Več kot 30 kazalcev pomaga pri določevanju statusa sistema. Administrator lahko nastavi samodejno pošiljanje alarmne elektronske pošte oz. sporočil SMS na svoj mobilni telefon, kadar izbrani kazalci dosežejo določeno mejo. Ekspertni sistem, ki je del RMS-a, dodatno pomaga pri uspešnem reševanju morebitnih problemov.
- **Podpora za obračunavanje uporabe sistema** (angl. Support for charge-back systems). V določenih organizacijah se uporaba informacijskih sistemov po oddelkih obračunava v namene določevanja nastalih stroškov. RMS omogoča nastavljanje cene na minuto priključene seje, ceno na uporabo pomnilniškega prostora ali ceno na uporabo procesorskega časa. Uporabnike lahko združimo v stroškovne skupine, na podlagi katerih lahko izdelamo razne izpise. Izpisi nam lahko služijo kot obračunski listi.

RMS se izvaja kot Windows NT storitev (angl. NT services) v ozadju MetaFrame strežnika, ki smo ga izbrali za analizo. Grafični prikazi analiz nam olajšajo delo in učenje s tem orodjem. Z uporabo RMS-a bodo administratorji lahko ugotovili in hkrati odpravili morebitne performančne slabosti MetaFrame strežnika.

4.1.4.7 Storitve za upravljanje namestitve

IMS (angl. Installation Management Services) je namenjen avtomatizaciji namestitvenega procesa novih aplikacij. Prav tako omogoča replikacijo nove aplikacije na več MetaFrame strežnikih v določeni organizaciji. Z uporabo IMS-a lahko novo aplikacijo oz. novo različico aplikacije namestimo na več strežnikih v nekaj minutah. IMS je sestavljen iz dveh delov: Packager in Installer. Installer je nameščen na vseh MetaFrame strežnikih, ki so v organizaciji. Tako je namestitev nove aplikacije sestavljena le iz 'pakiranja' in 'objavljanja' (angl. Package and Publish). Pri pakiranju aplikacije nam pomaga čarovnik, ki generira t.i. paket, katerega nato enostavno pošljemo oz. objavimo vsem MetaFrame strežnikom, ki imajo nameščen Installer. Aplikacija bo nato samodejno nameščena na teh izbranih strežnikih. Prav tako lahko vse aplikacije, nameščene preko IMSa odstranimo iz določenih strežnikov na zelo hiter in enostaven način.

Kot je razvidno je IMS še posebej priročno orodje v organizacijah, kjer imamo opravka z več MetaFrame strežniki in frekventivnimi obnovitvami aplikacij. V takih

okoljih lahko IMS prinese velike prihranke administrativnega časa in s tem posledično denarja.

4.1.4.8 VideoFrame

Vedno več organizacij se odloča za digitalni video in avdio prenos z namenom izboljšati kvaliteto komunikacije s svojimi zaposleni oz. strankami. Tehnologija za digitalni prenos zvoka in slike se imenuje pretok (angl. Streaming). Pri pretakanju se avdio in video podatki prenašajo preko omrežja do odjemalca. Odjemalec mora imeti nameščene določene kodeke (angl. CODEC), ki omogočajo prikaz slike in zvoka. Kodek je kratica za kodiranje in dekodiranje analognih signalov v digitalno obliko in nasprotno. Z uporabo kodeka zagotavljamo optimalni izkoristek hitrosti podatkovne povezave in kakovost predvajanja digitaliziranih podatkov v osnovno obliko. Najbolj znani videokodeki so MPEG, Indeo, Quicktime in Video for Windows (Pahor, 2002).

Citrix ponuja tehnologijo VideoFrame kot dodatek MetaFrameu. VideoFrame izkorišča izredno učinkovit ICA protokol in Citrix Video kodek za prenos visoko kvalitetnega zvoka in slike. Pri prenosu se samodejno prilagaja razpoložljivi pasovni širini, tako da je vsak odjemalec deležen optimalne kvalitete. Kot dodatek MetaFrameu, VideoFrame takoj izkoristi infrastrukturo, ki smo jo nastavili z implementacijo MetaFramea.

Zelo učinkovita uporaba VideoFramea in MetaFramea je npr. oddaljeno šolanje uporabnikov. Uporabniki lahko poleg aplikacije, ki je predmet učenja, še vidijo in poslušajo učitelja.

4.1.4.9 MetaFrame za UNIX

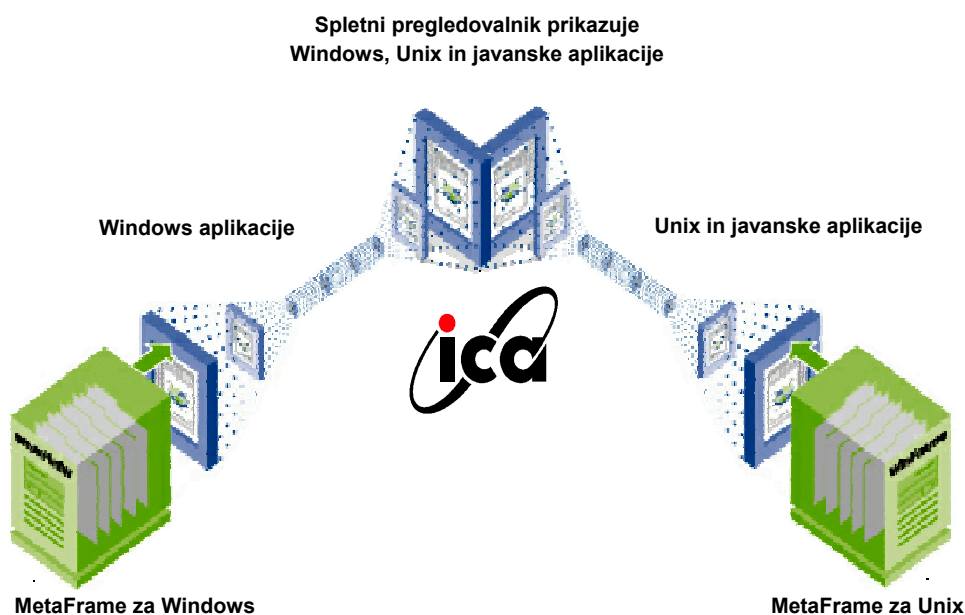
Po izrednem uspehu MetaFrame strežnikov za Windows aplikacije je podjetje Citrix tudi objavilo MetaFrame za UNIX, X Windows in javanske aplikacije. Produkt trenutno podpira Sun Solaris in HP-Unix operacijski sistem in ponuja podobno funkcionalnost kot MetaFrame za Windows.

Jedro MetaFrame za Unix produkta je modificiran X11R6.3 strežnik. Ta protokol skrbi za ICA seje, ki so priključene na MetaFrame za Unix strežnik. Podprte so vse standardne X11 aplikacije, ki sedaj uporabljajo modificiran MetaFrame X strežnik namesto klasičnega X11 strežnika. V primerjavi z X11 protokolom Citrixov ICA protokol porabi sto krat manj pasovne širine (Citrix, 2002), kar omogoča uporabo X11 aplikacij preko ozkopasovnih modemskih povezah. Odjemalci pa lahko prihajajo iz širokega nabora naprav, ki podpirajo ICA protokol.

Poleg X11 aplikacij lahko strežemo tudi javanske aplikacije. Pojavlja se seveda vprašanje, čemu bi distribuirali javanske aplikacije prek ICA protokola, saj so že same po sebi koncipirane tako, da se lahko izvajajo na skoraj vseh operacijskih sistemih in napravah. V teoriji to drži. V praksi pa se še vedno srečujemo z različnimi težavami, kot npr. čas za prenos javanske aplikacije na odjemalca in problemi z različnimi verzijami javanske virtualne mašine. Tako se aplikacija na nekaterih verzijah javanske virtualne mašine obnaša drugače kot na drugih.

Z uporabo ICA protokola seveda odpade čas za prenašanje in nameščanje javanske aplikacije. Tudi javanska virtualna mašina je vedno enaka, saj se bo javanska aplikacija izvajala na MetaFrame strežniku. To seveda tudi zmanjšuje stroške razvoja, kajti testiranje javanske aplikacije bo potrebno le na strežniški javanski virtualni mašini. Omeniti je seveda potrebno, da je javanska virtualna mašina velik programski paket, ki ga ne more izvajati vsak odjemalec. Ta problem odpade z uporabo ICA lahkega odjemalca, ki ga lahko izvaja že naprava s procesorsko močjo Intel 286 procesorja.

Slika 14: MetaFrame za Unix



Vir: Citrix, 2002

MetaFrame za Unix je lahko pomemben dodatek k obstoječim MetaFrame za Windows strežnikom. Strežnike je moč vključiti v obstoječe strežniške farme in Load Balancing arhitekture. S pomočjo NFuse-a lahko X11 in javanske aplikacije sedaj uporabljamo na poslovnih portalih s pomočjo standardnih spletnih pregledovalnikov.

4.2 Microsoft Terminal Services Edition (TSE)

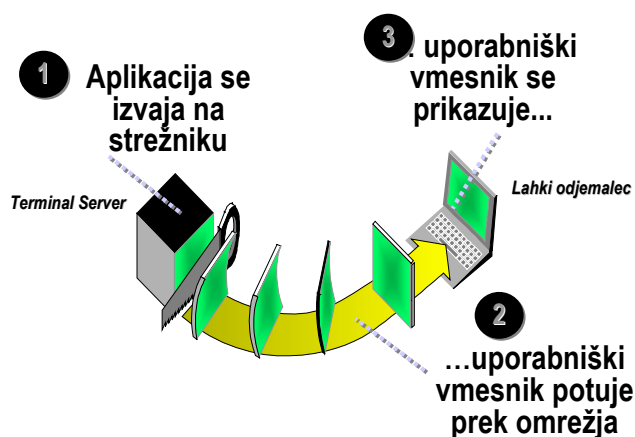
4.2.1 Predstavitev Microsoft TSE

Po izrednem uspehu Citrixovega WinFrame produkta se je podjetje Microsoft leta 1997 odločilo, da licencira večuporabniško jedro MultiWin za Windows NT od podjetja Citrix. Microsoft je nato integriral MultiWin tehnologijo v operacijski sistem Windows NT 4.0 in produkt poimenoval Windows NT Server Terminal Services Edition (TSE). Pri operacijskem sistemu Windows 2000 je dodatek Terminal Services že implementiran v samem jedru. Windows NT Terminal Services Edition sestavljajo naslednje tri komponente:

- **Terminal Server.** To je večuporabniško jedro, ki omogoča upravljanje več hkratnih sej na Windows NT strežniku. Klasične Windows aplikacije ne potrebujejo nobenih sprememb, da se lahko izvajajo na Terminal Serverju. Razvidno je, da je temelj Terminal Serverja dejansko MultiWin jedro, ki ga je izdelalo podjetje Citrix in ga uporablja v MetaFrame dodatku.
- **Remote Desktop Protokol (RDP).** Ključna komponenta Terminal Serverja je RDP protokol, ki omogoča t.i. super lahkega odjemalca (angl. super-thin client) priključitev na Terminal Server preko omrežja. Protokol temelji na ITU (International Telecommunications Union) T.120 protokolu, ki je mednarodni večkanalni konferenčni protokol in se uporablja tudi v Microsoftovem programskem paketu NetMeeting. Optimiran je za širokopasovna okolja in naj bi v prihodnosti podpiral kriptiranje sej.
- **"Super-lahki" Odjemalec.** Programska oprema na odjemalcu prikazuje splošno znani 32 bitni Microsoft Windows uporabniški vmesnik na naslednjih strojnih opremah:
 - Windows Terminali,
 - osebni računalniki z operacijskim sistemom Windows 95, Windows NT, Windows 2000,
 - osebni računalniki z operacijskim sistemom Windows 3.11.

Kot je razvidno iz zgoraj navedenega, se Microsoft Terminal Server razlikuje od Citrix MetaFrame strežnika predvsem v prenosnem protokolu in odjemalcu. Ker prenosni protokol in odjemalec nista tako izpopolnjena kot Citrixov ICA protokol in odjemalec, se je možno odločiti za nadgradnjo Terminal Serverja z Citrix MetaFrame dodatkom. Slika 15 prikazuje koncept Terminal Serverja, ki je seveda enak Citrixovem modelu.

Slika 15: Delovanje Terminal Serverja



Vir: Microsoft, 2000a, str. 2

4.2.2 Večuporabniško jedro Windows NT

Windows NT operacijski sistem ni bil nikoli predviden za večuporabniške namene. Šele z implementacijo Citrixovega MultiWin jedra za Windows NT je uporabnikom omogočeno prijavljanje na navidezne Windows NT seje.

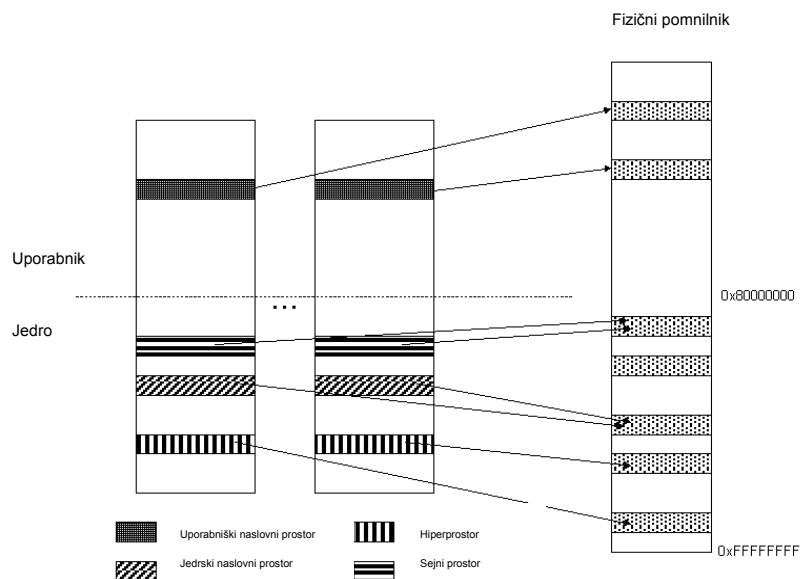
Da bi dosegli večuporabniške sposobnosti, je bilo treba dodati nove komponente in spreminjati obstoječe komponente jedra operacijskega sistema. Module, kot npr. Virtual Memory Manager (VMM) in Object Manager (OM), je bilo potrebno prilagoditi večuporabniškemu okolju.

Object Manager sedaj vsakemu objektu, ki se tvori znotraj določene seje, prilepi enoličen identifikator. Ta identifikator je povezan s sejo, v kateri je bil ustvarjen. Če npr. zaženemo aplikacijo znotraj seje št.1, bo Object Manager prilepil imenu objekta številko seje: "\\ime aplikacije:1".

Virtual Memory Manager preslikuje naslove navideznega pomnilniškega prostora določenih procesov na fizične naslove v računalniškem pomnilniškem prostoru. Pomnilniški prostor Windows NT procesov je razdeljen na dva 2 gigabajtna odseka. To sta uporabniški (angl. user - procesno specifični naslovi) in jedrski odsek (angl. kernel - sistemsko specifični naslovi). Za uporabniški prostor skrbi VMM, tako da vsak proces lahko naslavlja le svoj pomnilniški prostor in ne more naslavljaliti pomnilniškega prostora drugih procesov. V nasprotju z uporabniškim prostorom je jedrski prostor skupen vsem procesom znotraj Windows NT sistema. Ker si vsi procesi delijo isti jedrski prostor, je pri več hkratnih sejah na enem strežniku prihajalo do pomanjkanja jedrskega prostora. Rešitev tega problema je prostor znotraj jedrskega prostora in se je imenuje sejni prostor (angl. SessionSpace). Sejni prostor preslikujemo na podlagi tekočih sej. Vsaka seja dobi

svoj identifikator seje in sejni prostor znotraj jedrskega prostora, tako da vsak proces znotraj določene seje naslavlja samo svoj navidezni jedrski prostor.

Slika 16: Preslikava jedrskega in uporabniškega pomnilniškega prostora



Vir: Microsoft, 2000a, str. 4

Poleg sprememb Virtual Memory Managerja in Object Manager je bila še dodana nova Windows NT storitev, ki se imenuje "Terminal Server" (termsrv.exe). Predstavlja glavni kontrolni proces, ki skrbi za upravljanje, inicializacijo in prekinitev uporabniških sej. Storitve "Terminal Server" je popolnoma neodvisna od protokola, s katerim deluje, tako da je dovoljena uporaba RDP ali ICA protokola za povezavo z odjemalci.

4.2.3 Remote desktop Protocol (RDP)

Remote Desktop Protocol je nadgradnja ITU (International Telecommunications Union's) T.120 protokola. Omogoča večpasovno komunikacijo med strežnikom in odjemalcem, tako da lahko na navideznih kanalih prenašamo uporabniški vmesnik, periferijo, tipkovnico in miško posebej. Med drugim dovoljuje tudi večtočkovno (angl. Multipoint) posredovanje podatkov, kar omogoča posredovanje podatkov večim odjemalcem hkrati.

Podjetje Microsoft se je odločilo za uporabo RDP protokola v Terminal Serverju zaradi dobrega razvojnega izhodišča tega protokola. RDP namreč ponuja do 64.000 različnih navideznih kanalov za prenos podatkov. Prav tako je dovoljeno večtočkovno posredovanje podatkov.

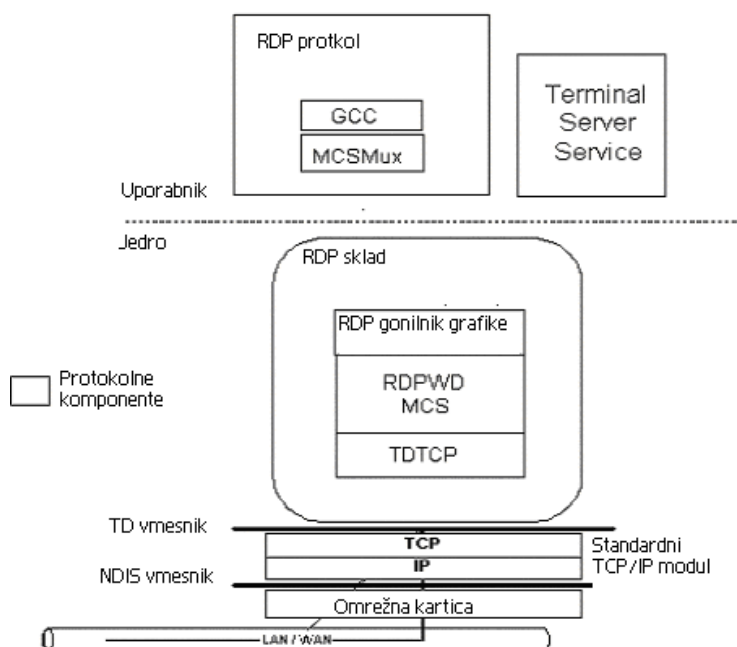
Trenutno se uporablja le en sam kanal za prenos tipkovnice, miške in prezentacijskih podatkov oz. uporabniškega vmesnika. RDP naj bi podpiral več

različnih transportnih protokolov, kot npr.: IPX, NetBios itd., vendar trenutna verzija RDP protokola podpira le TCP/IP.

Pri pošiljanju in prejemanju podatkov skozi RDP sklad potekajo aktivnosti kot je definirano pri 7 plastnem OSI (Open Systems Interconnection) modelu. Podatki določene aplikacije oz. storitve se posredujejo preko protokolnih skladov, kjer se kriptirajo, zavijejo, okvirijo in zapakirajo na omrežni protokol. Od tam se pošljejo odjemalcu.

Pri prejemanju podatkov se vrstni red aktivnosti obrne, tako da se paket odvije, odkriptira itn., dokler podatki ne pridejo do končne aplikacije oz. storitve. RDP protokol nastopa med 4. in 7. plastjo OSI modela, kjer se podatki kriptirajo, ovijajo in okvirjajo.

Slika 17: Komponente RDP protokol



Vir: Microsoft, 2000a, str 7.

RDPDD (RDP Display Driver), gonilnik za prikaz grafike pretvori spremembe Windows uporabniškega vmesnika v obliko, ki jo razume RDPWD (RDP Winstation Driver), ki nato sprejete informacije pakira in pripravlja za prenos do odjemalca po enem od kanalov T.120 MSC protokola. Ta plast lahko komunicira s katerikoli protokolom, ki se nahaja pod njo, trenutno je to TCP/IP.

V nasprotnem primeru odjemalec pošilja RDP kodirane pakete, ki vsebujejo informacije o vhodnih enotah (miška in tipkovnica). RDPWD jih sprejme, razpakira in posreduje določeni uporabnikovi seji.

Kot je razvidno, je koncept RDP protokola neodvisnost od transportne plasti sedemplastnega OSI modela, kar pomeni, da lahko po potrebi to plast

zamenjamo. Trenutno je ta plast TCP/IP, v prihodnosti pa lahko pričakujemo tudi druge protokole.

Posebne izboljšave pri prenosu grafičnih objektov Windows uporabniškega vmesnika so predvsem naslednje:

- **Inteligentno kodiranje** (angl. Intelligent encoding). Ponovno prikazovanje grafičnih objektov se tako zakodira, da odjemalec ponovno osveži le tiste točke, ki so bile spremenjene. Na ta način se pošiljajo samo spremembe.
- **Hranjenje slik in črkovnih oblik** (angl. Glyph and Bitmap Caching). Vsak RDP odjemalec rezervira 1,5 megabajtov pomnilnika z namenom, da bo v tem prostoru hranil črkovne oblike in slike. Črkovne oblike predstavljajo pisavo, ki se uporablja v določenem objektu. Strežnik odjemalcu pošlje vse potrebne črkovne oblike, ki jih bo ta rabil za izpis določenega besedila. V nadaljevanju odjemalec od strežnika dobi le tekst, ki ga nato s pomočjo črkovnih oblik izriše na ekranu. Na enak način se hranijo slike (npr. ikone), ki se na ta način samo enkrat pošljejo do odjemalca.
- **Premiki oken** (angl. Screen-to-Screen transfers). Napredne grafične kartice pospešujejo prikazovanje grafike s kopiranjem vsebin določenih območij na druga področja. Podobno tehniko uporablja RDP odjemalec, kot je npr. razvidno pri premikanju določenega aplikacijskega okna. Okno se premakne takoj, kajti ukaz strežnika je vseboval le nove koordinate okna, ne pa ponoven izris ekrana.
- **Kompresiranje podatkov** (Bulk compression). Po želji se lahko vklopi kompresija podatkov, ki lahko zmanjša število RDP paketov do 50 odstotkov.

Kot je že omenjeno, RDP uporablja napreden algoritem za kodiranje grafičnih podatkov, ki je podoben Unixovem X Window protokolu. Vsi splošni oz. ponavljajoči izrisi se pošiljajo kot ukazi in ne kot grafika. Ta metoda izredno zmanjša količino podatkov, ki je potrebna za prikaz novega ekrana oz. osvežitev trenutnega. Podjetje Microsoft je objavilo omrežne pasovne zahteve, ki so potrebne za izris splošnega Windows okna. Laboratorijski testi so pokazali, da RDP porabi do 40 kilobitov na sekundo za zgoraj omenjeni izris Windows okna (Kaplan, 2000, str. 63).

Konkretno RDP obnavlja ekran po naslednjem postopku: Uporabnik zažene aplikacijo, ki javi GDI-ju (Graphical Device Interface), kje in kako naj nariše aplikacijsko okno. GDI preda ta ukaz RPDD-ju (RDP Device Driver) preko standardnih API knjižnic. To je podoben proces kot pri standardnih Windows NT

sistemih. Razlika je seveda v RPDD-ju, ki ukaze preda RDP protokolu, ta jih pa pošlje odjemalcu. Ukazi, ki jih posreduje RDP, so predvsem naslednji:

- **TextOut ()**. Ukaz omogoča prikaz določenega teksta na odjemalčevem ekranu. GDI informira RDP o lokaciji in glyphih (grafična oblika teksta, podobno kot font) teksta. RDP nato odjemalcu posreduje tekst in morebitne glyphe, ki jih ta še nima.
- **PatBlit ()**. Pattern Block Transfer se uporablja za risanje barvnih blokov. Tako preko omrežja potujejo le ukazi, ne pa celoten blok grafike, ki bi zasedal veliko pasovno širino.
- **LineTo ()**. Ukaz omogoča izris 3-dimenzionalne črte. Uporablja se za izris kvadratov in lahko vsebuje le 6 bajtov za celoten izris črte.

Z operacijskem sistemom Windows 2000, ki že v osnovi vsebuje TSE jedro, je Microsoft izdal tudi novo različico RDP protokola. To je verzija 5, ki v primerjavi z verzijo 4 vsebuje predvsem naslednje izboljšave:

- podpora lokalnim tiskalnikom,
- podpora lokalnim diskom,
- funkcija "Izreži in prilepi" med odjemalcem in strežnikom,
- senčenje,
- hranjenje slik in črkovnih oblik na disku,
- izboljšave odzivnosti (manjša pasovna širina).

Opazno je, da se Microsoft poskuša približati funkcionalnosti Citrixovega ICA protokola, vendar tudi RDP verzija 5 močno zaostaja za ICA protokolom. Še vedno je podprt samo TCP/IP kot transportni protokol, pa tudi odzivnost se ne more primerjati z ICA protokolom, ki z 20 kilobitno povezavo nudi zadovoljivo delovanje.

4.2.4 Odjemalec RDP

Odjemalec RDP je možno namestiti na vse Windows temelječe operacijske sisteme. To so predvsem Windows 3.11, Windows 95/98 in Windows NT/2000. V prihodnosti naj bi bil podprt tudi operacijski sistem Windows CE.

Odjemalec RDP je velik 130 kilobajtov (.exe) in uporablja 300 kilobajtov delovnega prostora in 100 kilobajtov za prikazovanje podatkov.

PRIJAVA NA STREŽNIK

Odjemalec posreduje strežniku zahtevo za prijavo preko TCP vrat. Strežnik nato ustvari novo sejo in z odjemalcem ureja nadaljnje podrobnosti povezave. Kot prvo

je potrebno definirati način zaščite oz. kriptiranja podatkov med odjemalcem in strežnikom. Ti so lahko naslednji:

- **Nizka zaščita** (angl. Low encryption). Kriptirajo se le paketi, ki potujejo od odjemalca do strežnika. To so lahko zaupni vnosni podatki kot npr. gesla in šifre.
- **Srednja zaščita** (angl. Medium encryption). Kriptirajo se poleg paketov, ki potujejo od odjemalca do strežnika, tudi paketi ki potujejo od strežnika do odjemalca. Uporablja se splošno znani kodirni algoritem RC4 z 40 bitnim ključem.
- **Visoka zaščita** (angl. High encryption). Visoka zaščita prav tako kot srednja kriptira pakete v obe smeri, ampak zato uporablja 128 bitni ključ.

Vsi odjemalci rezervirajo 1,5 megabajtov pomnilniškega prostora, ki ga uporabljajo za shranjevanje slik in črkovnih oblik. Tukaj se shranjujejo uporabljene pisave in slike kot npr. ikone, orodne vrstice, kurzorji ipd. Prostor se prepisuje z LRU (angl. Last recently used) algoritmom. Treba je dodati, da verzija 5 RDP protokola podpira tudi hranjenje na disku, tako da lahko prostor za hranjenje povečamo in hkrati naredimo statičnega.

Po definiciji načina kriptiranja se preveri licenčno stanje, kajti za vsak odjemalec, ki je priključen na strežnik je potrebno imeti Windows NT Server licenco. Če je na volji še dovolj prostih licenc se uporabniku dovoli priključitev na strežnik. Šele zdaj lahko uporabnik vpiše svojo uporabniško ime in geslo, ki ga strežnik preveri. V primeru, da že obstaja seja z uporabnikovo domeno in imenom, se uporabniku prikaže prav ta seja. Če je takih sej več, lahko izbira med njimi.

IZVAJANJE APLIKACIJ

Po prijavi na strežnik se uporabniku prikaže standardno Windows namizje. Ko uporabnik izbere (dvoklikne) določeno aplikacijo, se ta ukaz miške prenese do Terminal Serverja, kjer se določena aplikacija naloži v navidezni pomnilniški prostor. Navidezni prostor postane del uporabnikove seje in prav tako vsi procesi, ki bodo ustvarjeni znotraj aplikacije.

Vsi procesi znotraj Terminal Serverja si delijo isti jedrski in uporabniški prostor, kot je že bilo opisano v prejšnjem poglavju. Da bi dosegli deljenje programskih modulov med procesi se uporablja t.i. 'Copy-on-write' tehnika. Vsi procesi uporabljajo enak pomnilniški prostor ne glede na identifikator seje. Kadar določen proces spreminja skupni pomnilniški prostor, ga VMM (Virtual Memory Manager) prekopira na drugo lokacijo in šele tam spremeni. Proces več ne uporablja skupnega pomnilniškega prostora, ampak spremenjeno kopijo le-tega. 'Copy-on-

write' tehnika se je izkazala kot zelo učinkovita, kajti pri večih priklopljenih uporabnikih lahko prihranimo ogromno virov.

Treba je omeniti, da se 'Copy-on-write' tehnika uporablja le na 32-bitnih aplikacijah. Windows NT dovoli 16-bitnim aplikacijam, da tečejo v svojem VDM emulatorju (Virtual Dos Mashine), kjer pa ni možno uporabljati zgornje tehnike. VDM emulator poleg tega zasede ogromno procesorskega časa, kajti vse 16-bitne operacije je treba prevesti v 32-bitne, kar zahteva procesorsko moč. Poraba virov 16-bitne aplikacije je lahko do 40 odstotkov večja kot pri porabi 32-bitne aplikacije. Zaradi tega uporaba 16 bitnih aplikacij ni priporočljiva, čeprav je podprta.

PREKINITEV SEJE

Pri prekinitvi uporabnikove seje se vsi podatki o seji prepíšejo na disk, tako da se sprostí fizični pomnilnik. Terminal Server si zapomni uporabnika in njegovo domeno, tako da lahko naslednjič prekinjeno sejo zopet naloži in prikaže uporabniku. Omogočeno je tudi zamenjati resolucijo ekrana med prekinitvijo in ponovno prijavo na določeno sejo. Prav tako se uporabnikova seja prekine, kadar strežnik izgubi povezavo z odjemalcem, kar v praksi pomeni, da nestabilnost odjemalca oz. omrežne povezave ne more ogrožati opravljenega dela v tekoči seji. V primeru, da izgubimo omrežno povezavo, se kasneje enostavno ponovno priključimo na strežnik, ki bo takoj ugotovil, da obstaja prekinjena seja z našim imenom in domeno ter nam to sejo ponudil.

ODJAVA S STREŽNIKA

Odjava s strežnika je ponavadi enostaven proces. Ko se uporabnik odjavi, se vsi procesi, ki so povezani z uporabnikovo sejo, zaključijo in ves pomnilniški prostor se sprostí. Tukaj je seveda treba upoštevati 'Copy-on-write' tehniko, ki omogoča deljenje programskih modulov med sejami. V primeru, da si več sej deli določen pomnilniški prostor, se ta prostor ne bo sprostil, dokler je seja, ki ga uporablja, še aktivna.

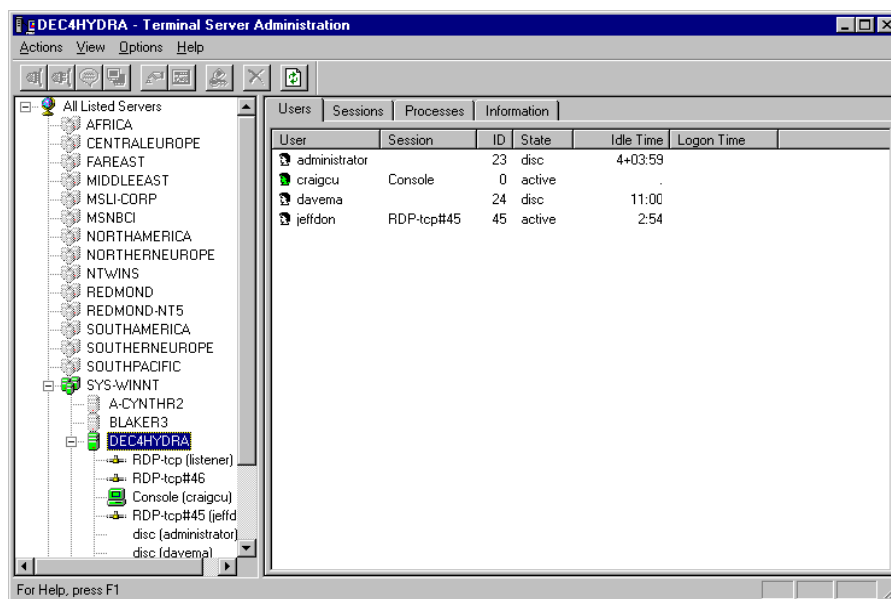
4.2.5 Orodja za administracijo

Za upravljanje Terminal Serverja so nam voljo razna orodja. Najbolj pomembna bodo predstavljena v nadaljevanju.

- **Terminal Server Administration Utility.** To je osrednje orodje za administracijo strežnika. Prikazuje in upravlja tekoče seje na vseh Terminal Serverjih, ki so v naši organizaciji. Med funkcionalnosti, ki nam jih nudi so:
 - prekinitve določene seje,
 - pošiljanje opozorila določeni seji,

- resetiranje seje,
- prikaz statusa seje,
- prikaz informacij o odjemalcu,
- prikaz 'user' in 'system' procesov,
- prisilno končanje seje,
- informacije o Terminal Serverju.

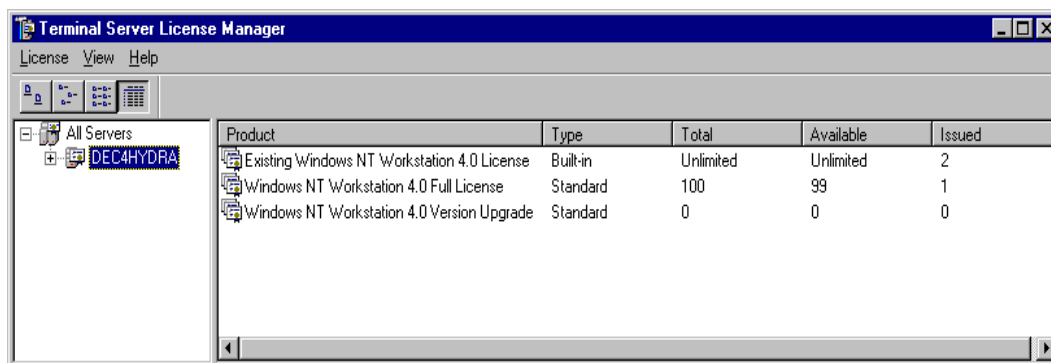
Slika 18: Terminal Sever Administration Utility



Vir: Microsoft, 2000b, str. 12

- **Terminal Server Licence Manager.** Omogoča administratorju sistema pregled nad tem, katere licence so v uporabi in koliko jih je še na voljo. Terminal Server RDP odjemalci potrebujejo dve vrsti licenc. Prva je standardna Microsoft Client Access Licence (CAL) za dostop do Windows NT datotek in tiskalnikov, druga pa je Windows NT Workstation licenca.

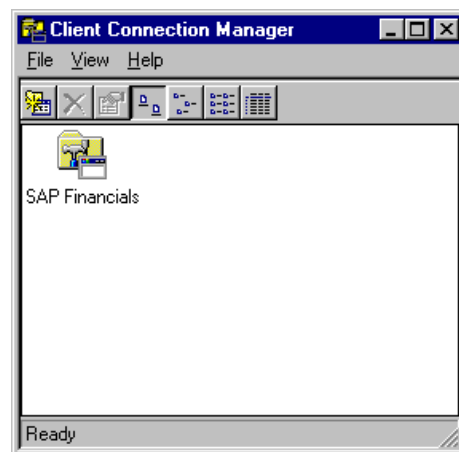
Slika 19: Terminal Sever Licence Manager



Vir: Microsoft, 2000b, str. 10

- **Distributed File System support (DFS).** Omogoča priključitev na DFS sistem in omogoča skupno rabo imenikov na Terminal Serverju preko DFS sistema. S sistemom DFS lahko kreiramo drevo imenikov, ki je sestavljeno iz imenikov različnih strežnikov. Na ta način lahko na enem mestu prikazujemo podatke, ki so razpršeni po večih strežnikih. DFS imenikom lahko damo logična imena, kar dodatno olajša delo z njimi.
- **Client Connection Manager.** To orodje omogoča administratorju sistema in končnim uporabnikom, da definirajo priključitve na Terminal Serverje oz. neposreden zagon določene aplikacije na Terminal Serverju. Ustvari se ikona, s katero se lahko zaženemo aplikacijo, ne da bi se morali prijaviti na Terminal Server in aplikacijo zagnati iz tamkajšnjega namizja.

Slika 20: Neposreden dostop do aplikacije preko Client Connection Managerja



Vir: Microsoft, 2000b, str. 10

- **Terminal Server Client Creator.** Orodje nam omogoča izdelavo namestitvenih disket z RDP odjemalci za različne tipe odjemalcev. Ti so lahko Windows 3.11, Windows 95,98 in Windows NT.

Microsoft Terminal Server nam nudi osnovna orodja za upravljanje. V primerjavi z Citrixovim Metaframe manjkajo orodja kot npr. objavljanje aplikacij, vstavljanje aplikacij v spletne strani, upravljanje z resursi ipd. Seveda pa je možno Windows TSE strežnik nadgraditi z Citrix ICA MetaFrame strežnikom.

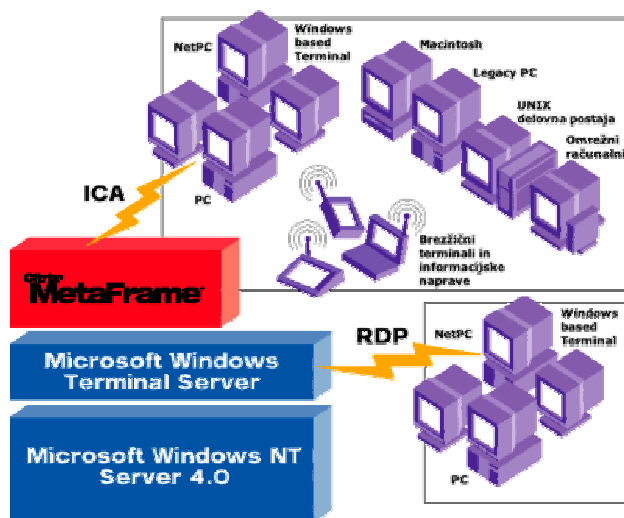
V katerih primerih je nadgradnja Microsoft TSE strežnika s Citrix MetaFrame dodatkom smotrna, bo opisano naslednjem poglavju.

4.3 Kriteriji za izbiro med Microsoft Terminal Serverjem in Citrix MetaFrame strežnikom

V prejšnjem poglavju sem opisal večuprabiško jedro MultiWin, ki omogoča Windows NT strežniku izvajanje več sej hkrati. MultiWin jedro je enako pri Terminal Serverju in MetaFrame strežniku, saj ga je Microsoft licenciral od Citrixa. Razlika med MetaFrame strežnikom in Microsoft Terminal Serverjem je torej v prenosnih protokolih RDP in ICA ter posledično v programski opremi za odjemalce.

Citrix MetaFrame sedaj lahko kupimo kot nadgradnjo Windows NT Terminal Serverja oz. Windows 2000 Advanced Serverja.

Slika 21: MetaFrame kot dodatek Terminal Serverju



Vir: Citrix, 1998

Kot je razvidno s slike 21, lahko preko RDP protokola do Terminal Serverja dostopajo le naprave, ki uporabljajo Windows operacijski sistem. Šele dodatek MetaFrame z ICA protokolom nam omogoča uporabo standardnih Win32 aplikacij iz naprav kot so: Unix delovne postaje, Macintosh računalniki, Pda-ji (Ipaq), Intel 286 računalniki z operacijskim sistemom DOS.

RDP in ICA protokol opisujeta strukturo pete in šeste plasti OSI sedemplastnega omrežnega modela. Transportni protokoli kot npr. TCP/IP pa so na tretji in četrti plasti. Kljub temu RDP protokol podpira le TCP/IP protokol. ICA podpira IPX/SPX, NetBEUI in asinhrono povezave. To je prednost v velikih organizacijah, kjer še vedno najdemo veliko Novell NetWare strežnikov, ki lahko uporabljajo IPX/SPX protokol.

Oba protokola podpirata osnovne systemske avdio zvoke, kot je standardno piskanje osebnih računalnikov. Vendar samo ICA podpira izvajanje WAV

(Windows Wave tehnologija avidopodatkov) datotek in 16-bitno SoundBlaster podporo v določenih sejah. Čeprav prenos zvoka ni pglavitni element pri lahkih odjemalcih, pa vendar omogoča dobro osnovo za prenos zvoka in slike, ki postaja vedno bolj popularno.

Lokalno tiskanje se z uporabo ICA odjemalca izkaže kot zelo enostavno, saj ICA odjemalec lokalne tiskalnice samodejno preslika v tekočo sejo. Pri RDP odjemalcu mora to opravilo izvesti uporabnik.

Preslikava lokalnih pogonov se prav tako izvede samodejno z uporabo ICA odjemalca. Uporabniki vidijo ikono 'My Computer', preko katere dostopajo do lokalnih pogonov, ponavadi "C:" in "D:". Pri RDP odjemalcu pa je potrebno lokalne pogone predati v skupno rabo in jih nato znotraj seje preslikati, kar seveda predstavlja potencialne varnostne luknje, saj lokalni pogoni postanejo vidni vsem omrežnim uporabnikom. Tabela 3 je prikazuje primerjavo med Microsoft RDP in Citrix ICA protokolom:

Tabela 3: Primerjava med RDP in ICA protokolom

Lastnost / Funkcionalnost	Opis	RDP	ICA
Odjemalci	Odjemalec za Windows 3.11, Windows 95/98, Windows NT/2000,	✓	✓
	Odjemalec za MS-DOS		✓
	Odjemalec za Unix delovne postaje		✓
	Odjemalec za Macintosh računalnike		✓
	Odjemalec za brskalnik (Internet Explorer, Netscape)		✓
	Odjemalec za Java (JVM)		✓
Transportni protokol	TCP/IP	✓	✓
	SPX, IPX, NetBEUI, direktno asinhrono		✓
Preslikava imenikov in tiskalnikov	Samodejna preslikava lokalnih imenikov in tiskalnikov odjemalca		✓
Avdio	Podpora osnovnim zvokom	✓	✓
	Podpora .wav datotekam		✓
Spletno objavlanje aplikacij	Objavljanje aplikacij v spletnih portalih (Citrix NFuse)		✓
Senčenje	Senčenje omogoča administratorju priklop in sodelovanje pri uporabnikovi seji	✓*	✓
Load Balancing v grućah	Load Balancing omogoća povezovanje strežnikov v gruće in tako razporeditev obremenjenosti med strežnike		✓
Samodejno obnavljanje odjemalca	Samodejno obnavljanje programske opreme na odjemalcu		✓

* Napovedano v novih verzijah RDP protokola

Vir: Kaplan, 2000, str. 134

Treba je še poudariti, da ICA protokol v primerjavi z RDP protokolom porabi manjšo pasovno širino. Zadovoljiva naj bi bila 20 kilobitna povezava, medtem ko RDP odjemalec zahteva vsaj 40 kilobitno povezavo za zadovoljivo delo (Kaplan, 2000, str. 63).

Tabela 3 nam prikazuje superiornost ICA protokola nad RDP protokolom. Vendar obstaja kategorija, kjer Citrix ICA podleže. To je priporočena cena \$5000 ameriških dolarjev za 10 uporabnikov (Reeser, 1999). RDP protokol dobimo skupaj s Terminal Serverjem.

Da bi ugotovili, ali naj se odločimo za dodatek Citrix MetaFrame, moramo vedeti kaj želimo doseči. V nadaljevanju bo opisanih nekaj tipičnih scenarijev, kjer se moramo odločati za RDP oz. ICA odjemalce (Reeser, 1999).

- **Scenarij 1.** Veliko podjetje ima nekaj Unix in Macintosh postaj, na katerih želi izvajati Windows aplikacije.

Rešitev: RDP odjemalci ne bodo dovolj, saj podpirajo samo Windows operacijske sisteme. V tem primeru je potreben dodatek MetaFrame z ICA odjemalci.

- **Scenarij 2.** Majhno podjetje ima 10 osebnih računalnikov povezanih v lokalnem omrežju. Uporabniki uporabljajo Corel Office Suite aplikacije, Netscape in računovodsko aplikacijo, ki je bila razvita v Oracle Forms.

Rešitev: Podjetju bo ustrezal RDP odjemalec Terminal Serverja. Finančno ne bi mogli upravičiti nabavo MetaFrame dodatka.

- **Scenarij 3.** Velika zavarovalnica ima več kot 100 agentov na terenu. Agenti imajo prenosni računalnik, tiskalnik in modem. Želja je, da se lahko prijavijo na centralni informacijski sistem in tam ažurirajo svoje opravljeno delo.

Rešitev: Zavarovalnica bi lahko uporabljala le RDP odjemalce, vendar je zahtevana pasovna širina dokaj velika glede na to, da bodo agenti uporabljali modemske povezave do zavarovalnice. Prav tako RDP odjemalci lokalnih tiskalnikov ne preslikajo takoj, tako da bi bilo potrebno dodatno šolanje zavarovalnih agentov. Glede na veliko število uporabnikov bi bil koristen tudi Load Balancing za MetaFrame farme, ki bi nove uporabnike samodejno preusmerjal na najmanj obremenjen strežnik. Zavarovalnica bi se zaradi performančnih dejavnikov morala odločiti za MetaFrame.

- **Scenarij 4.** Podjetje se je odločilo za nakup 200 Windows Terminalov, preko katerih bodo uporabniki uporabljali standardne Office aplikacije in Oracle aplikacije.

Rešitev: Prav tako kot v prejšnjem primeru bi podjetje lahko izbralo Terminal Server z RDP odjemalci. Pri 200 odjemalcih bi bilo potrebnih vsaj 5 Terminal Serverjev, kar pa predstavlja precej administracijskega dela. Zelo primerno bi bilo MetaFrame Load Balancing orodje za več strežnikov, povezanih v farmo in Installation Management Service. Tudi v tem primeru bi bil priporočljiv Citrix MetaFrame dodatek k Terminal Serverju.

- **Scenarij 5.** Podjetje ima 15 internih uporabnikov, ki uporabljajo samo Office aplikacijo Word for Windows, in odjemalca za elektronsko pošto Outlook Express. Vsi uporabniki uporabljajo Windows terminale.

Rešitev: V tem primeru bo podjetju zadostoval Windows Terminal Server z RDP odjemalcem. Uporabnike bo treba izšolati o tem, kako preslikajo svoj lokalni tiskalnik v Terminal Server sejo. Čeprav Citrix ICA odjemalec to uredi samodejno, to še ne bi upravičilo finančnega stroška za njegovo nabavo.

Kot je razvidno iz zgornjih primerov, so pri izbiri med RDP odjemalci in ICA odjemalci odločilnega pomena kriteriji, ki jih prikazuje tabela 4.

Tabela 4: Kriteriji za izbiro RDP oz. ICA odjemalca

Kriterij	Microsoft Terminal Server (RDP)	Citrix MetaFrame (ICA)
Naprave, s katerimi se želimo povezati z aplikacijskim strežnikom	Zahtevan je operacijski sistem Windows in protokol TCP/IP	Operacijski sistemi Windows in drugi ter razni protokoli (glej tabelo 3, na str.46)
Število uporabnikov, ki se bo povezovalo z aplikacijskim strežnikom	Priporočljivo do 10 uporabnikov oz. en strežnik. Pri večjemu številu strežnikov postane administracija zahtevna	Nad 10 uporabnikov in več strežnikov. Pri večjem številu strežnikov postanejo administracijska orodja MetaFrame strežnika nepogrešljiva
Pasovna širina, ki je na voljo med strežniki in odjemalci	Večja od 40 kb/s na uporabnika	Vsaj 20 kb/s na uporabnika, kar omogoča povezovanje do strežnika celo preko GSM telefonov

Priporočljivo je, da se v primeru možnosti uporabe RDP protokola pilotsko testira tudi Citrix MetaFrame z ICA protokolom, kajti performančne razlike so v določenih primerih lahko velike (Reeser, 1999).

5 PRIMER UPORABE LAHKIH ODJEMALCEV: FT PALMS ASIA

V tem poglavju bom opisal postopek uvajanja lahkih odjemalcev s tehnologijo Citrix MetaFrame na primeru projekta Financial Times PALMS - Asia. Kot je že omenjeno, so lahki odjemalci predvsem primerni za organizacije, ki imajo dislocirane enote. Prav takšno podjetje je Financial Times Ltd., ki ima sedež v Londonu. Financial Times Ltd. urejuje in izdaja najbolj razširjeni poslovni dnevnik Financial Times, ki je vsebinsko primerljiv s slovenskim dnevnikom Finance, vendar pokriva gospodarska dogajanja po celem svetu in dosega dnevno pol milijonsko naklado. Ker Financial Times izhaja po celem svetu, so tudi poslovalnice tega podjetja razpršene po vseh finančnih metropolah sveta.

Predstavništva Financial Timesa lahko najdemo v New Yorku, Frankfurtu, Hong Kongu in Tokiu. Funkcija teh poslovnih enot je, poleg zbiranja poslovnih novosti in trženja oglasnega prostora, predvsem pridobivanje novih naročnikov, komunikacija z naročniki in logistika distribucije dnevnika.

Vsaka poslovalnica je pred uvedbo novega informacijskega sistema uporabljala avtonomni informacijski sistem za distribucijo dnevnika, ki je deloval le v lokalnem omrežju poslovalnice in je bil razvit v razvojnem okolju Microsoft Foxpro. Sistem je deloval po načinu debelega odjemalca in je bil precej zastarel, poleg tega pa ga ni bilo možno dograjevati naprej, ker je podjetje Microsoft ukinilo vso podporo za orodje Foxpro, pa tudi razvijalce je bilo težko pridobiti. Zaradi tega je bil razvit nov informacijski sistem za distribucijo časopisa PALMS (Print and Logistic Management System), ki deluje v okolju Magic in uporablja relacijsko bazo podatkov Oracle. Sistem PALMS je bil že od samega začetka načrtovan tako, da naj bi podpiral vse poslovalnice Financial Timesa. To naj bi zagotavljalo centralni pregled nad naročniško bazo in seveda omogočalo kvalitetnejše poizvedovanje na podlagi trenutnih in bodočih prodajnih podatkov.

Po uspešni uvedbi novega informacijskega sistema na sedežu podjetja v Londonu, je bilo treba razširiti uporabo sistema na ostale poslovalnice. Prvi dve poslovalnici naj bi bili Hong Kong in Tokio. Posamezne poslovalnice so povezane s sedežem podjetja s Frame Relay omrežno povezavo. Da bi omogočili azijskim poslovalnicam priključitev do centralnega sistema v Londonu, so se v podjetju Financial Times odločili za uporabo tehnologije lahkih odjemalcev.

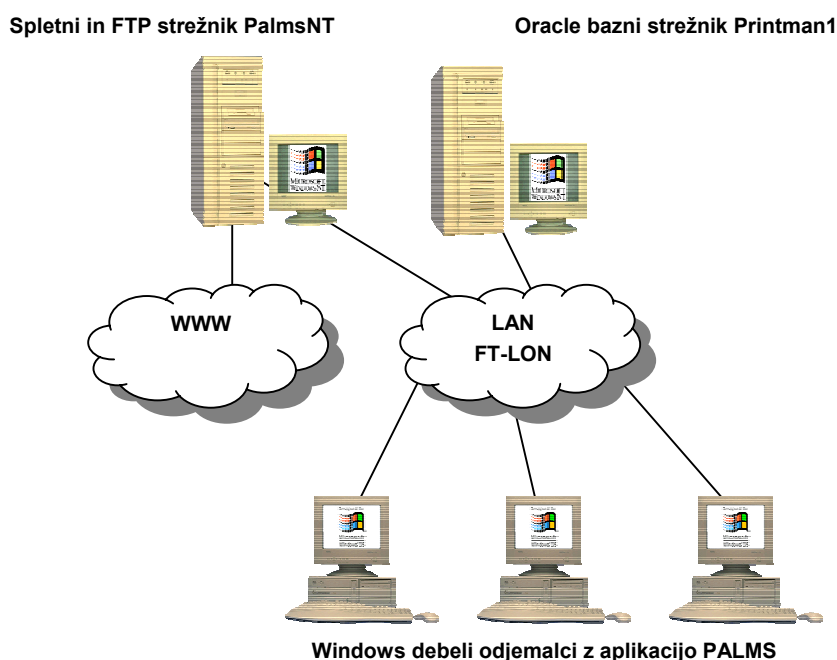
Projekt, ki je bil namenjen lokalizaciji aplikacije PALMS in implementaciji lahkih odjemalcev v Tokiu in Hong Kongu, je bil poimenovan PALMS – Asia in bo opisan podrobneje v naslednjih poglavjih.

5.1 Opis informacijskega sistema FT PALMS

V tem poglavju bom opisal informacijski sistem Financial Times Print and Logistic Management System (v nadaljevanju PALMS), pri razvoju katerega sem osebno sodeloval v vlogi Oracle konzultanta. Prisoten sem bil predvsem pri lokalizaciji sistema za azijske poslovalnice ter testiranju in prilagajanju aplikacije za delovanje na MetaFrame strežniku.

PALMS je po arhitekturi debeli odjemalec / strežnik aplikacija, ki je bila razvita v orodju Magic za Windows izraelskega proizvajalca MSE in podatkovni bazi Oracle na AIX Unixu. Potrebno je dodati, da PALMS ni klasična debeli odjemalec / strežnik aplikacija, saj se določen del procesiranja odvija na Oracle strežniku, kjer se v veliki meri uporablja Oracleov PL/SQL programski jezik v obliki procedur in prožilcev (angl. triggers). Vendar se na odjemalcih generira veliki del izpisov in prenosnih datotek. Takšni procesi pa zahtevajo veliko pasovno širino med strežnikom in odjemalcem, tako da PALMS ni primeren za omrežja z ozko pasovno širino. Generirani izpisi, kot npr. 'nalog za tisk', se nato objavljajo na spletnem strežniku v obliki Adobe Acrobat PDF datotek, kjer so dosegljivi za tiskarne.

Slika 22: Arhitektura aplikacije PALMS

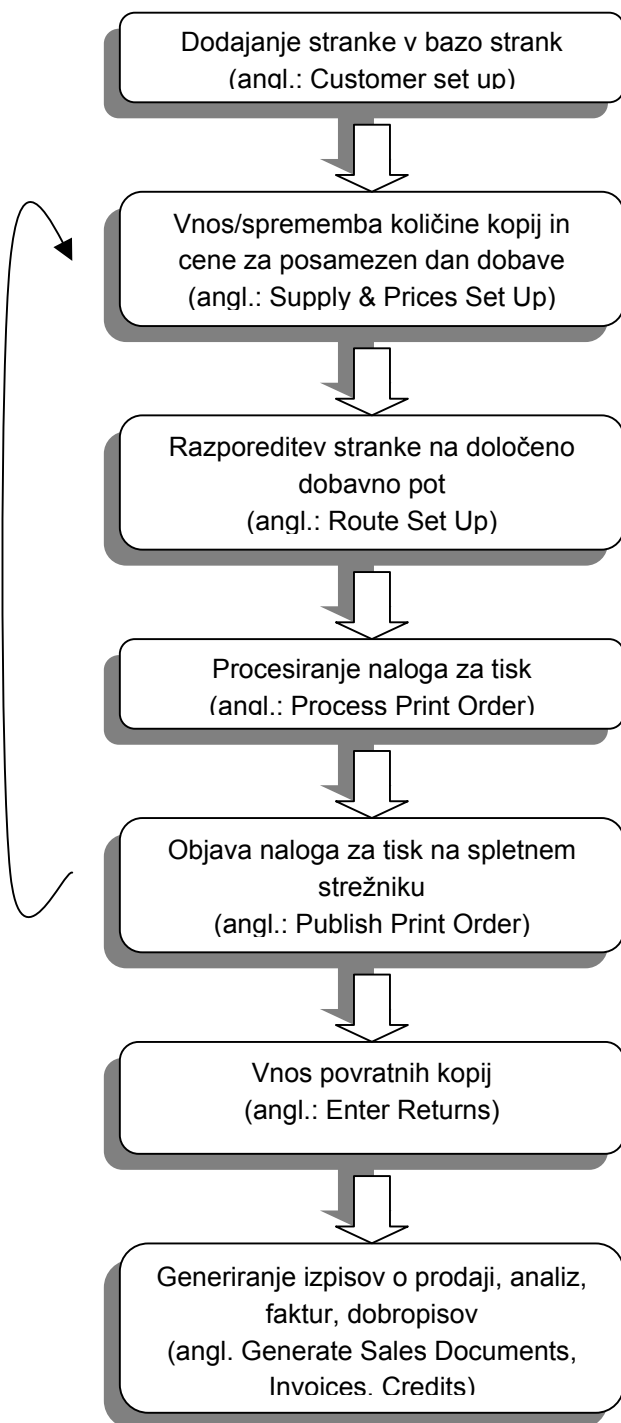


Vir: Financial Times, 2001

Palms pokriva vsa področja cirkulacije časopisa Financial Times, od baze naročnikov, naročila naročnikov in zgodovine dobave posameznega naročnika. Zagotovljen je zajem povratnih oz. neprodanih kopij, na podlagi česar se določijo

podatki o prodaji in bodoči realizaciji. Generirajo se vsakodnevni dokumenti za potrebe tiskarn, ki tiskajo časopis. Ti dokumenti so npr. 'nalog za tisk', ki vsebuje podatke o številu potrebnih časopisov za določenega naročnika na dan izida Financial Timesa. Sistem generira tudi račune in dobropise, ki se potem pošljejo naročnikom. Seznam izdanih računov in dobropisov se v elektronski obliki posreduje finančnemu oddelku, kjer ga prevzame računovodski informacijski sistem.

Slika 23: Zaporedje delovnih procesov v aplikaciji PALMS



Vir : Financial Times, 2001

Slika 23 prikazuje zaporedje delovnih procesov v aplikaciji PALMS. V nadaljevanju bo vsak proces na kratko opisan.

DODAJANJE NAROČNIKA V BAZO NAROČNIKOV

Osrednji del aplikacije je baza naročnikov. Pomemben podatek predstavlja naslov, na katerem bo naročnik prejemal izhode časopisa. Prav tako lahko določimo plačnika za naročnika, saj ni nujno da bo naročnik tudi plačnik, kot je to primer pri velikih podjetjih, ki prejemajo časopis na večih različnih lokacijah.

Slika 24: Ekranska maska za vnos naročnika

Vir: Financial Times, 2001

VNOS/SPREMEMBA KOLIČINE KOPIJ ZA POSAMEZEN DAN DOBAVE

Spreminjanje naročil količine izvodov je vsakdanje opravilo operaterjev v klicnem centru cirkulacijskega oddelka. Naročniki pokličejo in operaterju sporočijo, koliko kopij želijo prejeti določen dan.

Slika 25: Določevanje števila kopij posameznemu naročniku

Prod	Sub-Type	Type	Abc Type	Remarks
FT	RET	TR	NT	

Supply	1) Mon	2) Tue	3) Wed	4) Thu	5) Fri	6) Sat
This week	4900 14/08	5400 15/08	5450 16/08	5525 17/08	5350 18/08	7100 19/08
Next week	4900 21/08	5400 22/08	5450 23/08	5525 24/08	5350 25/08	7100 26/08
Last week	4900 07/08	5400 08/08	5450 09/08	5525 10/08	5350 11/08	7100 12/08

* This screen shows default supplies only. Where a supply is shown in red it will be overridden by a temporary supply that also exists.
Go to Customer Supply screen to see all supplies

Supply	Mon	Tue	Wed	Thu	Fri	Sat	Sun
TR	4900	5400	5450	5525	5350	7100	0

Exit Go to Customer supplies

Vir: Financial Times, 2001

RAZPOREDITEV NAROČNIKOV NA DOLOČENO DOBAVNO POT

Vsakega naročnika je potrebno razporediti na t.i. dobavno pot. To je pot, ki jo bo tovarnjak peljal iz določene tiskarne. V primeru da imamo naročnika blizu tiskarne v Londonu, ga torej razporedimo na dobavno pot, ki pelje iz Londona.

Slika 26: Razporeditev naročnika na dobavno pot

Route ID	Sequence	Sequence type	Drop point
COLL10	10	D THA-NEW	☒
COLL10	20	C THA-NEW	☐
COLL10	30	C SD-CIT	☐
COLL11	10	D NEW-FLA	☒
COLL11	20	C NEW-FLA	☐
COLL13	10	D JOH-EUR	☒
COLL13	20	C JOH-EUR	☐
COLL13	30	C JOH-EUR-P	☐
COLL21	10	D JM-GWH-1	☒
COLL21	20	C JM-GWH-1	☐
COLL21	30	C STA-DAR	☐
COLL21	50	C UNI-LH253	☐
COLL22	10	D WHS-CRO-E	☒

Delivery address:

THAMES NEWS SERVICES LTD
27-31 WEBBER STREET
LONDON
SE1 8QW

Audit

Vir: Financial Times, 2001

PROCESIRANJE NALOGA ZA TISK

Dnevna naloga uporabnikov je izdelava naloga za tisk. (angl. Print Order). To je dokument, ki vsebuje podatke o številu naročenih izvodov za določenega naročnika. Nalog za tisk se generira en dan pred izidom časopisa, tako da so

spremembe števila naročenih izvodov možne do 12. ure dneva pred izidom časopisa. Seveda je potrebno generirati nalog za tisk za vsako tiskarno posebej.

Slika 27: Procesiranje nalogov za tisk

Print order (CON)

Execution time		Delivery parameters					Oracle status	
Date	Time	Office	Prod	Issue date	Pattern_id	Print centre	Job ID	
26/07/2000	15:42:58	CON	FT	27/07/2000	MON-FRI	LON	709/1995	Waiting
26/07/2000	15:43:48	CON	FT	27/07/2000	MON-FRI	MIL	709/1995	Waiting
26/07/2000	15:43:12	CON	FT	27/07/2000	MON-FRI	JON	707/1994	Waiting
26/07/2000	15:42:34	CON	FT	27/07/2000	MON-FRI	MAD	706/1993	Waiting
26/07/2000	15:42:24	CON	FT	27/07/2000	MON-FRI	RBX	705/1992	Waiting
26/07/2000	15:43:54	CON	FT	27/07/2000	MON-FRI	FRA	704/1991	Waiting
25/07/2000	15:38:15	LON	FT	27/07/2000	MON-FRI2K	LEE	1947/1998	Done
25/07/2000	15:37:53	LON	FT	27/07/2000	MON-FRI2K	LON	1946/1997	Done
25/07/2000	15:00:00	LON	FT	26/07/2000	MON-FRI2K	LEE	1945/1996	Done
25/07/2000	15:00:00	LON	FT	26/07/2000	MON-FRI2K	LON	1944/1995	Done
25/07/2000	12:00:00	LON	FT	26/07/2000	MON-FRI2K	LEE	1943/1994	Done

More info

Change execution time

Exit

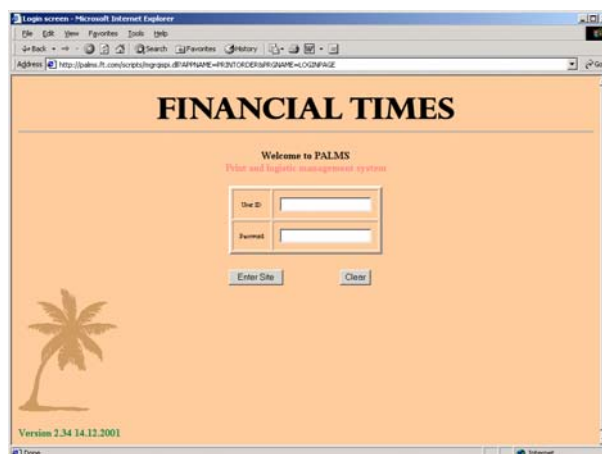
Vir: Financial Times, 2001

Procesiranje nalogov za tisk se izvaja v obliki PL/SQL procedure na Oracle strežniku, tako da uporabniki čakajo dokler se njihova paketna obdelava ne zaključi in nato nadaljujejo z objavo naloga za tisk.

OBJAVA NALOGA ZA TISK NA SPLETNEM STREŽNIKU

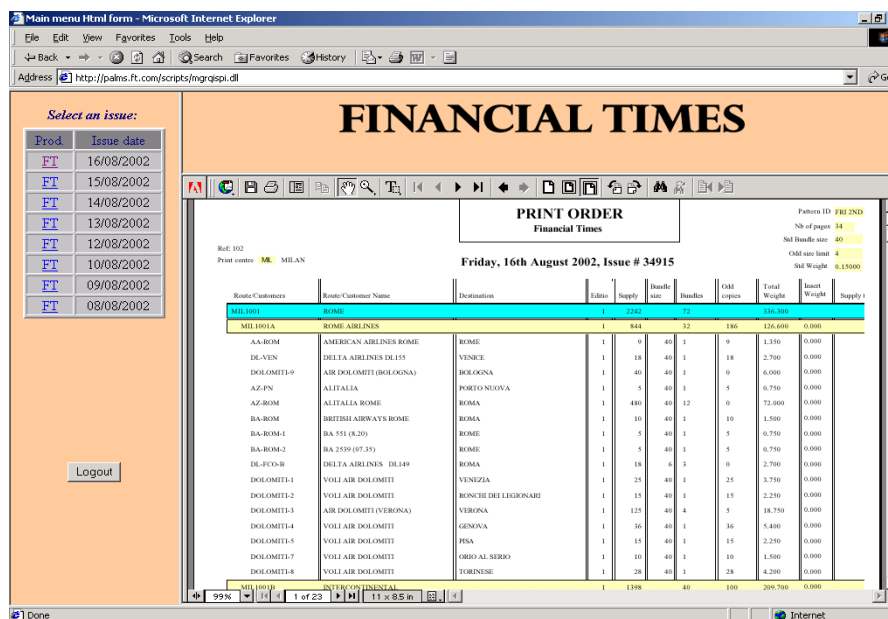
Po končani izdelavi naloga za tisk uporabniki preverijo njegovo ustreznost. Nalog za tisk si lahko ogledajo s programskim paketom Adobe Acrobat Reader, saj so vsi dokumenti, ki jih izdela PALMS, v PDF (Portable Document Format) formatu. Ko je nalog za tisk ustrezen, ga uporabniki lahko objavijo na spletnem portalu za tiskarne, od koder ga tiskarne vsak dan prenašajo.

Slika 28: Spletni portal za tiskarne



Vir: Financial Times, 2002

Slika 29: Predogled naloga za tisk



Vir: Financial Times, 2002

VNOS POVRATNIH KOPIJ

Naročniki Financial Timesa so poleg posameznikov večinoma distributerji časopisov, ki pa seveda ne prodajo vseh naročenih izvodov. Neprodane izvide vrnejo in jih ne plačajo. Naročniki število neprodanih izvodov javijo operaterjem, ki jih nato vnesejo v aplikacijo PALMS.

Slika 30: Vnos povratnih kopij

Manual returns

Product

Customer ID

Press F5 to zoom on customer list
Press Tab to move to the supply screen

Address:

W H SMITH & SON LTD
PO BOX 40
8 ACRE ROAD
READING, BERKS
RG2 0XZ

Returns

Issue date	Return date	Week	Supply sub-type	Total supplies	Returns	Qty correction
29/04/2000	03/05/2000	2000-17	RET ABC type: NT Supply type: TR	6950	1963	2031
28/04/2000	03/05/2000	2000-17	RET ABC type: NT Supply type: TR	5971	1313	
27/04/2000	03/05/2000	2000-17	RET ABC type: NT Supply type: TR	6445	1605	
26/04/2000	03/05/2000	2000-17	RET ABC type: NT Supply type: TR	6180	1568	
25/04/2000	03/05/2000	2000-17	RET ABC type: NT Supply type: TR	6144	1688	
24/04/2000	03/05/2000	2000-17	RET ABC type: NT Supply type: TR	4000	1646	

Vir: Financial Times, 2001

GENERIRANJE IZPISOV

Ob koncu meseca je potrebno izdelati fakture in dobropise, ki se nato pošljejo naročnikom. PALMS nudi izdelavo faktur in dobropisov ter ustreznih izpisov, ki so potrebni za finančni oddelek. Poleg finančnih izpisov je na razpolago veliko število analitičnih izpisov, s pomočjo katerih se ugotavlja prodaja po določenih področjih.

5.2 Opis in namen projekta FT – PALMS ASIA

Podjetje Financial Times je razvilo nov informacijski sistem za distribucijo časopisa PALMS, ki je bil uspešno implementiran na sedežu podjetja v Londonu. Kot je opisano, je sistem namenjen upravljanju distribucijske mreže, ki je potrebna da se dnevnik Financial Times dobavi vsakemu naročniku v najkrajšem času. Uporabnikom pomaga pri planiranju, analizi in izvedbi distribucijskega procesa. Uvedba sistema na sedežu podjetja je pripomogla avtomatizaciji poslovnih procesov in prihranjanju časa.

Projekt Financial Times – PALMS ASIA je bil namenjen implementaciji informacijskega sistema PALMS v poslovalnicah Hong Kong in Tokio. Azijske poslovalnice so bile izbrane zaradi majhnosti v primerjavi z New Yorkom oz. Frankfurtom, saj je število uporabnikov v Hong Kongu le šest, v Tokiu pa jih je sedem. Po uspešni implementaciji v azijskih poslovalnicah naj bi sledile poslovalnice v New Yorku in Frankfurtu. Pričakovane poslovne koristi projekta so naslednje (Financial Times, 2001):

- pridobitev centralnega informacijskega sistema,
- pridobitev centralne naročniške baze,
- povečati prodajo časopisa z bolj učinkoviti distribucijo le-tega,
- pregled prodaje po celem svetu,
- boljši pregled finančnih tokov.

Naslednje faze predstavljajo glavna opravila, ki jih je bilo treba opraviti v okviru projekta:

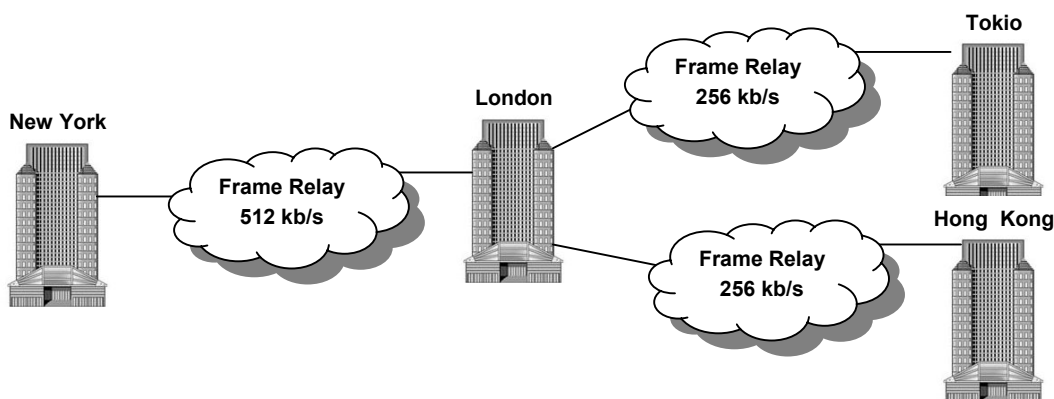
- ugotovitev razlik poslovnih procesov London – Asia za nadaljnjo fazo lokalizacijo aplikacije,
- lokalizacija informacijskega sistema PALMS za potrebe azijskih uporabnikov,
- migracija podatkov iz starega informacijskega sistema,
- testiranje informacijskega sistema s strani uporabnikov,
- potrditev ustreznosti sistema s strani uporabnikov,

- implementacija lahkih odjemalcev,
- testiranje sistema z uporabo lahkih odjemalcev,
- potrditev ustreznosti uporabe lahkih odjemalcev s strani uporabnikov,
- vzporedna produkcija (paralelna s starim sistemom hkrati),
- primerjava podatkov nov – stari sistem in potrditev ustreznosti s strani uporabnikov,
- izklop starega sistema.

Razvidno je, da je bila prva polovica projekta namenjena lokalizaciji informacijskega sistema azijskim uporabnikom, v katero se ne nameravam poglobljati podrobneje, saj bi to preseгло okvire magistrske naloge. Razlike so se pojavile predvsem na finančnem modulu sistema, kajti lokalni davčni sistem v Hong Kongu in Tokiu je seveda drugačen od angleškega. Nekaj sprememb je bilo potrebnih tudi pri določenih izpisih, vendar je večina sistema ustrezala potrebam uporabnikov, tako da dodaten razvoj ni zahteval veliko virov. Tudi migracija podatkov iz starega informacijskega sistema je bila izvršena brez večjih težav, saj Foxpro uporablja standardne datoteke, s katerimi se je enostavno povezovati.

Za implementacijo lahkih odjemalcev je bila odločilna predvsem omrežna povezava poslovalnic z informacijskim centrom v Londonu ter število uporabnikov. Poslovalnice New York, Hong Kong in Tokio so povezane s sedežem v Londonu preko Frame Relayja z 256 oz. 512 kb/s pasovno širino za prenos podatkov, kajti vsi uporabljajo skupen strežnik za elektronsko pošto in dokumente Lotus Notes v Londonu.

Slika 31: Frame relay povezave med poslovalnicami Financial Times



Vir: Financial Times, 2001

Frame Relay je izboljšana različica starejših paketno preklonih topologij, kot npr. X.25. Frame Relay zapakira vsak paket v okvir (angl. frame), ki vsebuje tudi naslov paketa. V nasprotju z X.25 protokolom obravnava vsak paket kot

brezhiben. V primeru defektnega paketa lahko določeno vozlišče zahteva ponovno pošiljanje paketa od vsakega prejšnjega vozlišča. Takšno preklapljanje okvirov (angl. frame switching) omogoča povečane performanse. Frame Relay je implementiran na prvih dveh slojih OSI modela in omogoča pasovno širino prenosa podatkov do 1544 megabitov na sekundo.

V primeru oddaljenih poslovalnic pasovna širina 256 kb/s seveda ne omogoča uporabe klasične arhitekture debeli odjemalec / strežnik. Število hkratnih uporabnikov je v Hong Kongu 6, v Tokiu pa 7, kar je približno 55 oz. 65 kb/s na uporabnika. Zaradi tega smo se v podjetju Financial Times odločili za uporabo lahkega odjemalca Citrix MetaFrame. Pri primerjavi med Microsoft Terminal Serverjem in Citrix MetaFrame je bila izbira za dodatek MetaFrame narejena na podlagi kriterijev, ki sem jih opisal v poglavju 4.3 (glej str. 48, tabela 4). Odločilni razlogi so bili predvsem naslednji:

- Citrix ICA protokol deluje zadovoljivo tudi z 20 kb/s pasovne širine za prenos podatkov na uporabnika.
- V podjetju je veliko Macintosh računalnikov, za katere obstaja Citrix ICA odjemalec.
- Povezovanje strežnikov v farme. Zaradi kasnejšega širjenja informacijskega sistema na druge poslovne enote po svetu bo število uporabnikov naraslo. Tako bo potrebno dodajati strežnike v farmo, da bodo zadovoljevali večjo število uporabnikov.
- Citrix MetaFrame omogoča neposredno modemsko povezavo do strežnika. To je v primeru izpada omrežne povezave med oddaljenimi enotami in centralo seveda ključnega pomena.

Po zaključeni fazi lokalizacije aplikacije in testni migraciji podatkov smo pričeli z implementacijo lahkih odjemalcev.

5.3 Proces implementacije lahkih odjemalcev

V tem poglavju bom opisal, kateri koraki so potrebni za implementacijo Citrix MetaFrame lahkih odjemalcev. Pri vseh korakih bom navedel izkušnje iz projekta PALMS - Asia.

5.3.1 Analiza

Preden začnemo z implementacijo moramo seveda analizirati problematiko, ki jo želimo rešiti z uporabo lahkih odjemalcev. To je bilo opisano že v prejšnjem

poglavju vendar nam bodo pri kasnejšem planiranju pomagali odgovori na naslednja vprašanja (Kanter, 1998, str. 121).

- Kakšen poslovni problem želimo rešiti ?
Namen našega projekta je, omogočiti oddaljenim poslovalnicam v Hong Kongu in Tokiu uporabo centralnega informacijskega sistema, ki se nahaja na sedežu podjetja v Londonu.
- Katere aplikacije želimo uporabljati ?
Uporabljala naj bi se aplikacija PALMS.
- Koliko uporabnikov bo uporabljalo aplikacije na MetaFrame strežniku ?
Šest uporabnikov v Hong Kongu in sedem uporabnikov v Tokiu, skupaj 13 uporabnikov.
- Kakšno povezavo imajo uporabniki z MetaFrame strežnikom ?
Poslovne enote v Hong Kongu in Tokiu so povezane z Londonsko centralo z Frame Relay povezavo, tako da so Windows NT domene FT-LON, FT-TOK in FT-HK med sabo povezane. Vsaka poslovna enota razpolaga z 256 kb/s pasovne širine, kar predstavlja približno 55-65 kb/s na uporabnika.
- S kakšno strojno opremo razpolagajo uporabniki ?
Uporabniki imajo vsi vsaj Intel Pentium osebne računalnike. Operacijski sistem je Windows 98 oz. Windows NT. Vsi so povezani v lokalno omrežno domeno.

5.3.2 Načrtovanje strežnika

5.3.2.1 Strojna oprema

Naslednji korak je določitev kapacitet in položaja MetaFrame strežnika oz. strežnikov, ki jih bomo uporabljali. Pri načrtovanju nam pomagajo odgovori iz analize.

Za delovanje Microsoft Terminal Serverja rabimo 133 megaherčni procesor Pentium, 32 megabajtov ram pomnilnika in trdi disk z najmanj 128 megabajtov prostora. Citrix priporoča, da rezerviramo še dodatnih 85 megabajtov za namestitev MetaFrame dodatka. To so uradne zahteve, ki jih je seveda bolje ignorirati, če želimo zares uporabljati MetaFrame strežnik. To so zares minimalne zahteve, s katerimi spravimo strežnik v pogon in omogočimo enemu uporabniku uporabo aplikacij.

Specifikacijo strežnika lahko razdelimo na naslednje elemente (Anderson, 2002, str. 43):

- **Procesor.** Danes so na voljo že dvogigaherčni procesorji. Vendar je poleg frekvence procesorja pomemben tudi predpomnilnik, ki ga ta ima. V predpomnilniku se hranijo podatki, ki jih procesor večkrat uporablja, tako da ni potrebno dostopati do pomnilnika RAM računalnika, kar je časovno zahtevno. V večini primerov je procesor z manjšo taktno frekvenco in večjim predpomnilnikom lahko hitrejši od tistega z višjo taktno frekvenco in manjšim predpomnilnikom. Odločilno je tudi število procesorjev, predvsem v večuporabniškem sistemu kot je MetaFrame strežnik. Priporočljivo je uporabljati večprocesorski sistem oz. uporabiti matično ploščo, ki omogoča kasnejšo dodajanje procesorjev. Treba je poudariti, da hitrost procesorja ni ozko grlo pri MetaFrame strežnikih. Ozko grlo je RAM pomnilnik, kot je razvidno iz naslednje točke.
- **Pomnilnik.** Najbolj vpliven element zmogljivosti strežnika predstavlja RAM pomnilnik. MetaFrame strežnik uporablja štiri gigabajte pomnilniškega prostora. V tem prostoru hrani vse podatke, ki jih uporablja. Ta navidezni pomnilniški prostor je deloma hranjen v RAM pomnilniku, deloma pa v odstranjevalni datoteki (angl. paging file), ki se nahaja na trdem disku. Operacijski sistem poskuša hraniti najbolj uporabljene podatke v RAM pomnilniku, ker dostop do njega traja le nekaj nanosekund, medtem ko dostop do odstranjevalne datoteke, locirane na trdem disku, traja nekaj mikrosekund. Iz tega je razvidno, da v večuporabniškem okolju RAM pomnilnik igra pomembno vlogo, saj več hkratnih uporabnikov potrebuje več pomnilnika. Koliko RAM pomnilnika potrebujemo je seveda odvisno od tega, koliko uporabnikov bo uporabljalo MetaFrame strežnik in predvsem kakšne aplikacije bodo uporabljali. Citrix priporoča 16 megabajtov pomnilnika na uporabnika, dodati pa je treba še 4-8 megabajtov po aplikaciji, ki jih uporabljajo. V praksi se izkaže, da je dobro imeti čim več RAM pomnilnika, saj ta zmanjšuje dostop do trdega diska. Priporočljivo je začeti z vsaj 512 megabajti pomnilnika in nato po potrebi dodajati pomnilne enote.
- **Trdi disk.** Ključni element učinkovitega MetaFrame strežnika je hiter in zanesljiv trdi disk. Priporočljivo je uporabiti trde diske s SCSI vmesnikom (angl. Small Computer System Interface), saj omogočajo prenos podatkov s hitrostjo do 80 mb/s. Če želimo uporabljati RAID diske (angl. Redundant Array of Independent Disks), je priporočljivo uporabljati strojni RAID, saj je ta hitrejši od RAID-a, ki temelji na programski opreми. Namen RAID-a je zagotavljanje večje odpornosti diskov zoper strojne napake in s tem varovanje podatkov. To je sicer vprašljivo v primeru MetaFrame strežnika, saj naj praviloma ne bi hranili nobenih podatkov na njem. Vsekakor pa strojna RAID tehnologija v določenih

primerih poviša odzivnost trdih diskov. Za datotečni sistem je priporočljivo izbrati NTFS (angl. NT File System) namesto zastarelega FAT (angl. File Allocation Table) sistema. NTFS omogoča večjo varnost in hitrost dostopa do datotek.

- **Omrežna kartica.** Priporočljivo je uporabljati omrežno kartico z visoko hitrostjo. Čeprav sam ICA protokol ne zahteva veliko pasovne širine, moramo izhajati iz tega, da bo MetaFrame strežnik komuniciral tudi z baznim strežnikom. Pri uporabi aplikacij arhitekture debeli odjemalec / strežnik je priporočljivo imeti čim večjo pasovno širino povezave med podatkovnim strežnikom in MetaFrame strežnikom.
- **Video kartica.** Lahki odjemalci so namenjeni predvsem aplikacijam, ki niso grafično zelo zahtevne. Zato se lahko zadovoljimo s povprečno grafično kartico, ki jo dobimo na tržišču. ICA protokol priporoča uporabo 16 – 256 barv za čim manjšo pasovno širino. Seveda pa je omogočena tudi večja ločljivost.
- **Število strežnikov.** Razumljivo je, da lahko večjo zanesljivost sistema dosežemo z več majhnimi strežniki, povezanimi v strežniško farmo, kot z enim velikim strežnikom. V primeru izpada enega strežnika bodo uporabniki tega strežnika sicer izgubili trenutno sejo. Ponovno pa se bodo lahko priključili na drug strežnik.

Na podlagi zgornjih kriterijev smo se odločili za dva strežnika tipa Compaq Proliant 800 z naslednjo specifikacijo:

- Procesor: 2 x Pentium II 750 Mhz
- Pomnilnik: 512 megabajtov RAM pomnilnika
- Trdi disk: 30 gigabajtov
- Diskovni krmilnik: SimBios SCSI-3
- Omrežna kartica: Compaq Netelligent 10/100 TX PCI UTP Controller 2.3

Ključnega pomena je odločitev za dva strežnika, kajti v primeru izpada enega bodo uporabniki lahko nadaljevali delo na drugem.

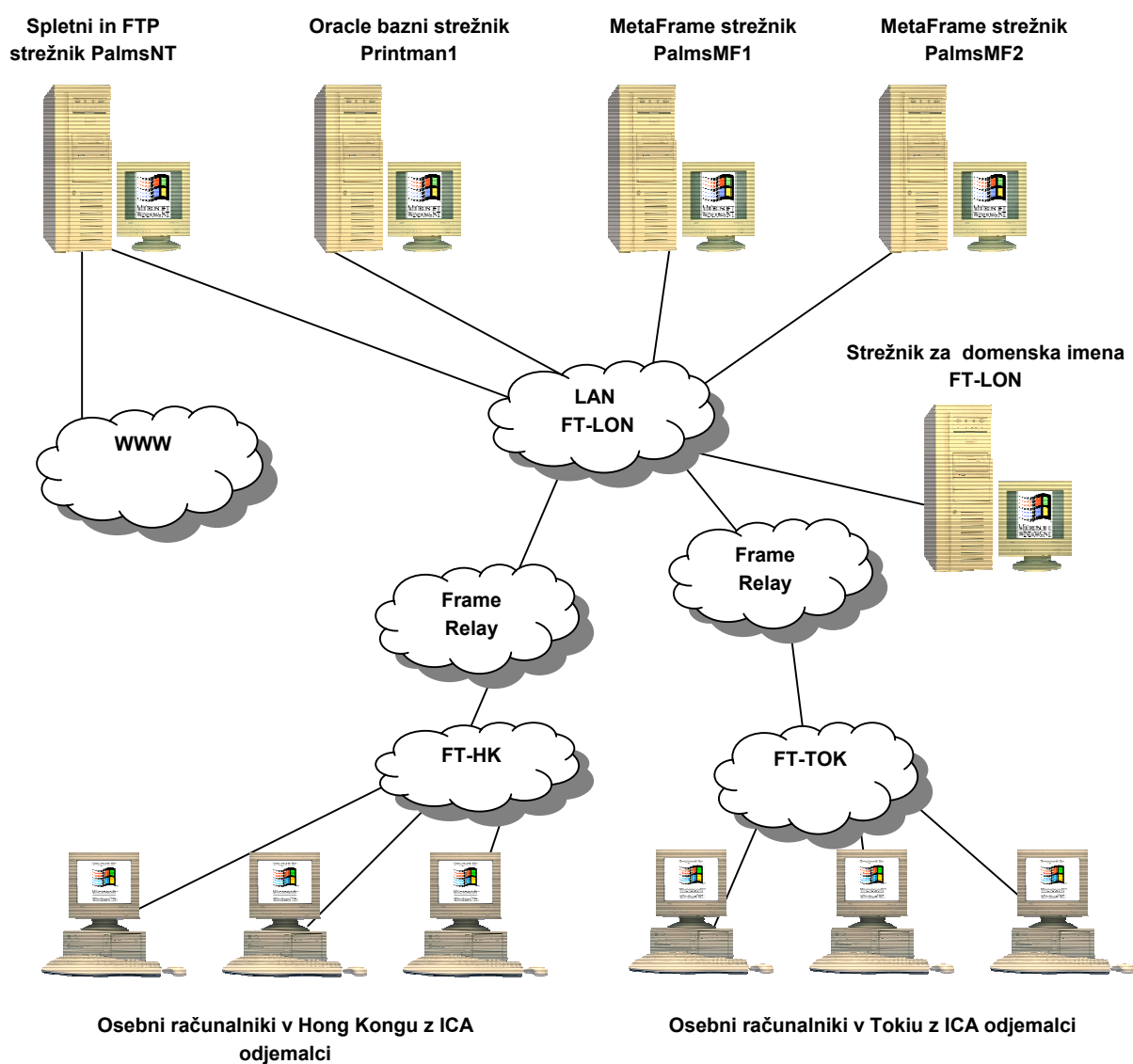
5.3.2.2 Domenska struktura

Pomemben del implementacije je načrtovanje pozicije MetaFrame strežnika v Windows NT domeni. MetaFrame strežnik ne zahteva prisotnosti Windows NT domene, vendar to ni priporočljivo, saj bi morali uporabniki imeti različne račune za dostop na vsakem posameznem MetaFrame strežniku. To bi zelo oteževalo administracijo večjih skupin uporabnikov. Vendar imajo dandanes vsa večja podjetja vsaj eno Windows NT domeno, praviloma celo več. Vsaka Windows NT

domena ima krmilnik domene, na katerega bi lahko namestili MetaFrame strežnik. MetaFrame strežnik lahko deluje kot strežnik za domenska imena, vendar je to skrajno nepriporočljivo. Opravilo domenskega strežnika zahteva veliko resursov, tako da bi po nepotrebnem otežali delo aplikacijskega strežnika, učinkovitost obeh bi pa seveda padla.

V okviru projekta PALMS – ASIA smo se odločili, da bomo MetaFrame strežnika postavili v Windows domeno FT-LON in ju poimenovali PalmsMF1 in PalmsMF2. FT-LON je glavna domena sedeža podjetja v Londonu. V Hong Kongu najdemo domeno FT-HK, v Tokiu pa FT-TOK, ki so povezane z domeno FT-LON prek Frame Relay povezave. Omrežno strukturo in mesto MetaFrame strežnikov PalmsMF1 in PalmsMF2 prikazuje slika 32.

Slika 32: Omrežna struktura centrale v Londonu in oddaljenih lokacij



Vir: Financial Times, 2001

V primeru izpada Frame Relay povezave med dislocirano poslovalnico in informacijskim centrom v Londonu je uporabnikom na voljo modemski dostop do domene FT-LON, prek katere se lahko povežejo z MetaFrame strežniki. Modemski dostop naj bi zagotovil povezavo vsaj 28 kb/s, praviloma pa je ta 40 kb/s, kot smo lahko ugotovili pri testiranju. Takšna pasovna širina že nudi zadovoljivo delovanje ICA odjemalca.

5.3.2.3 Namestitev programske opreme

Po izbiri strojne opreme lahko namestimo programsko opremo na strežnik. Najprej je potrebno namestiti Microsoft Terminal Server, nato pa še dodatek Citrix MetaFrame. Za to delo smo angažirali certificiranega Citrix konzultanta, ki je svoje delo opravil v enem dnevu.

5.3.2.4 Določitev strežniških farm

Strežniška farma predstavlja združitev več MetaFrame strežnikov z namenom lažjega upravljanja. Strežniki iz različnih domen so lahko del iste farme, vendar je en strežnik lahko prisoten samo v eni farmi. Strežniška farma je podlaga za modul izenačevanja obremenitve (angl. Load Balancing), ki bo opisan v nadaljevanju. Znotraj strežniške farme lahko objavljamo aplikacije, kar olajša uporabnikom oz. ICA odjemalcem dostop do teh. ICA odjemalec nima podatkov o strežniku, na katerem se nahaja aplikacija, ampak ima le podatke o ICA pregledovalniku (angl. ICA browser). Ta hrani informacije o vseh dosegljivih MetaFrame strežnikih, njihovih naslovih in aplikacijah.

MetaFrame strežnika PalmsMF1 in PalmsMF2 smo združili v strežniško farmo PalmsMF Farm. Aplikacija, ki smo jo objavili znotraj farme, pa je seveda PALMS.

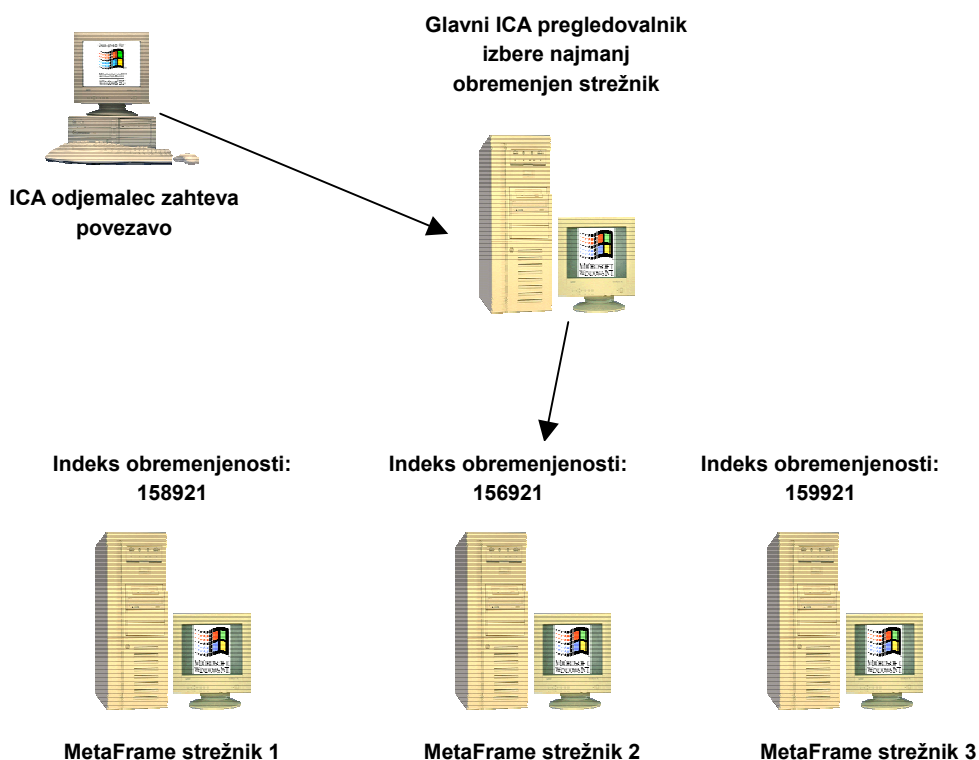
5.3.2.5 Nastavitev izenačevanje obremenitve

Izenačevanje obremenitve (angl. Load Balancing) služi delitvi obremenitve med strežniki v določeni farmi. Za to nalogo storitev izenačevanje obremenitve uporablja t.i. ICA pregledovalnik. Kot je že omenjeno v prejšnjem poglavju, hrani ICA pregledovalnik seznam vseh MetaFrame strežnikov in podatke o njihovi trenutni obremenitvi. Ko določen ICA odjemalec zahteva povezavo z objavljeno aplikacijo, ki uporablja storitev izenačevanje obremenitve, dejansko poišče ICA pregledovalnik. ICA pregledovalnik preusmeri odjemalca na najmanj obremenjen strežnik. Status obremenjenosti določenega MetaFrame strežnika prejema ICA pregledovalnik v obliki numeričnega indeksa. Čim manjši je indeks, tem manjša je obremenjenost strežnika. Indeks obremenjenosti se računa na podlagi nastavljenih parametrov, ki so bili podrobneje opisani v poglavju 4.1.4.1.

Po vzpostavitvi povezave med ICA odjemalcem in MetaFrame strežnikom odjemalec ostane povezan s tem strežnikom ne glede na njegovo obremenjenost. Če bi se ta strežnik zaradi nepredvidljivih razlogov zrušil, bi se uporabniki lahko ponovno povezali z enim od preostalih strežnikov v farmi.

Razvidno je, da igra sam ICA pregledovalnik pomembno vlogo pri razvrščanju povezav do MetaFrame strežnikov. Tako je lahko vsak MetaFrame strežnik hkrati tudi ICA pregledovalnik. Seveda je lahko samo en strežnik glavni pregledovalnik (angl. Master Browser). V primeru da glavni ICA pregledovalnik preneha delovati, se samodejno določi naslednji delujoč MetaFrame strežnik kot ICA pregledovalnik. Priporočljivo je, da naj bi bil MetaFrame strežnik, ki služi kot glavni ICA pregledovalnik, čim manj obremenjen, saj mora poleg izvajanja aplikacij skrbeti še za pravilno razvrščanje odjemalcev in pridobivanje informacij o razpoložljivih strežnikih in njihovih statusih obremenjenosti. V praksi je to seveda težko izvedljivo, kajti težko bomo upravičili MetaFrame strežnika samo za opravljanje funkcije ICA pregledovalnika. Slika 33 prikazuje proces izbire najmanj obremenjenega MetaFrame strežnika.

Slika 33: Proces izbire najmanj obremenjenega strežnika



Vir: Kaplan, 2000, str. 420

MetaFrame strežnik PalmsMF1 smo določili kot glavni ICA pregledovalnik, strežnik PalmsMF2 pa kot varnostni ICA pregledovalnik, ki se bo samodejno aktiviral v primeru da PalmsMF1 odpove. Za kriterij kalkulacije indeksa obremenjenosti smo

določilo število uporabnikov, ki so priključeni na določen strežnik. Tako se bodo uporabniki sorazmerno razporejali na oba strežnika, kajti v našem primeru vsak uporabnik zaseda približno enak del virov, saj uporabljajo isto aplikacijo.

Po končani namestitvi in konfiguraciji strežnikov smo nadaljevali z namestitvijo komponent, ki so bile potrebne za delovanje aplikacije PALMS, ki naj bi se izvajala na MetaFrame strežnikih.

5.3.3 Namestitev aplikacij

V nasprotju s klasičnimi odjemalci / strežniki arhitekturami se aplikacija pri Citrix MetaFrame arhitekturi lahkih odjemalcev izvaja na strežniku. ICA odjemalec ne izvaja nobenega procesiranja, razen prikazovanja grafičnega uporabniškega vmesnika. Ena največjih prednosti Terminal Serverja in MetaFrame strežnika je možnost izvajanja Windows aplikacij brez kakršnikoli sprememb. To seveda ne drži stoodstotno. Aplikacije, ki so bile načrtovane za večuporabniško okolje, ponavadi ne rabijo nobenih sprememb in delujejo brez napak. Težave pa lahko nastopijo, kadar aplikacije uporabljajo istoimenske datoteke za hranjenje začasnih podatkov, tako da lahko pride do njihove kolizije pri več hkratnih uporabnikih. Napaka je pogosto v absolutnem naslavljanju datotečne poti, tako da dva uporabnika poskušata uporabljati isto datoteko, kar seveda povzroči nepredvidljive in nezaželene rezultate.

Namestitev novih aplikacij poteka preko standardne Windows ikone dodaj/odstrani programe, ki se nahaja na krmilni plošči. Pozorni moramo biti le na to, da nameščamo nove programe vedno preko te funkcije in ne z neposrednim zagonom standardnih namestitvenih datotek. Windows aplikacije namreč svoje nastavitve ponavadi hranijo v Windows registru (angl. Windows registry). Ker ima vsak uporabnik svoje nastavitve v aplikacijah, bi lahko v večuporabniškem okolju prišlo do kolizije teh nastavitvev. MultiWin jedro MetaFrame strežnika ta problem rešuje z ustvarjanjem senčnih registrov za vsakega uporabnika. Tako se pri namestitvi novih aplikacij nove vrednosti v registru podvajajo v senčnih registrih vseh uporabnikov z namenom, da imajo vsi uporabniki privzete nastavitve nove aplikacije. Po namestitvi nove aplikacije lahko zaženemo še tako imenovane acs skripte (angl. application compatibility scripts), ki so na voljo za bolj razširjene programske pakete, kot npr. Word, Excel itd. Acs datoteke uredijo določeno aplikacijo tako, da brezhibno deluje v večuporabniškem okolju. Po namestitvi novih aplikacij lahko začnemo z njihovim testiranjem.

5.3.3.1 Seznam aplikacij

Preden začnemo s testiranjem aplikacij je priporočljivo, da ustvarimo seznam vseh aplikacij, ki smo jih namestili na MetaFrame strežnik in jih nameravamo uporabljati

v produkcijskem okolju (Kaplan, 2000, str. 407). Seznam naj bi nam pomagal pri ugotovitvi nestandardne uporabe programske opreme. Nesmiselno je v isti organizaciji uporabljati dva programska paketa, ki imata enako funkcionalnost, kot npr. Microsoft Office in Lotus SmartSuite. Bolje je, če se odločimo samo za eno in si s tem zmanjšamo kompleksnost sistema, hkrati pa ne zmedemo uporabnikov. Predvsem pa nam seznam aplikacij služi za preglednost sistema in lažjega dodajanja novih strežnikov v primeru širitve strežniške farne. Pri vseh aplikacijah je seveda potrebno dodati navodila za namestitev, da se namestitev lahko izvrši čim hitreje.

V primeru projekta PALMS – Asia smo na MetaFrame strežnik namestili le tiste aplikacije, ki so nujni za delovanje sistema. Naslednja tabela prikazuje seznam le-teh.

TABELA 5: Seznam nameščenih aplikacij na MetaFrame strežnikih PalmsMF

Ime aplikacije	Opis	Velikost	Datotečna Pot
Magic 8.3 Runtime	Izvajalno okolje za aplikacijo PALMS, ki je razvita v orodju Magic	21 MB	\\Magic8\\Palms\\
Oracle odjemalec 8.1.7	Odjemalec za dostop podatkovne baze Oracle	40 MB	\\Orant\\
Adobe Acrobat Reader 4.0	Program za pregledovanje PDF datotek, ki jih izdeluje PALMS	10 MB	\\Program Files\\Adobe\\
Internet Explorer 5.0	Pregledovalnik spletnih strani	15 MB	\\Program Files\\Internet Explorer\\

Vir: Financial Times, 2001

Seznam nameščenih aplikacij oz. komponent je razmeroma majhen, ampak namen PalmsMF MetaFrame strežnikov je, zaenkrat izključno izvajanje aplikacije PALMS.

5.3.3.2 Testiranje aplikacij

Vsaka aplikacija naj bi šla skozi dve fazi testiranja: komponento testiranje in sistemsko testiranje. Pri komponentnem testiranju preverjamo samo obnašanje aplikacije na večuporabniškem MetaFrame strežniku, pri sistemskem testiranju pa preverjamo obnašanje aplikacije na polno obremenjenem strežniku. Cilj testiranja je čim bolj podrobno in izčrpno preverjanje določene aplikacije, kar pa seveda pri omejenih časovnih resursih ni vedno možno.

Komponentno testiranje

Cilj komponentnega testiranja je opazovanje obnašanja aplikacije v večuporabniškem okolju MetaFrame strežnika. To je še posebej pomembno pri aplikacijah, ki niso bile načrtovane za takšno okolje. Ponavadi se težave pojavijo pri starejših programih, predvsem 16-bitnih DOS aplikacijah. Komponentno testiranje delimo po funkcionalnostih določene aplikacije (Kaplan, 2000, str. 407):

- **Generične funkcije.** Testiranje generičnih funkcij aplikacij se nanaša na preverjanje opravil, ki so prisotna v večini Windows aplikacij. Primeri generičnih funkcij so npr. zagon aplikacije, izhod iz aplikacije, odpiranje datoteke, izreži – prilepi itd. Preverjanje generičnih funkcionalnosti nam zagotavlja, da se aplikacija obnaša pravilno v večuporabniškem okolju MetaFrame strežnika. Priporočljivo je, da izdelamo sezname za testiranje aplikacij, ki jih nato lahko prilagajamo določenim aplikacijam, kot prikazuje tabela 6 na str.68.
- **Specifične funkcije.** Kot pove že samo ime, je ta del testiranja specifičen za vsako aplikacijo posebej. Pri vsaki aplikaciji naj bi preverili vsaj eno specifično funkcionalnost in sestavili seznam za testiranje. Primer specifične funkcije je lahko zagon novega makra v Microsoft Excelu. Tabela 7 na str. 69 prikazuje seznam za specifično testiranje za aplikacijo PALMS.

Sistemske testiranje

Sistemske testiranje zagotavlja pravilno obnašanje aplikacije na polno obremenjenem strežniku. V tej fazi se ponavadi testira tudi obnašanje modula izenačevanje obremenjenosti (angl. Load Balancing). Sistemske testiranje praviloma vključuje naslednje postopke (Kaplan, 2000, str. 408):

- Izvajanje komponentnega testiranja na produkcijsko konfiguriranem MetaFrame strežniku. Na takem strežniku so nameščene vse aplikacije, ki jih bomo uporabljali. Strežnik je vključen v končno domeno, izenačevanje obremenjenosti je vključeno in strežnik je člen strežniške farme. Cilj takega testiranja je ustvariti okolje, ki je podobno oz. je produkcijsko okolje.
- Testiranje integracijskih funkcij. Tukaj gre za preverjanje povezovalnih funkcij aplikacij kot npr. povezava z zunanjo podatkovno bazo.
- Testiranje obremenjenosti. Cilj je ustvariti čim več hkratnih sej na strežniku in seveda zagon čim več instanc aplikacij. To lahko dosežemo z avtomatiziranimi skriptami ali s komercialnimi paketi, ki omogočajo testiranje aplikacij. Najlažji in najbolj učinkoviti način pa je seveda vključiti čim več ljudi in skupaj z njimi hkrati uporabljati in komponentno testirati aplikacijo.

- Testiranje aplikacije z različnimi odjemalci. Komponentno testiranje naj bi izvedli tudi na vseh načrtovanih ICA odjemalcih in seveda na vseh omrežnih točkah, kjer jih želimo uporabljati.

5.3.3.3 Testni seznam

Izdelava testnih seznamov je seveda ključnega pomena za uspešno testiranje primernosti določene aplikacije za večuporabniško okolje MetaFrame strežnika. Vsak testni seznam predstavlja testni cikel. Testnih seznamov oz. testnih ciklov naj bi bilo čim več, tako da lahko preverimo čim več funkcionalnosti aplikacije. V primeru neuspešno opravljenega testnega cikla je potrebno napako odpraviti in vse testne cikle ponoviti. V nadaljevanju sta prikazana dva testna cikla, ki smo ju uporabili v okviru projekta PALMS – Asia.

Tabela 6: Generični testni seznam za program Adobe Acrobat Reader

Korak	Test	Opis	Pričakovani rezultat	Rezultat	OK	Opombe
1	Zagonska metoda #1	Dvoklikni ikono Adobe Acrobat	Adobe Acrobat se naloži	Adobe Acrobat naložen	✓	
2	Odpri zadnji dokument	Meni File -> 1	Odpre se zadnji dokument	Zadnji dokument je na ekranu	✓	
3	Zapri document	Meni File -> Close	Dokument se zapre	Dokumenta se zapre	✓	
4	Odpri raziskovalec	Start ->Run ->Explorer	Odpre se raziskovalec	Raziskovalec odprt	✓	
5	Odpri dokument z:\out\po103_fra_ft_0806_1.pdf	Dvoklikni datoteko po103_fra_ft_0806_1.pdf	Odpre se nalog za tisk v Adobe Acrobat Readerju	Dokument odprt	✓	
..						
25	Zapri Adobe Acrobat	Meni File / Exit	Adobe Acrobat se zapre	Adobe Acrobat izgine	✓	

Vir: Financial Times, 2001

Tabela 6 prikazuje izsek iz testnega seznama za komponentno generično testiranje Adobe Acrobat Readerja. Poleg klasičnega zagona programa se preveri tudi asociacija datotečne končnice PDF z Adobe Acrobat Readerjem. Program Adobe Acrobat Reader se v aplikaciji PALMS uporablja za pregled in morebiten tisk dokumentov, ki jih izdelava PALMS.

Tabela 7: Specifični testni seznam za funkcionalnost dodajanja naročnika v aplikaciji PALMS

Korak	Test	Opis	Pričakovani rezultat	Rezultat	OK	Opombe
1	Odpri seznam naročnikov	Meni Setup / Customers	Prikaže se okno z seznamom naročnikov	Seznam naročnikov na ekranu	✓	
2	Dodaj fiktivnega naročnika	Meni Edit / Create	Prikaže se nova prazna maska	Prazna maska na ekanu	✓	
3	Shrani naročnika	Klikni Gumb Shrani	Naročnik shranjen in v seznamu naročnikov	Seznam naročnikov na ekranu (z novim naročnikom)	✓	
4	Izbriši naročnika	Meni Edit / Delete	Potrditveno vprašanje	Potrditveno vprašanje	✓	
5	Izbriši naročnika (2)	Potrdi (klikni gumb Yes)	Naročnik izbrisan	Seznam naročnikov (brez izbrisanega naročnika)	✓	
6	Zapri seznam naročnikov	Klikni gumb Close (X)	Seznam naročnikov se zapre	Seznam naročnikov izgine	✓	

Vir: Financial Times, 2001

Testni cikel, ki je prikazan v tabeli 7, je namenjen preverjanju dodajanja in brisanja fiktivnega naročnika v seznamu naročnikov. Poleg specifičnega testiranja, s tem testom opravimo deloma tudi sistemsko testiranje integracije, saj manipulacija naročnika zahteva komunikacijo z relacijsko bazo podatkov Oracle, ki je na baznem strežniku. Hkrati tudi preverimo pravilno delovanje Oracle odjemalca, ki je nameščen na MetaFrame strežniku, saj se aplikacija PALMS preko Oracle odjemalca povezuje z bazo podatkov.

Nenačrtovano testiranje

Za vse postopke testiranja naj bi imeli sezname testiranja, ki nam povejo, katere funkcionalnosti je potrebno preveriti. Pri nenačrtovanem testiranju pa gre za testiranje iz glave in izvajanje funkcij, ki sploh niso na seznamu testiranja. S takšnim testiranjem lahko včasih ugotovimo kakšno nepravilnost, ki je drugače ne

bi odkrili. Nenačrtovano testiranje naj bi se izvajalo šele po uspešno opravljenem komponentnem in sistemskem testiranju (Kaplan, 2000, str. 410).

Testiranje aplikacije PALMS je potekalo na bodočih produkcijskih strežnikih brez večjih problemov. Posebej pozorni smo bili pri komunikaciji z Oracle strežnikom, saj je moral MetaFrame strežnik za vsakega uporabnika ustvariti svojo povezavo z Oracle strežnikom. Pri sistemskem testiranju odjemalcev na osebnih računalnikih smo naleteli na majhno kozmetično napako. Vsa okna v aplikaciji PALMS so namreč obarvana v tradicionalni roza barvi podjetja Financial Times, ki pa je v 256 barvnem načinu ICA odjemalcev postala moteča siva šrafura. Ločljivost odjemalcev bi lahko povečali, vendar bi to pomenilo nepotrebno povečanje pasovne širine, tako da smo ta pojav odpravili na strani aplikacije, saj je bila potrebna le majhna modifikacija barvnih nastavitev.

Obremenjenost strežnika smo testirali z desetimi hkratnimi uporabniki na enem strežniku, ki so izvajali komponentno testiranje. Obremenjenost strežnika je bila še vedno majhna, saj se večina procesiranja izvede na Oracle strežniku, ki je v produkcijski fazi projekta tudi postal ozko grlo sistema in ga je bilo treba nadgraditi.

5.3.4 Testiranje sistema z uporabniki

Po zaključenem testiranju sistema, ki so ga izvedli administratorji strežnika in razvijalci lahko preidemo na testiranje aplikacije, ki ga opravljajo končni uporabniki.

5.3.4.1 Namestitev odjemalcev

V primeru da bodo uporabniki uporabljali Windows Terminale, so ICA odjemalci praviloma že nameščeni. Če pa imamo opravka z osebnimi računalniki, moramo ICA odjemalce seveda najprej namestiti. Tukaj lahko izbiramo med naslednjimi možnostmi (Anderson, 2002, str. 153):

- namestitev iz omrežnega pogona,
- namestitev iz diskete ali CD roma,
- namestitev iz spletnega pregledovalnika.

Slika 34: Namestitev ICA odjemalca iz pregledovalnika na intranetu



Vir: Anderson, 2002, str.153

Pri vseh namestitvah je potrebna interakcija uporabnika, vendar ne zahteva strokovnega znanja. Najbolj smiselna izbira je seveda namestitev iz pregledovalnika, saj zahteva najmanj resursov. Namestitveno datoteko za ICA odjemalec lahko objavimo na intranetu ali pa končne uporabnike oz. administratorje napotimo kar neposredno na domačo spletno stran podjetja Citrix, kjer so nam na voljo najnovejši odjemalci. Po končani namestitvi ICA odjemalec sam raziše omrežno okolje in nam ponudi morebitne objavljene aplikacije.

V primeru projekta PALMS – Asia smo se odločili za namestitev ICA odjemalca neposredno iz domače spletne strani podjetja Citrix. To delo so opravili lokalni administratorji na dislociranih enotah v Tokiu in Hong Kongu. Brez težav so definirali tudi dostop do objavljene aplikacije PALMS.

5.3.4.2 Izbira uporabnikov

Najprej je potrebno izbrati določeno število uporabnikov, ki bodo testirali postavljeni sistem. Število uporabnikov ne sme biti preveliko, vsekakor pa morajo pokrivati vse module aplikacije, ki jih želimo testirati. Prav tako naj bi izbrani uporabniki bili naklonjeni projektu in tehnologiji lahkih odjemalcev. Če imamo opravka z oddaljenimi lokacijami, je seveda nujno, da sodelujejo tudi taki uporabniki. Zgoraj navedeno lahko povzamemo v naslednjih točkah (Kaplan, 2000, str. 489):

- izbrati je treba manjše število uporabnikov, ki pokrivajo vse delovne procese aplikacije, ki jo želimo testirati,
- sodelovati morajo tudi uporabniki iz oddaljenih lokacij,

- izbrani uporabniki morajo biti naklonjeni projektu in tehnologiji lahkih odjemalcev,
- uporabniki naj bi bili računalniško pismeni, kar pomeni da naj bi razumeli, s kakšno tehnologijo imajo opravka. Seveda ni potrebno, da imajo znanje programerjev ali administratorjev, vendar je priporočljivo, da se znajdejo v operacijskem sistemu Windows. To nam bo pomagalo pri odpravi morebitnih problemov in njihovi definiciji.

Za testiranje produkcijskega sistema pri projektu PALMS – Asia smo izbrali t.i. super uporabnike (angl. Superuser). Ti uporabniki so imeli dober vsebinski pogled v aplikacijo, saj so že prej aktivno sodelovali pri definiciji potreb za lokalizacijo aplikacije. Tudi motivacija uporabnikov je bila visoka, saj so bili naveličani tedanjega sistema za podporo distribucije, ki je bil zastarel in ni nudil vseh funkcionalnosti, ki jih je nudil nov sistem PALMS. Šolanje uporabe sistema PALMS je potekalo v Londonu in je bilo bolj vsebinskega značaja, vendar je bila uporabnikom razložena tudi tehnologija, s katero bodo uporabljali nov informacijski sistem.

5.3.4.3 Podpora uporabnikom

Pri implementaciji vsakega novega informacijskega sistema je ključnega pomena učinkovita podpora uporabnikom. Vsaka slaba izkušnja uporabnikov lahko vpliva na uspeh oz. neuspeh projekta, po drugi strani pa hitra in učinkovita pomoč uporabnikom zmanjša krivuljo učenja in odpor do novega sistema. V začetni fazi implementacije je pomembno, da vse napake beležimo in posredujemo razvijalcem oz. administratorjem sistema. Kasneje lahko sestavimo napotke za reševanje tipskih problemov, ki jih potem posredujemo oddelku za pomoč uporabnikom (angl. help desk).

Vsekakor pa je treba z novo tehnologijo lahkih odjemalcev seznaniti tudi oddelk za pomoč ter ga usposobiti za uporabo novih možnosti, kot je npr. senčenje uporabnikov. Z uporabo te funkcije odpade dolgotrajno definiranje problema z uporabnikove strani, saj operater takoj vidi, kje je uporabnik obtičal.

V fazi uporabniškega testiranja so pri projektu PALMS - Asia prejemale klice za podporo kar razvijalci sami, saj je bilo največ pripomb vsebinskih in niso bile povezane z večuporabniškim okoljem MetaFrame strežnika. Kot zelo učinkovito orodje se je izkazalo orodje senčenja, saj smo lahko določene postopke izvedli razvijalci, uporabniki pa so le sledili dogajanju na ekranu in se hkrati učili. Tudi odprava programskih napak v aplikaciji in vpeljava novih verzij je potekala hitro, saj je bilo treba namestiti novo verzijo le na MetaFrame strežnika.

Problemi, ki so se pojavili, so bile predvsem kratkotrajne prekinitve povezav med dislociranimi enotami in centralo v Londonu, kar pa ni bilo usodno. Uporabniki so se enostavno ponovno priključili na MetaFrame strežnik, ki jim je nato posredoval njihovo staro sejo, tako da niso izgubili nobenega opravljenega dela.

Pasovne širine povezave pa so vplivale na probleme pri tiskanju, saj je po omrežju moralo potovati kar nekaj megabajtov podatkov, ki so bili potrebni za grafični izpis naloga za tisk. Rešitev problema je bila dokaj enostavna, saj so vsi izpisi v obliki PDF datotek objavljeni na spletnem strežniku, ki je dosegljiv preko svetovnega spleta in je namenjen prenašanju dokumentov do tiskarn. Tako so uporabniki izpise najprej prenesli iz spletnega strežnika na svoj lokalni računalnik in jih nato iztiskali. Treba je poudariti, da se je ta problem pojavil le pri tiskanju dolgih izpisov in ne pri njihovem pregledovanju, saj se pri pregledovanju dokumenta prenaša le slika ekrana do uporabnika.

5.3.5 Vzporedna produkcija in produkcija

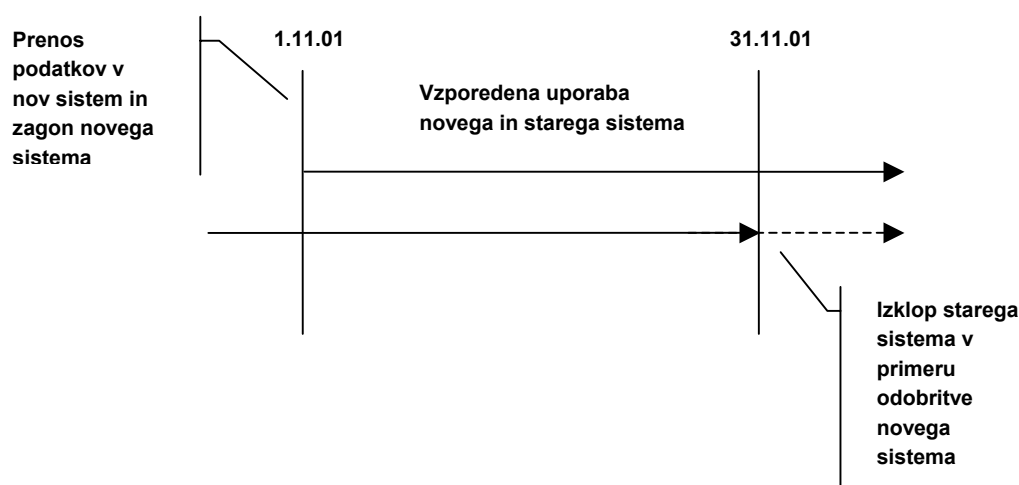
Po uspešnem testiranju in odobritvi sistema smo se odločili za produkcijsko uporabo aplikacije PALMS. Glede na to, da so uporabniki uporabljali star sistem za distribucijo časopisa, smo za prehod v produkcijo imeli na razpolago naslednje tri metode migracije (Bisbal et al., 1998, str. 13):

- **Prekinitev in start** (angl. Cut-and-Run). Migracija podatkov v nov sistem in ustavitev starega sistema. Nov sistem popolnoma nadomesti starega in uporabniki uporabljajo le še nov sistem. Tak pristop seveda predstavlja zelo velik rizik in se praviloma ne uporablja pri poslovno kritičnih sistemih, kajti v primeru, da se nov sistem izkaže kot nezaupljiv ostanemo brez zadovoljive informacijske podpore.
- **Fazni prehod** (angl. Phased Interoperability). Da bi zmanjšali rizik, naj bi se prehod na nov sistem izvajal v majhnih korakih. Zamenjamo le majhne komponente informacijskega sistema naenkrat. To seveda predpostavlja, da lahko izbran informacijski sistem razdelimo na več manjših samostojnih aplikacij.
- **Vzporedna produkcija** (angl. Parallel Operations). Nov in star informacijski sistem se hkrati uporabljata za določen čas. Tako se lahko podatki novega sistema primerjajo s podatki starega sistema, ki veljajo kot pravilni. Ko se nov sistem izkaže kot zaupljiv, lahko starega izklopimo. Vzporedna produkcija zmanjša tveganje, ki se pojavlja z vpeljavo novega informacijskega sistema, kajti v primeru odpovedi novega sistema lahko spet preidemo na star preizkušen sistem.

Odločili smo se za prehod na nov informacijski sistem z metodo vzporedne produkcije (angl. Parallel Operations). Tak način vpeljave novega informacijskega sistema nam omogoča prekinitev uporabe novega sistema v primeru nepredvidenih težav, ne da bi ostali brez informacijske podpore. To je seveda dodaten napor za uporabnike, vendar neizogibno pri sistemih, ki so kritični za uspešno poslovanje podjetja, kar sistem distribucije časopisa za podjetje Financial Times vsekakor je.

Pri vzporednem produkcijskem obdobju bodo uporabniki en mesec uporabljali obe aplikaciji in rezultate novega sistema primerjali z rezultati starega sistema. Časovna doba enega meseca je potrebna zaradi izdelave faktur za naročnike, ki se izdelajo na koncu meseca. To je bil končni test za nov sistem, saj so se podatki seveda morali usklajevati s tistimi iz starega sistema.

Slika 35: Vzporedno produkcijsko okolje



Vir: Financial Times, 2001

Pri vzporedni produkciji smo seveda imeli opravka s polnim številom uporabnikov, vendar so MetaFrame strežniki delovali zadovoljivo hitro, saj sta bila na voljo dva strežnika in sistem za izenačevanje obremenjenosti je uporabnike sorazmerno razvrščal na oba strežnika. Problemi so se začeli pojavljati na Oracle strežniku, ki je poleg Londonskih uporabnikov sedaj moral podpirati še 13 dodatnih uporabnikov. To se je dogajalo le občasno, saj je časovna razlika med Londonom in Hong Kongom osem, Tokiom pa devet ur. Problem smo odpravili z nadgradnjo pomnilnika Oracle strežnika.

Razen problemov, ki so se pojavljali v testnem obdobju, nismo srečali nobenih novih problemov, tako da smo ob koncu meseca naredili temeljito primerjavo med izpisi faktur in dobropisov novega in starega sistema. Pojavile so se seveda razlike, vendar smo pri njihovem raziskovanju odkrili napake pri starem sistemu, oz. so bile te razlike utemeljene. Po tem, ko so uporabniki nov sistem odobrili, smo

izklopili stari informacijski sistem in uspešno prešli v produkcijsko okolje ter uspešno zaključili projekt Palms – Asia.

5.4 Pridobljene izkušnje in poslovne koristi

Izkušnje

Prva občutna korist pri uporabi MetaFrame tehnologije je bila možnost hitrega prilagajanja spremembam. Pri klasičnem odjemalcu / strežniku okolju je potrebno vsako novo različico poslovne aplikacije previdno planirati in praviloma izvršiti preko noči. Z uporabo MetaFrame strežnikov je potrebno zamenjati aplikacijo le na enem strežniku in vsi uporabniki lahko takoj delajo z novo verzijo aplikacije. Tako smo bili sposobni popravke na aplikaciji izdajati v zelo kratkem časovnem obdobju.

Za zelo koristno se je izkazala tudi velika zanesljivost MetaFrame strežnikov. Kot je že omenjeno, ni prišlo do nobenih resnih izpadov sistema v času vzporedne produkcije. Število klicev na oddelku za pomoč uporabnikom se je zmanjševalo, pa tudi klice, ki so prišli, je bilo možno rešiti na samem oddelku. Zelo majhno število je bilo potrebno posredovati do razvijalcev. Tako lahko ugotovimo, da je uvedba lahkih odjemalcev zmanjšala neposredne stroške podpore in odpravila stroške oz. izgube, ki so nastale zaradi uporabnikov nezmožnih opravljanja dela.

Ko je prišlo do problemov, ki so bili ponavadi povezani z vsebino aplikacije, se je funkcija senčenja izkazala kot zelo učinkovito orodje pri podpori uporabnikov. S senčenjem je bilo moč ugotoviti, kje je uporabnik obtičal in po potrebi prevzeti njegovo sejo ter odpraviti nastali problem. Uporabnik sam pa je lahko sledil dogajanju na svojem ekranu in se hkrati učil. Tudi tukaj lahko ugotovimo, da na ta način prihranimo čas in s tem znižamo stroške, saj odpade uporabnikovo razlaganje problema in tolmačenje situacije.

Odziv uporabnikov je bil pozitiven. Pri tem je treba poudariti, da se večina po določenem času ni več zavedala tega, da se aplikacija izvaja na strežniku, saj je bila ikona PALMS na njihovem namizju in aplikacija se je obnašala kot vsaka druga. Všeč jim je bilo, da so lahko po odpovedi njihovega osebne računalnika nadaljevali z delom v aplikaciji PALMS natanko tam, kjer so pred tem ostali, medtem ko so v drugih aplikacijah izgubili vse delo, ki so ga opravili.

Poslovne koristi

S poslovnega vidika je bil projekt PALMS – Asia popolni uspeh. Glavni namen projekta je bil dosežen, saj so uporabniki v Hong Kongu in Tokiu uporabljali centralni informacijski sistem za distribucijo časopisa.

Projekt je poudaril dejstvo, da je treba MetaFrame aplikacijske strežnike vzdrževati kot včasih velike računalnike, saj pri njihovi odpovedi uporabniki ne morejo izpolniti svoje poslovne funkcije. Po drugi strani pa je treba varovati le eno točko v omrežju. Dejansko se je izkazalo, da je dosegljivost aplikacije v celoti večja kot pa pri klasični postavitvi debeli odjemalec / strežnik, saj uporabnikom vedno uspe zrušiti svoj osebni računalnik in s tem tudi aplikacije, ki so nameščene na njem.

Poslovne koristi, ki jih je podjetju Financial Times prinesla implementacija lahkih odjemalcev v okviru projekta PALMS – Asia, lahko povzamemo v naslednjem seznamu:

- pridobitev centralnega informacijskega sistema za distribucijo časopisa in s tem:
 - centralno bazo naročnikov,
 - centralni pregled nad prodajo,
 - centralni pregled nad finančnimi tokovi,
- centralizirano upravljanje informacijskega sistema PALMS,
- zmanjševanje stroškov zaradi večje dostopnosti in zanesljivosti sistema in s tem hkrati povečano produktivnost uporabnikov,
- prihranek denarja zaradi podpore heterogenim odjemalcem oz. možnost uporabe starejših osebnih računalnikov,
- enostavne možnosti kasnejše širitve uporabe sistema PALMS na druge oddaljene poslovalnice z dodajanjem dodatnih MetaFrame strežnikov,
- zmanjšanje možnosti izgube podatkov zaradi centralnih varnostnih kopij in manjše možnosti okužbe sistema z virusi.

Razvidno je, da uvedba lahkih odjemalcev zmanjšuje skupne stroške lastništva aplikacije in povečuje učinkovitost upravljanja in njeno podporo.

Težko je natančno ovrednotiti prihranek stroškov, vendar lahko okvirno kvantificiramo določene prihranke, ki so posledica večje dostopnosti in zanesljivosti sistema in s tem hkrati povečajo produktivnost uporabnikov.

Predpostavimo lahko, da povprečna plača operaterjev v Veliki Britaniji znaša 25.000 britanskih funtov letno (Clifford et al., 1998, str. 16) tako, da po dodanih podatkih določeno podjetje plačuje 50.000 britanskih funtov za zaposlenega. Tako lahko ovrednotimo eno delovno uro zaposlenega na 25 britanskih funtov (50.000 / 165 h X 12 mesecev).

Statistike povprečne dostopnosti informacijskih sistemov so odvisne od arhitekture določenega sistema in so naslednje (Clifford et al., 1998, str. 16):

- 90% pri klasični arhitekturi debeli odjemalec / lahki strežnik,
- 98% največja možna dostopnost pri arhitekturi debeli odjemalec / lahki strežnik,
- 99.9% pri dobro organiziranemu centralnemu računalniku.

Pri uporabi lahkih odjemalcev s tehnologijo Citrix MetaFrame je dostopnost do informacijskega sistema naslednja (Clifford et al., 1998, str. 16):

- 95% dostopnost sistema v povprečni organizaciji,
- 99% dostopnost sistema v organizaciji z zelo dobro urejeno službo za informatiko,
- 99.9% dostopnost sistema z uporabo strežniških farm.

Iz navedenega lahko ugotovimo, da je v primerjavi s klasičnim modelom debeli odjemalec / lahki strežnik z uporabo lahkih odjemalcev možno nedostopnost sistema (angl. system downtime) zmanjšati za 1 - 5%. Sedaj lahko izračunamo prihranek stroškov, ki temelji na produktivnem času uporabnikov. V povprečni organizaciji bi znašal prihranek v enem letu 2500 britanskih funtov na uporabnika, v organizaciji z zelo dobro urejeno službo za informatiko pa 500 - 995 britanskih funtov na uporabnika.

Z uporabo funkcije senčenja je podpora uporabnikom učinkovitejša, tako da lahko ocenimo, da se je produktivnost operaterjev za podporo uporabnikom dvignila vsaj za 20% (Clifford et al., 1998, str. 17), kar po zgornjih podatkih o plačah predstavlja 10.000 britanskih funtov za enega operaterja na leto.

Pri projektu Financial Times PALMS – Asia gre za organizacijo z zelo dobro urejeno službo za informatiko in za 13 uporabnikov lahkih odjemalcev ter uporabo strežniških farm. Prihranek je torej 13 X 995 britanskih funtov, kar znese 12.935 britanskih funtov na leto. Ta vsota je najmanj, kar lahko pričakujemo, saj nismo upoštevali časa, ki ga uporabnik porabi za to, da lahko nadaljuje z delom tam, kjer ga je moral prekiniti zaradi nedostopnosti sistema. Prav tako niso upoštevane izgube, ki nastanejo zaradi nedostopnosti sistema. Prištejemo lahko še dodatnih 10.000 funtov zaradi povečane produktivnosti operaterja, ki je zadolžen za podporo uporabnikom aplikacije PALMS.

Ugotovimo lahko, da vpeljava lahkih odjemalcev za uporabo aplikacije PALMS zmanjša skupne stroške lastništva aplikacije za vsaj 23.935 britanskih funtov na leto. Treba je poudariti, da ni upoštevan prihranek stroškov, ki nastane zaradi

centraliziranega upravljanja informacijskega sistema. Ta prihranek je odvisen od števila novih različic aplikacije in števila končnih uporabnikov nove različice aplikacije. Težko bi bilo kvantificirati nastali prihranek, kajti število podatkov, ki bi jih bilo treba upoštevati, je preveliko. Nabavna vrednost enega strežnika in potrebne programske opreme znaša približno 5.000 angleških funtov (Citrix, 2002), torej 10.000 angleških funtov za dva strežnika. Iz tega je razvidno, da se je investicija v MetaFrame strežnike povrnila že v prvem letu njihove uporabe.

Vsekakor pa ne smemo zanemariti nevarnosti, ki jih prinese uvedba lahkih odjemalcev. Kot prvo lahko omenimo dejstvo, da postanemo odvisni od enega centralnega strežnika oz. strežnikov. Če odpovejo aplikacijski strežniki, ne moremo uporabljati sistema. Prav tako postanemo odvisni od omrežja. Brez delujočega omrežja ne bomo mogli uporabljati sistema. Vendar je to tveganje, ki ga moramo sprejeti za pridobitev centralnega informacijskega sistema.

6 SKLEP

V okviru magistrskega dela sem proučil in opisal tehnologijo lahkih odjemalcev. Ob pregledu razvoja računalniških arhitektur sem prikazal razloge za nastanek arhitekture lahkih odjemalcev, ki očitno povzema svoje prednosti od velikih centralnih računalnikov, saj je koncept velikih računalnikov in lahkih odjemalcev zelo podoben. Pri obeh modelih se aplikacije izvajajo na strežniku, do odjemalcev oz. terminalov pa potuje le uporabniški vmesnik. Temeljna razlika je seveda prav v uporabniškem vmesniku, ki je na terminalih centralnih računalnikov bil strogo tekstoven, pri lahkih odjemalcih pa uporabljamo sodoben uporabniku prijazen grafičen vmesnik. Lahke odjemalce lahko definiramo kot naprave, ki nudijo le najnujnejšo funkcionalnost za interakcijo s strežnikom.

Osredotočil sem se na najbolj razširjeno in uveljavljeno tehnologijo lahkih odjemalcev podjetja Citrix, ki jo je licenciral tudi Microsoft in je danes temeljni del operacijskega sistema Windows 2000. Prikazal sem možne uporabe take tehnologije in opisal temeljne elemente, prenosni protokol ICA in MultiWin operacijsko jedro, ki omogočajo delitev uporabniškega vmesnika od strežniškega dela in tako uporabo lahkih odjemalcev za delo s standardnimi Windows aplikacijami. Opravljena je bila primerjava in podani kriteriji za odločitev med Microsoft TSE tehnologijo in Citrixovim dodatkom MetaFrame. Razlika se je pokazala predvsem v prenosnih protokolih Microsoft RDP in Citrix ICA, ki se uporabljajo za prenos podatkov med odjemalcem in strežnikom. Ugotovljeno je, da protokol ICA nudi boljše možnosti uporabe, saj zaradi svoje odprtosti in nezahtevnosti na strani odjemalcev ne postavlja tehnoloških omejitev za implementacijo. V nasprotju s tem protokol RDP omejuje odjemalce na naprave z operacijskim sistemom Windows.

Na primeru projekta Financial Times Palms - Asia sem prikazal proces vpeljave lahkih odjemalcev z uporabo Citrix MetaFrame tehnologije. Namen projekta je bil priključitev dislociranih enot v Hong Kongu in Tokiu na centralni informacijski sistem za distribucijo časopisa v Londonu. Razmeroma majhne pasovne širine omrežnih povezav dislociranih enot z informacijskim centrom v Londonu so bile ključne za odločitev vpeljave lahkih odjemalcev, saj je prav to ena izmed pglavitnih prednosti te tehnologije.

Po uspešni zaključitvi projekta Palms – Asia sem lahko analiziral pridobljene izkušnje in ugotovil naslednje pridobljene koristi ob uporabi lahkih odjemalcev:

- **Centralizirano upravljanje.** S centraliziranim upravljanjem aplikacij in podatkov lahko prihranimo čas in s tem stroške pri vzdrževanju programske opreme, saj je potrebno določene spremembe vpeljati le na enem mestu. Pri projektu PALMS – Asia smo nove različice aplikacije lahko izdajali brez dolgotrajnega načrtovanja, saj ni bilo potrebno obnoviti programske opreme na vsakem odjemalcu posebej. Poleg tega je boljša tudi zaščita, saj so občutljivi podatki in aplikacije na strežniku in ne na delovnih postajah uporabnikov. Prav tako je povečana zanesljivost in dostopnost sistema, saj so strežniki skrbno varovani in vzdrževani.
- **Podpora oddaljenim uporabnikom.** Ne glede na vrsto povezave (WAN, ISDN, klicna povezava, intranet ali ekstranet) lahko uporabnike povežemo z aplikacijami, ne da bi morali zato spreminjati konfiguracijo sistema. Oddaljeni uporabniki lahko uporabljajo aplikacijo s kakovostjo, ki se lahko primerja s tisto v lokalnem omrežju, saj je potrebna le skromna pasovna širina. To je bil ključni vzrok za vpeljavo lahkih odjemalcev pri projektu Palms – Asia.
- **Podpora različnim in obstoječim odjemalcem.** Zaradi močno zmanjšanih potreb po strojnih zmogljivostih odjemalca, starejših namiznih računalnikov ni treba zamenjati, da bi lahko uporabljali najbolj zmogljive sodobne ključne aplikacije. Tako določenim uporabnikom aplikacije PALMS ni bilo treba zamenjati osebnih računalnikov, saj so nekateri računalniki bili prešibki za uporabo novega sistema. Da bi podprli različne odjemalce, ni več treba vzdrževati in podpirati več aplikacij. Odjemalci obstajajo za osebne računalnike, Macintosh računalnike, delovne postaje Unix, dlančnike in druge naprave.
- **Enostavno širjenje sistema.** V primeru povečanja števila uporabnikov lahko enostavno dodamo nove aplikacijske strežnike k obstoječim. Tako povečamo kapacitete našega sistema in omogočamo hitro širitev informacijskega sistema.

- **Podpora poljubnim aplikacijam.** Načeloma lahko uporabimo vsako Windows aplikacijo v našem večuporabniškem okolju, vendar se izkaže, da je potrebno aplikacije skrbno testirati in včasih tudi modificirati. V našem primeru niso bile potrebne nobene spremembe, saj je sistem bil načrtovan za večuporabniško okolje.

Iz zgoraj navedenega je moč ugotoviti, da uporaba lahkih odjemalcev zniža stroške lastništva aplikacije, saj močno zmanjša čas, ki je potreben za upravljanje, vzdrževanje in podporo uporabnikov pri določeni aplikaciji. Ker je dostopnost informacijskega sistema povečana v primerjavi s klasičnim modelom debeli odjemalec / lahki strežnik, so lahko prihranki skupnih stroškov lastništva aplikacije vsaj 500 – 2500 britanskih funtov letno na uporabnika. V organizaciji z desetimi uporabniki lahkih odjemalcev bomo lahko investicijo v MetaFrame strežnike povrnili v manj kot enem letu.

Prednosti lahkih odjemalcev je veliko, vendar tudi ti niso brez slabosti. Njihova šibka točka je hkrati tudi njihova prednost, saj pri centralizaciji informacijskih sistemov postanemo odvisni od enega centra in povezav, ki jih imamo z njim. Pri natančnem pogledu v zgodovino pa lahko ugotovimo, da so sistemi velikih računalnikov z enakimi tveganji uspešno delovali več desetletij.

Lahki odjemalci so uporabni v mnogih situacijah – pri pisarniškem poslovanju, za podporo mobilnim uporabnikom, v učilnicah in proizvodnih dvoranah s težkimi delovnimi razmerami. Prenosne terminalske sisteme lahko uporabimo zunaj zaprtih prostorov, razpoložljivi pa so tudi že vodotesni lahki odjemalci z zaslonom iz tekočih kristalov in vodotesno tipkovnico. Na strežnik se lahko povežejo preko antene in radio povezave. Tudi iz dlančnika si lahko naredimo lahkega odjemalca. Tehnologija je primerna tudi za izdelavo spletnih strani in uporabo po ekstranetu. O zrelosti tehnologije in njenem priznanju s strani podjetij po vsem svetu priča 80% podjetij iz seznama Fortune 500, ki uspešno uporabljajo Citrix tehnologijo lahkih odjemalcev.

Menim, da uporaba lahkih odjemalcev omogoča vračanje vloženih sredstev z zniževanjem skupnih stroškov lastništva aplikacij. Znižajo se začetni in kasnejši stroški aplikacij, kar omogoči hitrejšo in trajno vračanje naložbe.

Vsekakor koncept velikih centralnih računalnikov ni tehnologija preteklosti, ampak vse bolj tehnologija prihodnosti. S pridihom ironije bi lahko rekli, da se zgodovina ponavlja tudi v informacijski tehnologiji.

7 LITERATURA

1. Cole Bernard Conrad: The Emergence of Net-Centric Computing, Prentice Hall, 1999. 320 str.
2. Citrix: Citrix MetaFrame 1.8 Backgrounder, White Paper, Citrix Systems, June 1998. 53 str.
3. Clifford-Winters A., Hailstone R.: Windows by other means, Bloor Research, 1998. 24 str.
4. Craft Melissa, Keele Allen V., et. al: Configuring Citrix Metaframe for Windows 2000 Terminal Services, Syngress Media Inc., 2000. 608 strani
5. Cumberland Brian Craig, Carius Gavin, Muir Andrew: Microsoft Windows NT Server 4.0 Terminal Server: Technical Reference, Microsoft Press, Redmond, 1999. 392 str.
6. Cumhur Ercument Aksoy: Wireless Thin Client Optimization for Multimedia Applications, University of Florida, 2000. 166 str.
7. Černigoj Rihard: Legende se vračajo, Sistem, Ljubljana, November 2000, 2000, 11, str. 22-24
8. Dewire D.T.: Thin Clients, New York, McGraw Hill, 1998. 416 str.
9. Harwood Ted: Windows NT Terminal Server and Citrix MetaFrame, New Riders Publishing, 1998. 406 str.
10. Halfhill, T.R.: Cheaper Computing Part 1, BYTE, April 1997, Vol. 22, str. 66-80
11. Kanter Joel P.: Understanding Thin-Client/Server Computing, Microsoft Press, Redmond, 1998. 256 str.
12. Kapoor Atul, Total Cost of Application Ownership, The Tolly Group, 1999, 12 str.
13. Kaplan Steve, Mangus Marc, Templeton Mark: Citrix Metaframe for Windows Terminal Services: The Official Guide, Osborne McGraw-Hill, 2000. 608 strani
14. Keele Alen V.: Configuring Citrix Metaframe for Windows 2000 Terminal Services, Syngress Media Inc, 2000. 543 str.
15. Mann, K.: Thin Clients – Time for a down, Network News, Marc 1999, 51-52
16. Mathers Todd W. : Windows 2000 Thin Client Solutions, Macmillan Technical Publishing USA, Indianapolis, 2000. 650 str.
17. Maitland J.: Citrix Phones In, PC Week, London, Januar 1997, 19, str. 6
18. Mathers Todd W., Shawn P., Genoway: Windows NT Thin Client Solutions: Implementing Terminal Server and Citrix MetaFrame, Macmillan Technical Publishing, Indianapolis, 1998. 489 str.

19. Microsoft Corporation: Windows 2000 Terminal Server Edition: An Architectural Overview, White Paper, Microsoft Corporation, 2000a. 21 str.
20. Microsoft Corporation: Windows 2000 Terminal Server Edition: Reviewer's Guide, White Paper, Microsoft Corporation, 2000b. 56 str.
21. Microsoft Corporation: Windows 2000 Terminal Services Capacity Planning, White Paper, Microsoft Corporation, 2000c. 7 str.
22. Noris M., Winton N.: Energize the Network – Distributed Computing Explained, Addison Wesley, 1997. 483 str.
23. Parker Timothy: Teach Yourself TCP/IP in 14 Days, Second Edition, Indianapolis, Sams.net, 1996. 512 str.
24. Regel Julian: Does the Thin Client Paradigm provide a technical advantage over the traditional PC system for network oriented computing in business?, [URL:<http://www.white-tower.demon.co.uk/dissertation/>], April 2001, 21 str.
25. Rodstein Roddy, Duffield Phil: Citrix CCA MetaFrame 1.8 for Windows Exam Cram, First edition, The Coriolis Group, 2001. 448 str.
26. Simanski Robert E.: Reducing TCO Little Black Book: A Guide to Maximizing Benefits of the TCO Feature, The Coriolis Group, 2000. 512 str.
27. Sinclair Joseph T., Merkow Mark, Merkow Mark S.: Thin Clients Clearly Explained, 1st Edition, San Mateo, Morgan Kaufman publishers Inc., 1999. 350 str.
28. Tanenbaum Andrew S: Computer Networks. 3rd Edition, Upper Saddle River (N.J.): Prentice Hall, 1996, 813 str.
29. Thin-Client Networking: Bandwidth Consumption Using Citrix ICA, IT clarity, Tolly Research, Dec. 2000.
30. Trefil Laura: WinFrame 1.6 and 1.7 System Administration, Computing McGraw-Hill, 1998. 350 str.
31. Wayner P.: Inside the NC, BYTE November, Vol 21, 1996. str. 105-110
32. Wyse Technology: The Thin guide to Windows 2000, White Paper, Wyse Technology, 2000. 24 str.
33. Wyse Technology: Thin Clients, Windows Based Terminals, NC. What's the difference?, White Paper, Wyse Technology, Oktober 1997. 12 str.

8 VIRI

1. Anderson Christa, The Definitive Guide to MetaFrame XP, realtimepublishers.com, maj 2002
2. Bisbal Jesus, Lawless Deirdre, Bing Wu, Jane Grimson: Legacy Information System Migration: A Brief Review of Problems, Solutions and Research Issues, [URL:<http://www.cs.uu.nl/docs/vakken/swa/Reader/07-TCD-CS-1999-38.pdf>], September 2002, str. 13
3. Citrix, [URL: <http://www.citrix.com>], maj 2002
4. Desktop revolution, Netier Technologies Inc., 1999. 8 str.
5. Gartner Group, [URL: <http://www.gartner.com>], maj 2002
6. Interna gradiva, Financial Times Ltd., London, 2001
7. Internetworking Technology Overview, OSI Reference Model: Introduction. San Jose California: Cisco Systems Inc., [URL:http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc], maj 2002
8. Microsoft's Introduction to Terminal Services, [URL: <http://www.microsoft.com/windows2000/guide/server/solutions/terminal.asp>], April 2001
9. S. J. Yang: J. Nieh, Thin Is In, PC Magazine, 19, July 2000. str. 68
10. Pahor David, Drobnič Matija, et al.: Leksikon računalništva in informatike, Pasadena, Ljubljana, 2002
11. Reeser Tim, RDP or ICA Which should you choose?, Windows & .Net Magazine, May 1999, str. 2
12. The Evolution From Host Computing, Thin Planet, [URL:<http://www.thinplanet.com/tech/generic.asp?f=TDnumber&k=s&v=TD19573>], December 2001
13. Thin Client Architecture, Thin Planet, [URL:<http://www.thinplanet.com/tech/generic.asp?f=TDnumber&k=s&v=TD104331>], December 2001
14. Thin Client/Server Computing White paper, Citrix Systems, 1999. 20 str.
15. Wyse, [URL:<http://www.wyse.com>], Februar 2001
16. Zona Research, [URL:<http://www.zonaresearch.com/>], Februar 2001
17. Xiaochun Hu, ICS 690 Report, [URL:<http://www2.ics.hawaii.edu/~xiaochun/ics690/690report.htm>], April 2002, str. 9

Slovarček slovenskih prevodov tujih izrazov

okrajšava	angleški izraz	slovenski izraz
ALE	Application Launching and Embedding	izvajanje in vdelava aplikacij
API	Application Programming Interface	vmesnik uporabniškega programa
	Application Publishing	objavljanje aplikacij
ATM	Asynchronous Transfer Mode	način za asinhroni prenos
	Browser	Pregledovalnik
	Business Layer	poslovni nivo
CAD	Computer Aided Design	računalniško podprto načrtovanje
	Cut-and-Run	prekinitev in start
	Data Layer	podatkovni nivo
DOS	Disk Operating System	diskovni operacijski sistem
	Dumb Terminal	"neumni" terminal
FAT	File Allocation Table	tabela za dodeljevanje datotek
	Frame Relay	preklapljanje okvirov
GPRS	General Packet Radio Service	storitev za brezžični paketni prenos podatkov
GSM	Global System for Mobile communications	splošni sistem za mobilno komunikacijo
GDI	Graphical Device Interface	vmesnik grafične naprave
ICA	Independent Computing Architecture	neodvisna računalniška arhitektura
ISDN	Integrated Services Digital Network	digitalno omrežje z integriranimi storitvami
JVM	Java Virtual Machine	navidezna javanska mašina
LRU	Last Recently Used	nazadnje uporabljen
	Load Balancing	izenačevanje obremenitve
LAN	Local Area Network	krajevno omrežje
	Mainframe computer	osrednji računalnik
Net PC	Net Personal Computer	mrežni osebni računalnik
NC	Network computers	omrežni računalnik
NTFS	NT File System	datotečni sistem za NT
OM	Object Manager	upravitelj objektov
OSI	Open System Interconnection	povezovanje odprtih sistemov
	Parallel Operations	vzporedne operacije
	Phased Interoperability	fazni prehod
POS	Point of Sale system	blagajniški sistem
PDF	Portable Document Format	prenosna oblika dokumenta
	Presentation Layer	prikazovalni nivo
PALMS	Print and Logistic Management System	sistem za upravljanje tiskanja in distribucije
RAID	Redundant Array of Independent Disks	čezmerno polje samostojnih diskov

okrajšava	angleški izraz	slovenski izraz
RDP	Remote Desktop Protocol Session Shadowing	protokol za namizje na daljavo seja senčenje
SCSI	Small Computer System Interface	vmesnik za majhni računalniški sistem
TSE	Terminal Services Edition	izvedba s terminalskimi storitvami
TCO	Total Cost of Ownership	skupni stroški lastništva
TCP/IP	Transport Control Protocol and Internet Protocol	protokol za krmiljenje prenosa in internet protokol
VMM	Virtual Memory Manager	upravitelj navideznega pomnilnika
WAN	Wide Area Network	prostrano omrežje
WBT	Windows Based Terminals	terminal temelječi na Windowsih