

VIRTUALISOINTI

LAHDEN AMMATTIKORKEAKOULU

Tietotekniikan koulutusohjelma

Tietoliikennetekniikan suuntautumisvaihtoehto

Opinnäytetyö

Kevät 2008

Mikko Salmi

Lahden ammattikorkeakoulu
Tietotekniikan koulutusohjelma

SALMI, MIKKO: Virtualisointi

Opinnäytetyö, 54 sivua

Kevät 2008

Työn tavoitteena on kartoittaa ratkaisumahdollisuuksia ohjelmistojen yhteensopivuusongelmien ratkaisemiseksi. Työssä toteutetaan virtualisointia ohjelmistovirtualisointina ja laitteistovirtualisointina.

Tietotekniikassa virtualisoinnin voidaan ajatella jakautuvan kolmeen osa-alueeseen: laitteistovirtualisointiin, ohjelmistovirtualisointiin ja palveluvirtualisointiin. Näistä jokainen osa-alue on toimintaperiaatteeltaan hyvin erilainen toisiinsa nähden.

Laitteistovirtualisoinnilla tarkoitetaan tekniikoita, joilla yksi tai useampi fyysinen laite (isäntäkone) jaetaan pienempiin itsenäisiin osiin virtuaalisiksi laitteiksi. Nämä virtuaaliset laitteet sisältävät itsenäisesti toimivan käyttöjärjestelmän ja tämän palvelut sekä toimivat muutenkin kuten itsenäiset palvelinlaitteet. Yhden virtuaalisen palvelimen vikaantuminen ei vaikuta muissa virtuaalisissa palvelimissa toimiviin järjestelmiin. Laitteistovirtualisoinnilla saavutetaan hyötyjä vikasietoisuuden, joustavuuden ja laitteiden tehokkaamman hyödyntämisen osalta.

Ohjelmistovirtualisoinnilla tarkoitetaan sovelluksen erottamista käyttöjärjestelmästä, jolloin sovellus toimii omassa eristysympäristössään, joissa se voi käyttää tarvitsemaansa versiota käyttöjärjestelmän ohjelmakirjastosta, vaikka itse isäntäkäyttöjärjestelmässä olisi käytössä toinen versio. Ohjelmistovirtualisoinnilla voidaan saavuttaa siis tilanne, jossa sovellusten keskinäisiä yhteensopivuusongelmia on huomattavasti vähemmän kuin ilman virtualisointia. Ohjelmistovirtualisoinnin etuja voidaan vaihtoehtoisesti saavuttaa myös erinäisillä etäkäyttöratkaisuilla tai jopa yhdistää etäkäyttöratkaisut ja ohjelmistovirtualisointi toisiaan tukevaksi kokonaisuudeksi.

Palveluvirtualisointi luo palvelun käyttäjälle palvelun, joka toimii jaetulla laite- ja ohjelmistoalustalla. Vaikka palvelun asiakkaat toimivat täysin samassa ympäristössä, on ympäristöstä virtualisoitu tai eristetty eri asiakkaiden tiedot toisistaan.

Virtualisoinnilla saavutetaan hyötyjä sekä teknisen tuen tarpeen että hallintakustannusten vähentyessä. Myös uusien palveluiden tuottaminen tai sovellusten jakaminen loppukäyttäjille nopeutuu virtualisointia käytettäessä.

Avainsanat: Ohjelmistovirtualisointi, Virtualisointi

Lahti University of Applied Sciences
Faculty of Technology

SALMI, MIKKO: Virtualization

Bachelor's thesis in Information Technology, 54 pages

Spring 2008

The purpose of this work is to find possible solutions for software incompatibility issues through virtualization. Virtualization is carried out on both software and hardware level.

Virtualization is divided into three categories: hardware virtualization, software virtualization and service virtualization. Each of these categories works differently compared to each other.

With hardware virtualization physical host devices (servers) can be divided into multiple virtual devices. These virtual devices have their own operating systems and system services and work like independent physical devices. This means that a fault in one of the virtual machines crashes only that machine and does not affect other virtual machines running on the same host. With hardware virtualization it is possible to accomplish fault tolerance, flexibility and more efficient use of hardware resources.

Software virtualization technologies separate software from the operating system and make it run in a virtual isolation environment. In these environments the software can have the specific required version of the software library even if a later version of that library is installed on the host operating system. Having the required versions of libraries and software components installed reduces incompatibility issues between software. The same benefits can be accomplished by using remote software services. Software virtualization and remote software or desktop services can be used together to gain benefits from both technologies.

With service virtualization single hardware and software can be used to deliver a service to multiple customers. Separation between customers is done on the service level and therefore customers cannot access each other's data even the service operates on the same hardware.

In general, virtualization can be used to decrease the need for technical support and management and therefore generate savings in these sectors. Benefits of virtualization can also be seen when a new server or software is needed, as pre-installed and configured server templates and software can rapidly be deployed for the end user.

Keywords: Software Virtualization, Virtualization

SISÄLLYS

1 JOHDANTO	1
2 VIRTUALISOINTI YLEISESTI	2
2.1 Virtualisointi konseptina	2
2.2 Laitteistovirtualisointi	2
2.3 Ohjelmistovirtualisointi	5
2.4 Palveluvirtualisointi	7
3 NYKYINEN TARJONTA	8
3.1 Etäkäyttöratkaisut	8
3.1.1 Telnet/SSH/X Window System	8
3.1.2 Citrix Presentation Server	9
3.1.3 Microsoft Terminal Server	10
3.1.4 Ratkaisujen kustannukset ja vertailu tilanteeseen ilman etäkäyttöä	12
3.2 Laitteistovirtualisointi	14
3.2.1 VMWare	14
3.2.2 Microsoft Virtual Server	16
3.2.3 Virtuozzo	18
3.2.4 Ratkaisujen kustannukset ja vertailu tilanteeseen ilman virtualisointia	20
3.3 Ohjelmistovirtualisointi	21
3.3.1 Altiris – Software Virtualization Solution	21
3.3.2 Microsoft SoftGrid	24
3.3.3 Citrix Application Streaming	32
3.3.4 Ratkaisujen kustannukset ja vertailu tilanteeseen ilman virtualisointia	35
4 NYKYISET VIRTUALISOINTIRATKAISUT	37
4.1 Liiketoimintaa edistävät ratkaisut	37
4.2 Yleishyödylliset ratkaisut	37
4.3 Viihdekäyttö	38
5 TAPAUS DNA OY	39
5.1 Yleinen kuvaus toteutuksesta	39

5.2 Ratkaisujen kustannukset ja vertailu tilanteeseen ilman virtualisointia	45
5.3 Käyttäjäpalaute	47
5.4 Ratkaisulla saavutetut hyödyt	47
5.5 Mahdolliset tuotekonseptit	49
6 TULOKSET	51
7 ANALYYSI JA JOHTOPÄÄTÖKSET	52
8 TULEVAISUUS	53

LYHENNELUETTELO

AD	Active Directory (Microsoft Windows -ympäristöjen pääasiallinen tapa toteuttaa keskitetty tunnistus- ja toimilupapalvelu Windows-käyttöjärjestelmillä varustetuille tietokoneille. AD on hierarkkinen järjestelmä, joka koostuu erityyppisistä olioista. Työn kannalta AD:n tärkeimmät oliot ovat käyttäjät ja ryhmät. Käyttäjäryhmillä voidaan keskitetysti hallita käyttäjien pääsyoikeuksia verkon palveluihin ja resursseihin.)
ADSI	Active Directory Service Interface (Verkkoresurssien hallintaan tarkoitettu rajapinta)
AES	Advanced Encryption Standard (Vahva tiedonsalausmenetelmä)
ASP	Application Service Provider/Providing (Sovelluspalvelun tarjoaja)
BIOS	Basic Input-Output System (Tietokoneen peruslaitteistoa ohjaava järjestelmä)
EPA	U.S. Environmental Protection Agency Yhdysvaltojen ympäristönsuojeluviranomainen
ESD	Electronic Software Distribution (Sähköinen ohjelmistojakelu)
FB	Feature Block (SoftGrid sekvenssin osa)
FTA	File Type Association (Tiedostopäätteen ja sen avaaman ohjelman yhdistäminen toisiinsa)
HTTP	Hypertext Transfer Protocol (World Wide Webin käyttämä protokolla)
HTTPS	Hypertext Transfer Protocol Secure (World Wide Webin käyttämä salausta käyttävä protokolla)

ICA	<p>Independent Computing Architecture</p> <p>(ICA on arkkitehtuuri, jolla sovelluslogiikka irroitetaan käyttöliittymästä. ICA:a käyttämällä vain näppäimistön ja hiiren painallukset ja käyttöliittymän ruudunpäivitykset lähetetään palvelimen ja asiakkaan välillä. Tähän tiedonsiirtoon käytetään ICA-protokollaa. ICA-protokollan kuljettama informaatio voidaan salata ICA Encryption -ominaisuuden avulla, jolloin salauksen lisäksi saadaan käyttöön palvelimen oikeellisuustarkistus ja tiedon eheystarkistus. Protokolla sisältää virtuaalisia kanavia, joiden avulla esim. tulostukselle voidaan määritellä suurin mahdollinen käytettävä kaistanleveys ja pitää yhteys käytettävänä hitaallakin verkkoyhteydellä.)</p>
I/O	<p>Input/Output</p> <p>(Järjestelmän luku- ja kirjoitustoiminnot)</p>
MIT	<p>Massachusetts Institute of Technology</p> <p>(Massachusettsin teknillinen korkeakoulu)</p>
MMC	<p>Microsoft Management Console</p> <p>(Käyttöjärjestelmän ja palveluiden yleinen hallintatyökalu)</p>
NFS	<p>Network File System</p> <p>(Verkon välityksellä käytettävä tiedostojärjestelmä)</p>
NLB	<p>Network Load Balancing</p> <p>(Microsoftin kuormantasaustekniikka)</p>
OSD	<p>Open Software Descriptor</p> <p>(Sovellusta kuvaava tiedostotyyppi)</p>
RFC	<p>Request For Comments</p> <p>(Internetin toimintaa kuvaava muistio)</p>
RDP	<p>Remote Desktop Protocol</p> <p>(Monikanavainen protokolla, jota käytetään Windows Terminal Serveriin kytkeytymiseen. Protokollaa käytäviä sovelluksia on kehitetty useiden Windows versi-</p>

oiden lisäksi myös muille käyttöjärjestelmille. Kuvainformaation lisäksi alikanavissa voidaan siirtää yhteyden lisäominaisuuksien vaatimaa tietoa. RDP protokollan luomat yhteydet ovat salattu 128-bittistä RC4 salausalgoritmia hyödyntäen.)

RTCP

Real-Time Transport Control Protocol

(RTP kanssa toimiva tietoliikenneprotokolla, joka on määritetty RFC (Request For Comments) 3550:ssa. RTP:n tarkoituksena on kuljettaa itse tieto, ja RTCP hoitaa yhteyden hallinnan. Yhteyden hallinnan lisäksi RTCP kerää tietoa yhteyden laadusta, jonka avulla sovellus voi muuttaa toimintaansa muuttuneen yhteyslaadun mukaisesti. RTCP ei itsessään sisällä salaukseen tai autentikointiin tarvittavaa tietoa.)

RTP

Real-Time Transport Protocol

(RFC 3550:ssa määritelty tosiaikainen tietoliikenneprotokolla, joka on tarkoitettu äänen ja liikkuvan kuvan siirtämiseksi internetin välityksellä. Yleensä RTP:tä käyttävät sovellukset eivät välitä välistä kadonneista paketeista, vaan ne ovat herkkiä tiedonsiirtoviiveiden suhteen. Tämän vuoksi RTP kommunikoi usein UDP (User Datagram Protocol) -protokollalla, vaikkei standardi tätä määrittelekään.)

RTSP

Real Time Streaming Protocol

(Protokolla, jota käytetään reaaliaikaisen tiedon siirrossa. Protokollan avulla asiakasohjelma voi hallita palvelimen pään tiedonsiirtoa pyytämällä jotain tarvitsemaansa reaaliaikaisen median osaa. Protokolla on kuvattu RFC 2326:ssa ja se on kehitetty äänen ja videon siirtämistä varten.)

RTSPS

Real Time Streaming Protocol Secure

(RTSP salattuna)

SAN

Storage Area Network

	(Verkkotallennusjärjestelmä)
SGVAS	SoftGrid Virtual Application Server (SoftGrid virtuaaliohjelmistopalvelin)
SMP	Symmetric multiprocessing (Arkkitehtuuri, jossa yksi tai useampi samanlainen prosessori on yhteydessä yhteen muistiin)
SMS	Systems Management Server (Microsoftin sähköinen ohjelmistojakelu järjestelmä)
SVS	Software Virtualization Solution (Altiris:n ohjelmistovirtualisointijärjestelmä)
SSH	Secure Shell (Turvallinen pääteyhteys)
UDP	User Datagram Protocol (Tiedonsiirrossa käytettävä yhteydetön protokolla)
VCPU	Virtual Central Processing Unit (Virtuaalinen prosessori)
VLAN	Virtual Local Area Network (Virtuaalilähiverkko)
VMM	Virtual Machine Monitor (Virtuaalikoneita hallitseva komponentti)
VSA	Virtual Software Package Archive (Altiris SVS:n virtuaalinen ohjelma-arkisto)
VSP	Virtual Software Package (Altiris SVS:n virtuaalinen ohjelmapaketti)

1 JOHDANTO

Opinnäytetyön sisältö koostuu teoriaosuudesta ja käytännön osuudesta. Toisessa luvussa kerrotaan eri tarkoituksiin soveltuvista virtualisointimenetelmistä. Luvussa kolme käsitellään nykyisin tarjolla olevia ratkaisuja eri virtualisointitarpeisiin. Neljännessä luvussa on käsitelty nykyisten virtualisointitekniikoiden käyttötarkoituksia ja käyttömahdollisuuksia erilaisiin tarpeisiin. Viidennessä luvussa esitetään DNA Oy:n Etelä-Suomen paikallisorganisaation käyttöympäristö virtualisointiin liittyvin osin, miten virtualisointia on toteutettu, millaisiin ongelmiin on törmätty, ja miten virtualisointia voitaisiin jatkokehittää omaan käyttöön tai myytäväksi tuotteiksi.

Opinnäytetyön tavoitteena on kartoittaa ratkaisumahdollisuuksia ohjelmien yhteensopivuusongelmien ja jakelun helpottamiseksi sekä suunnitella ja toteuttaa ohjelmistovirtualisointia todellisessa tuotantoympäristössä. Työn piiriin kuuluu ohjelmistovirtualisoinnin lisäksi laitealustojen virtualisoinnin (virtuaaliset palvelimet) suunnittelu ja toteutus sekä tutustuminen ns. palveluvirtualisointiin. Vaikka kaikki opinnäytetyössä läpi käytävät tekniikat eivät suoranaisesti ole virtualisointitekniikoita, kuuluvat ne silti työn alueeseen joko vaihtoehtoisina ongelmanratkaisumenetelminä tai saavutettavien hyötyjen laajennusmahdollisuuksina. Laitteistovirtualisoinnin osalta keskitytään tuotantokäyttöön soveltuviin virtualisointijärjestelmiin ja jätetään työasemakäyttöön tarkoitetut järjestelmät vähemmälle huomiolle.

Ohjelmisto- ja laitteistovirtualisoinnin tavoitteena on helpottaa tietojärjestelmien (työasemat ja palvelimet) ylläpitoa, huoltoa sekä elinkaaren hallintaa. Omien tietojärjestelmien käytön helpottumisen lisäksi virtualisointitekniikoita on tarkoitus hyödyntää vaihtoehtona myös asiakkaille tarjottavien järjestelmien ja palveluiden toteuttamisessa.

2 VIRTUALISOINTI YLEISESTI

2.1 Virtualisointi konseptina

Tietotekniikan termeissä virtualisoinnilla tarkoitetaan tietokoneresurssien esitystä abstraktilla tasolla. Virtualisoinnin avulla siis piilotetaan todelliset resurssit näennäisen (virtuaalisen) resurssin taakse. Virtuaalinen resurssi voi olla esim. sovellus, tietokone tai tallennusväline. Resurssin loppukäyttäjä hyödyntää resurssia aivan kuin ennenkin, vaikka kyseinen resurssi on virtuaalinen. Virtualisoinnin avulla voidaan jakaa yhtä fyysistä resurssia useammalle loppukäyttäjälle, jotka eivät koe käytännön eroa todellisen ja virtuaalisen resurssin välillä. (Wikipedia 2008b.)

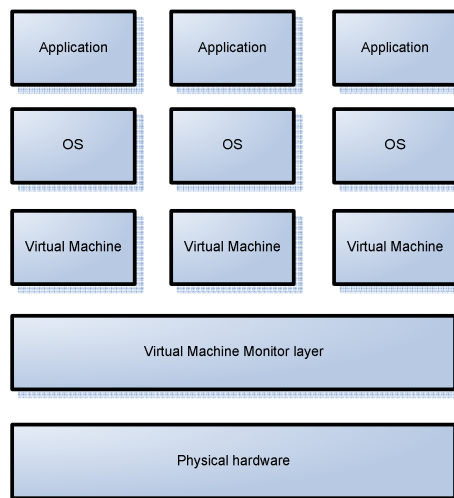
Virtualisoinnilla on tarkoitus luoda säästöjä helpottamalla ja keskittämällä hallintaa, tehostamalla hankittujen laiteresurssien käyttöä, ja nopeuttamalla resurssien käyttöönottoa. Virtualisointia toteutetaankin erilaisilla virtualisointitekniikoilla haluttujen etujen mukaisesti. (Sean, Donahue, Flanagan, Kelly, Kinney, Yuen, Lanoix, Kim, Richardson 2006, 27; Wikipedia 2008b.)

2.2 Laitteistovirtualisointi

Laitteistovirtualisoinnissa ajatuksena on usean palvelimen keskittäminen yhdelle fyysiselle laitteelle. Tällaisista virtualisoiduista palvelimista tai tietokoneista käytetään nimitystä virtuaalikone (englanniksi guest), kun taas näitä virtuaalikoneita ajavaa fyysistä laitetta kutsutaan nimellä isäntäkone (englanniksi host). Virtualisointia on tehty jo 60-luvulta lähtien, jolloin IBM käytti testissään isäntäkoneena IBM 7044 (M44) laitetta, jossa oli virtuaalikoneita pääkoneen (44X) testiimagesta. Virtualisoinnin alkua ajoilta on kotoisin on vieläkin käytössä oleva IBM:n ja MIT:n (Massachusetts Institute of Technology) yhteistyössä kehittämä VMM (Virtual Machine Monitor) periaate. (ProTrainIT Oy 2006, 1.2 – 1.3.)

Laitteistovirtualisointi on yksi eniten käytetyistä virtualisointitavoista käyttöjärjestelmien virtualisoinnissa. Tällaista menetelmää käytettäessä virtualisointi ei vaadi muita hankintoja, kuin fyysisen laitteiston ja VMM-kerroksen toteuttavan ohjel-

miston. VMM-kerros toteutetaan suoraan laitetason päälle, ja VMM-kerroksen yläpuolelle tulee virtuaalinen laitteisto sisältäen esim. muistin, prosessorin, BIOSin (Basic Input-Output System) ja tallennusvälineen eli kiintolevyn. Tähän virtuaaliseen laitteeseen asennetaan käyttöjärjestelmä samoin kuin fyysiselle alustalle. Virtuaalikoneen käyttöjärjestelmä ei tiedä olevansa virtuaalikoneessa, vaan näkee laitteiston samaan tapaan kuin laitteiston ollessa fyysinen, vaikka todellisuudessa VMM-kerros välittääkin virtuaalikoneiden pyynnöt suoritettavaksi isäntäkoneen laitteistossa. Laitteistovirtualisoinnin toimintaperiaate ilmenee kuviosta 1, jossa virtuaaliset käyttöjärjestelmät on eriytetty toisistaan ja niiden toimintaa ohjaa VMM-kerros.



KUVIO 1. Laitteistovirtualisoinnin toimintaperiaate. (ProTrainIT Oy 2006, 1.5.)

Virtuaalinen laitealusta tuo joustavuutta ja uusia mahdollisuuksia toteuttaa vikasietoisuutta. Koska käyttöjärjestelmälle näkyvä laitealusta on aina sama riippumatta fyysisestä laitealustasta, jolla virtualisointi toteutetaan, voidaan virtuaalikoneita siirtää isäntäkoneesta toiselle ilman, että virtuaalikone huomaa muutosta laitteistokokoonpanossaan. Vikasietoisuutta voidaan saada aikaiseksi joko kopioidamalla virtuaalikoneet aika-ajoin isäntäkoneelta toiselle, säilyttämällä virtuaalikoneiden tiedostoja levyjärjestelmällä, jota isäntäkoneet pääsevät käyttämään yhteisesti tai hankkimalla edistyneemmän virtualisointiympäristön, joka toimii klus-

teri-tyyppisesti ja tarjoaa automaattisen toipumisen vikatilanteista. (ProTrainIT Oy 2006, 1.14.)

Virtuaalikoneiden suurimmat edut taloudellisten etujen lisäksi ovat siis

- eristys, eli jos yhdessä virtuaalikoneessa on vikaa, muut jatkavat toimintaansa
- enkapsulointi, eli koko virtuaalikone näkyy isännälle pienenä määränä tiedostoja
- yhteensopivuus, eli virtuaalikoneen käyttöjärjestelmä näkee alustanaan standardin PC:n, eikä tarvitse erityisiä laiteohjaimia toimiakseen
- laitteistoriippumattomuus, eli virtuaalikone ei näe yksityiskohtia isäntäkoneesta, ja virtuaalikone voidaan näin ollen siirtää toiseen isäntään ongelmitta

(VMWare Inc. 2006, 1.2 – 1.3.)

Virtualisoinnista on pelkkien tietoteknisten etujen lisäksi myös energiataloudellista hyötyä. Konesaleissa sähkönkulutus jakautuu jäähdytyksen, ja laitteistojen kulutuksen kesken niin, että 55% energiasta menee konesalin jäähdytykseen, ja loput 45% itse tietotekniikkaan, joka synnyttää lämmön. Energiankulutus konesaleissa on Yhdysvaltojen ympäristösuojeluviranomaisen EPA:n (U.S. Environmental Protection Agency) mukaan kaksinkertaistunut viiden vuoden aikana. Konesalien energiankulutuksen kasvuun on vaikuttanut palvelukohtaisten palvelinten käyttö, eli käytännössä uutta käyttöön otettavaa palvelua varten on hankittu fyysinen laite vain tähän käyttötarkoitukseen. Tämän vuoksi palvelimien keskimääräinen kuormitus onkin vain 20 prosentin luokkaa. Käyttämällä virtuaalisia palvelimia fyysisten laitteiden sijaan saavutetaan edelleen samat edut lisäämättä fyysisten laitteiden määrää konesalissa. Tästä johtuen palvelinten virtualisoinnilla voidaan vaikuttaa konesalien energiankulutukseen sekä itse tietoteknisten laitteiden että jäähdytyksen osalta. (Lukkari 2007, 16 – 17; Torikka 2007, 18.)

2.3 Ohjelmistovirtualisointi

Ohjelmat aiheuttavat kustannuksia yritykselle sekä niiden hankintana että hallintana. Hankintakustannukset näkyvät suoraan ohjelmaa hankittaessa, mutta hallintakustannuksia ei voida ennalta täysin määrittää. Korkeiden kustannusten lisäksi sovellusten ongelmien ratkaisu, korjaus, päivittäminen ja asennus voi olla turhauttavaa. (Altiris 2006, 1.)

Ohjelmistojen kohdalta hallintakustannukset aiheutuvat seuraavista osatekijöistä: jakelu (Deployment), ylläpito (Updates), tuki (Support) ja käytön lopettaminen (Termination). Ohjelmistovirtualisoinnilla helpotetaan ohjelmien hallintaa, ja näin ollen ohjelmien kokonaiskustannukset pysyvät matalampina perinteisiin menetelmiin verrattuna. (Sean ym. 2006, 21.)

Jakelu

Sovelluksen jakelu on ensimmäinen asia joka täytyy tehdä, jotta haluttu sovellus saadaan käyttöön loppukäyttäjille. Jakelu voidaan toteuttaa perinteisillä menetelmillä asentamalla ohjelmisto yksittäin jokaiselle tietokoneelle tukihenkilön toimesta, jakelemalla ohjelmistopaketti ESD-järjestelmän (Electronic Software Distribution) avulla tai käyttäen Terminal Server- tai Citrix Metaframe-tyyppisiä etäkäyttöratkaisuja. (Sean ym. 2006, 21 – 22.)

Sovelluksen asennus tietokoneelle lisää ja/tai muuttaa tietokoneella olevia sovelluksia, ohjelmakirjastoja, tiedostoja sekä mahdollisesti myös käyttöjärjestelmän osia ja rekisteriasetuksia. Sovellukset asentavat osiaan eri hakemistoihin tietokoneen kiintolevyllä, eivätkä suinkaan enää pysy omissa hakemistoissaan kuten ennen Windows 95:n yleistymistä. Juuri tällaisen toiminnan vuoksi yksi sovellusten jakelun perusongelmista ovatkin sovellusten keskinäiset ristiriidat sovellusten ja ohjelmakirjastojen versioiden välillä. Ristiriitojen selvittäminen ja sovellusten toimintakuntoon saattaminen voi olla haastavaa jo pelkästään kahden eri sovelluksen kesken, ja ristiriitojen mahdollisuus vain kasvaa eksponentiaalisesti sovellusten määrän kasvaessa. Ongelma ei olisi näin laaja, jos yritykset voisivat toimia

hyvin rajoitetulla ja keskenään testatulla sovellusvalikoimalla, mutta sovellusten tarve (ja näin ollen myös sovellusten määrä) kasvaa ja ohjelmistoversiot muuttuvat jatkuvasti. Pahimmissa yhteensopivuusongelmissa voidaan joutua jopa tilanteeseen, jossa yhdellä käyttäjällä on käytössään useita tietokoneita, koska sovellukset eivät vain yksinkertaisesti toimi keskenään yhdessä työasemassa. (Sean ym. 2006, 21 - 22; Brown 2005, 3.)

Päivitys

Sovellusten päivitys on pakollinen prosessi, joka aiheuttaa kustannuksia yrityksen sovellusten hallinnassa. Mitä suositumpi sovellus on ja mitä pidempään se on markkinoilla, sitä suurempi määrä ohjelmistopäivityksiä ja korjauksia yleensä tulee. Päivityksien asentaminen yksittäin yritysten työasemille ja etäkäyttöpalvelimille on aikaa vievää ja näin ollen myös kallista. Kustannusten kertymisen lisäksi on hankalaa saada päivitykset asennettua nopeasti ja hallitusti jokaiselle yrityksen työasemalle. Tämän lisäksi useampi asennuskerta tarkoittaa myös useampaa mahdollisuutta epäonnistuneeseen ohjelmapäivitykseen. (Sean ym. 2006, 21.)

Tuki

Sovellusten tuen tarve kasvaa sovellusten määrän kasvaessa. Sovellusten käyttöopastuksen lisäksi yrityksen IT-henkilöstöä kuormittavat ristiriidat eri sovellusten ja sovellusversioiden välillä. Tuen tarvetta aiheuttavat myös sovellusten käyttäjät itse, esim. poistamalla vahingossa joitain sovelluksen tarvitsemia ohjelmakomponentteja. (Sean ym. 2006, 21.)

Käytön lopettaminen

Jossain vaiheessa sovellukset saavuttavat elinkaarensa pään, jolloin ne poistetaan käytöstä tai korvataan toisella sovelluksella, ja käytöstä poistetut sovellukset poistetaan tietokoneilta. Perinteisesti sovelluksen poistaminen on tehty tukihenkilöiden toimesta käsityönä (tai automaattisilla skripteillä), mistä johtuen sovelluksia ei välttämättä ole poistettu kaikilta tietokoneilta, ja ne voivat aiheuttaa

ongelmia myöhemmässä vaiheessa esim. ristiriitaisuuksilla muiden uudempien sovellusten kanssa. (Sean ym. 2006, 21.)

2.4 Palveluvirtualisointi

ASP:lla (Application Service Provider/Providing) tarkoitetaan sovelluksen vuokraamista toimittajalta valmiina kokonaisuutena. Tällöin palvelun vuokraaja hallinnoi ja ylläpitää palvelua sekä pääsynhallintaa usealle osapuolelle samoissa keskitetyissä laitteistoissa toimiviin palveluihin. Tällaisessa tilanteessa voidaan ajatella palvelun olevan virtualisoitu, koska yhden suuren palvelun sekä tämän palvelualan resurssit ja toiminnot ovat jaettu useampaan osaan useammalle asiakkaalle. (TIEKE Tietoyhteiskunnan kehittämiskeskus ry 2008.)

ASP-toimintamalli tarjoaa yrityksille ja yhteisöille edullisen vaihtoehdon oman järjestelmän rakentamiselle ja ylläpidolle. Suurimmat edut tulevat erityisesti pienille yrityksille pienistä aloituskustannuksista ja erittäin nopeasta käyttöönottoajasta. ASP-ratkaisuissa käyttökustannukset määräytyvät usein käyttäjämäärän mukaisesti. Toimintamallilla saavutetaan taloudellista säästöä aloituskustannusten lisäksi myös ylläpito- ja tukihenkilöstön tarpeen vähenemisen kautta. (Brain 2008.)

ASP-mallilla toteutetuista palveluista esimerkkinä mainittakoon virtuaalisten internet-sivustojen tai sähköpostipalveluiden tarjoaminen. Tällaisista palveluista etenkin sähköpostipalveluita on saatavana sekä ilmaispalveluina että lisäominaisuuksilla vastustettuina maksullisina palveluina. Vaikka edellä mainitut palvelut ovatkin perinteisiä palvelutuotteita, voidaan ASP-mallia hyödyntämällä tarjota lähes millaista palvelua tahansa. (Brain 2008.)

3 NYKYINEN TARJONTA

3.1 Etäkäyttöratkaisut

3.1.1 Telnet/SSH/X Window System

Tässä kappaleessa käsiteltäviä etäkäyttöratkaisuja käytetään pääasiassa UNIX käyttäjärjestelmien yhteydessä. Tällaiset etäkäyttöratkaisut eivät ole ratkaisuvaihtoehto työssä käsiteltävään ongelmaan eikä niitä näin ollen käsitellä syvällisesti tämän työn puitteissa.

Telnet ja SSH (Secure Shell) ovat tekstipohjaiseen etäkäyttöön soveltuvia ratkaisuja. Niiden avulla on mahdollista käyttää palvelimen komentorivikäyttöliittymää esim. internetin välityksellä. Telnet on salaamaton protokolla, ja näin ollen asiakkaan ja palvelimen välinen kommunikointi (mukaan lukien salasanat) tapahtuu selkokielisenä. Parannuksena telnetiin SSH tarjoaa sekä salauksen että muita lisäominaisuuksia. Yksi usein käytetty SSH:n lisäominaisuus on mahdollisuus tunnetuista X Window Systemin asiakasohjelmien data kulkemaan salatun SSH-yhteyden kautta. (Wikipedia 2008a.)

X Window System on UNIX maailman de facto standardiksi muodostunut graafisen käyttöliittymän moottori. X Window Systemistä käytetään yleisesti lyhyempää nimeä X. X:n modulaarinen client-server arkkitehtuuri jakautuu kahteen toisiinsa liittyviin osaan: sovelluksiin, eli asiakkaisiin että palvelimeen. Näistä osista palvelinta ajetaan käyttäjän työasemalla, mutta sovelluksia voidaan ajaa oman työaseman lisäksi jollain muulla työasemalla tai palvelimella, josta sovelluksen näkymä kuljetetaan verkon välityksellä X palvelimelle ja X palvelimelta käyttäjän toimenpiteet sovellukselle. (The Linux Information Project 2006.)

X Window System kehitettiin 1980-luvun puolessa välissä MIT:n ja Digital Equipment Corporationin yhteisprojektina. Projektin tarkoituksena oli tarjota opiskelijoille helppo pääsy tietokoneresursseihin olemassa olevilla epäyhteensopivilla tietokonelaitteilla. Nykyisin X:n käytössä oleva X11-protokolla julkaistiin

jo vuonna 1987, ja näin ollen voidaankin turvallisesti sanoa, ettei X:n kaltainen etäkäyttö ole uusi asia UNIX maailmassa. (The Linux Information Project 2006.)

3.1.2 Citrix Presentation Server

Citrix Presentation Server mahdollistaa sovellusten asentamisen, hallinnan ja julkaisemisen yhdestä keskitetystä paikasta. Näitä julkaistuja resursseja voidaan käyttää mistä tahansa millä tahansa laitteella, jolle on olemassa asiakasohjelma. Presentation Server -palvelinohjelmisto toimii Windows Server 2003 32- ja 64-bittisellä versiolla, ja sillä voidaan julkaista kaikki käyttöjärjestelmässä toimivat Windows- tai selainpohjaiset sovellukset. Sovelluksia voidaan ajaa joko palvelimella tai siirtää ne asiakkaalle suoratoistotekniikan avulla. Tässä kappaleessa käsitellään kuitenkin etäkäyttöosuutta, jolloin sovellukset suoritetaan palvelimella. (Citrix Systems Inc. 2007, 31.)

Presentation Server mahdollistaa useamman yhtäaikaisten käyttäjien käyttäminen palvelimella asennettuja sovelluksia ja resursseja. Eri käyttäjät ja heidän suorittamansa sovellukset toimivat omassa käyttäjäkohtaisessa istunnossa, jotka ovat suojattuja eivätkä vaikuta toisiinsa. Presentation Server tarvitsee toimiakseen palvelimen, johon on asennettu Windows 2003 Server, Terminal Services ja Citrix Presentation Server -ohjelmisto. Palvelinohjelmiston lisäksi palvelu tarvitsee toimiakseen sekä Microsoft Terminal Server- että Citrix-lisenssit. (Citrix Systems Inc. 2007, 32.)

Citrix Presentation Serveriä on saatavana kolmena eri versiona: Advanced-versio sisältää oleelliset toiminnot sovellusten julkaisemiseksi. Enterprise-versio lisää Advanced-versioon hallintamahdollisuuksia monimutkaisempien ympäristöjen tehokkaaseen hallintaan. Platinum-versio on kaikkein monipuolisin Presentation Server-versio ja sisältää Enterprise-version ominaisuuksien lisäksi tietoturvan ja suorituskyvyn hallintaan liittyviä ominaisuuksia. Tarkemmat eroavaisuudet versioiden välillä löytyvät Citrixin sivuilta ominaisuusmatriisina. (Citrix Systems Inc. 2007, 33.)

Citrix asiakasohjelmia on saatavilla Windows-, Java- ja UNIX-versioina. Tällaisella laajalla käyttöjärjestelmätuella mahdollistetaan sovellusten käyttö Windows ja UNIX-työasemista esim. Macintosh tai Windows CE laitteisiin. Windows-ympäristöön asiakasohjelmasta on julkaistu kolme versiota, Client for Web, Program Neighborhood Agent ja Program Neighborhood. Palvelun ylläpitäjä voi valita yrityksen käyttöön sopivimman asiakasohjelman, joka täyttää järjestelmälle asetetut toiminnalliset vaatimukset. Citrix asiakasohjelmat käyttävät yhteydessään ICA (Independent Computing Architecture) -protokollaa, joka sisältää kehittyneet tietoturva- ja tiedonsiirto-ominaisuudet. Asiakasohjelman ja palvelimen välinen liikenne voidaan salata vahvalla AES- (Advanced Encryption Standard) salauksella. Asiakasohjelmien ominaisuusmatriisi on ladattavissa Citrixin verkkosivuilta. (Citrix Systems Inc. 2007, 89 – 96.)

Presentation Serverin hallinta tapahtuu kolmella hallintasovelluksella. License Management Console on verkkopohjainen sovellus, jolla hallitaan ja valvotaan lisenssien käyttöä palvelinfarmissa. Access Management Consolella hallitaan mm. palvelimien yhteyksiä, tilaa ja julkaistuja sovelluksia. Presentation Server Consolella hallitaan esim. palvelinfarmin resursseja, kuormantasausta ja tulostimia. Näiden työkalujen avulla voidaan siis hallita kokonaista palvelinfarmia palvelimien määrästä riippumatta. (Citrix Systems Inc. 2007, 43 – 48.)

3.1.3 Microsoft Terminal Server

Microsoft Terminal Server on palvelu, joka mahdollistaa Windows-työpöydän käytön melkein pä millä tahansa laitteella, mukaan lukien sellaiset laitteet, joissa ei ole mahdollista ajaa Windowsia. Käyttäjän käyttäessä sovellusta Terminal Serverillä, itse sovelluksen suoritus tapahtuu palvelimella, ja näin ollen pelkästään näppäimistön, hiiren ja kuvaruudun informaatio siirretään verkon yli. Jokaisella käyttäjällä on käytössään oma yksittäinen istunto, joita palvelimen käyttöjärjestelmä hallitsee eikä näistä istunnoista näe toisten istuntojen tapahtumia. (Microsoft 2005, 1.)

Terminal Serverin eduksi perinteiseen henkilökohtaisen tietokoneen käyttöön nähden voidaan laskea sovellusten nopea ja keskitetty jakelu sekä käyttömahdolli-

suus mistä tahansa. Terminal Server -ympäristössä sovellus asennetaan ja päivitetään vain palvelimelle, jota kautta käyttäjät saavat ajan tasalla olevat ja valmiiksi konfiguroidut sovellukset käyttöönsä. Suurin etu sovellusten keskitetystä päivittämisestä on sellaisia sovelluksia käytettäessä, jotka tarvitsevat jatkuvaa päivittämistä. Tällaisella järjestelmällä voidaan myös olla varmoja, että jokainen käyttäjä käyttää sovelluksen viimeisintä versiota. (Microsoft 2005, 1.)

Terminal Serverin käyttämistä varten tarvitaan asiakasohjelmisto, joka muodostaa RDP (Remote Desktop Protocol) yhteyden Terminal Serverille. Asiakasohjelmisto tulee Windows XP:ssä, Windows Server 2003:ssa ja uudemmissa käyttöjärjestelmän mukana. Windows XP:ssä asiakasohjelmiston nimi on Remote Desktop Connection. Sovelluksen päänäkymä on hyvin yksinkertainen ja pelkistetty (kuvio 2), mutta lisäasetuksista yhteyden asetuksia voidaan säätää tarkemmin. Lisäasetuksista voidaan yhteyden nopeuden mukaisesti joko poistaa tai sallia työpöydän taustakuvan siirto, teemat jne. RDP yhteyden yli on mahdollista kuljettaa edellä mainitun informaation lisäksi myös erinäisiä uudelleenohjauksia, kuten:

- tiedostojärjestelmä: asiakaslaitteen levyasemat (myös verkkolevyt) näkyvät levyasemina Terminal Serverillä.
- sarjaportit.
- tulostimet: asiakaslaitteen kaikki tulostimet (myös verkkotulostimet) näkyvät palvelimelle.
- äänet: varoitusäänet voidaan siirtää palvelimelta asiakaslaitteelle.
- Smart Card: Smart Card-kirjautumista voidaan hyödyntää myös Terminal Server -istunnoissa.

Tiettyjä uudelleenohjauksia käytettäessä näytetään turvallisuusvaroitusta, joka on hyväksyttävä, jotta uudelleenohjaukset toimivat. (Microsoft 2005, 3 – 6; Microsoft 2007.)



KUVIO 2. Remote Desktop Connectionin päänäkymä

Windows Server 2003:sta lähtien etäyhteys on jaettu Remote Desktop for Administration –ominaisuuteen ja Terminal Serveriin. Windows 2000 Serverissä tämä oli toteutettu Terminal Services -ominaisuuden kahtiajaolla. Remote Desktop for Administrationia käytettäessä vain palvelimen ylläpitäjät voivat kytkeytyä palvelimen etätyöpöytään, kun Terminal Server mahdollistaa käytön suuremmalle määrälle käyttäjiä, joiden ei välttämättä tarvitse olla palvelimen ylläpitäjiä. Terminal Servereitä hallitaan Terminal Services Managerilla, jolla voidaan hallita useampaa palvelinta yhdellä hallintanäkymällä. (Microsoft 2005, 7 – 10.)

Suurissa toteutuksissa Terminal Servereistä voidaan luoda palvelinfarmeja. Tällainen konfiguraatio tuo järjestelmään sekä vikasietoisuutta että mahdollistaa kuorman tasaamisen palvelimien välillä. Session Directory ominaisuus mahdollistaa käyttäjien kytkeytymisen samalle palvelimelle kuin missä heillä on jo istunto olemassa, ja näin ollen istuntoja voidaan jättää auki myös palvelinvarmin ollessa kyseessä. Palvelinvarmia suunniteltaessa ja toteuttaessa on huomioitava, että käyttöjärjestelmänä on oltava Windows Server 2003 Enterprise Edition. (Microsoft 2005, 11.)

3.1.4 Ratkaisujen kustannukset ja vertailu tilanteeseen ilman etäkäyttöä

Työssä esitellyistä UNIX -perustaisista etäkäyttöratkaisuista on olemassa sekä avoimeen lähdekoodiin perustuvien ilmaisten ratkaisujen lisäksi myös maksullisia järjestelmiä. Näistä maksullisiin tuotteisiin on luonnollisesti saatavilla paremmin tukea sovelluksen toimittajan suunnalta, mutta myös ilmaisiin vaihtoehtoihin löytyy apua sovellusten käyttäjäyhteisöistä internetistä.

Etäkäyttöratkaisujen osalta työssä keskityttiin vain monen käyttäjän yhtäaikaisiin ratkaisuihin, koska yhden käyttäjän etäkäyttöratkaisut (esim. näytönkaappausratkaisut) eivät ole kustannusten kannalta järkeviä yritysten etäkäyttöratkaisuna. Tällöin jokaista etäkäyttöyhteyttä varten tarvittaisiin käyttöön erikseen valjastettu tietokone käyttöjärjestelmineen ja sovelluksineen. Lisenssikustannusten lisäksi tällaisessa tapauksessa useimmiten tulee huomioon otavaksi myös kasvava virrankulutus.

Etäkäyttöratkaisujen kustannukset muodostuvat tarvittavasta laitteistosta, ohjelmistolisensseistä, järjestelmän toteutuksesta ja operoinnista sekä käyttäjäkoulutuksesta. Sekä Citrix että Microsoft Terminal Server tarjoavat asiakasohjelmiston maksutta käyttöön, mutta käyttöoikeudesta maksetaan käyttäjäkohtaisesti per palvelin tai palvelinfarmi. Citrixin etäkäyttöratkaisussa tarvitaan Citrixin omien client lisenssien lisäksi myös Microsoftin Terminal Server Client Access License jokaista käyttäjää tai asiakaslaitetta kohden. Citrixin lisensointimalli eroaa Terminal Serverin vastaavasta niin, ettei lisenssejä sidota yhteyden muodostamisvaiheessa kiinteästi käyttäjälle, vaan ne vapautuvat yhteyden purkamisen yhteydessä, jolloin vapautunut lisenssi voidaan osoittaa toisen käyttäjän käyttöön.

Etätyöskentelyn mahdollistamisen lisäksi tietojärjestelmien etäkäytöllä saavutetaan epäsuoria etuja, jollaisiksi voidaan ajatella keskitetysti hoidettava sovellusten versionhallinta, tietoturva ja asetukset. Näin jokaiselle työasemalle ei täydy erikseen huolehtia sovellusversion ajantasaisuudesta tai mahdollisista poikkeavuuksista sovelluksen asetuksissa. Myös sovelluksen tuominen loppukäyttäjän käyttöön helpottuu ja nopeutuu etäkäyttöratkaisujen käytön myötä. Vaikka ratkaisulla voidaan saavuttaa suuriakin hyötyjä esim. sovellusten versionhallinnan ansiosta, totelevat sovelluksia tarjoavat palvelimet silti samoja lainalaisuuksia kuin loppukäyttäjien työasemat, ja näin ollen etäkäyttö ei ratkaise mahdollisia yhteensopivuusongelmia sovellusten välillä.

Sovelluksen etäkäytön mahdollistavia palvelimia ja niillä olevia sovelluksia huollettaessa ja päivitettäessä on huomioon otettava käyttökatkon tai epäonnistuneen sovelluspäivityksen vaikutus kaikkiin sovellusta tai etäkäyttöpalvelinta käyttäviin käyt-

täjiin. Tästä johtuen kyseisten palvelimien ja sovellusten huollot suunnitellaankin tehtäväksi varsinaisen toimistoajan ulkopuolella suoranaisten haittojen minimoimiseksi.

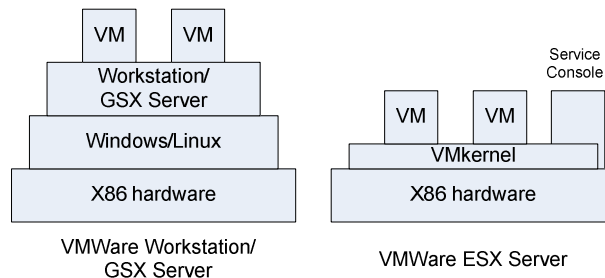
Vaikka työn varsinaiseen aihepiiriin ei kuulukaan päätteohjainen tietojenkäsittely, se on mahdollista etäkäyttöratkaisujen myötä. Tällaisessa tilanteessa käyttäjän pöydällä voi näytön ja ohjainlaitteiden lisäksi sijaita halpa thin client -tyyppinen tietokone, jonka suorituskyky ja käyttöjärjestelmä itsessään ovat hyvin rajattuja ja tietokoneen tarkoitus onkin vain ottaa yhteyttä etäkäyttöä tarjoavaan palvelimeen sovellusten käyttöä varten. Tällaista ratkaisua suunnitellessa tulee kartoittaa käyttäjien tarpeet sovellusten ja niiden vaatimien konetehojen suhteen, jotta tuottavuus ei kärsi liian tehottomien palvelimien vuoksi. Kustannustekijöiden lisäksi näin on mahdollista parantaa yrityksen tietoturva, koska sovelluksia eikä myöskään sovellusten tietoja säilytetä helposti hajoavilla tai varastettavilla työasemilla, vaan hyvin varmistetuilla ja fyysisesti tietoturvalisillä palvelimilla ja levyjärjestelmillä.

3.2 Laitteistovirtualisointi

3.2.1 VMWare

VMWare tarjoaa virtualisointiin kolmea erilaista ratkaisua, virtualisoinnin tarpeesta riippuen. VMWare ESX Server on virtuaalikoneiden ajamiseen kehitetty korkeaan suorituskykyyn tähtäävä alusta palvelinsovelluksille, ja VMWare GSX Server on kehittyneempi versio Workstationista (yhdeksi käyttäjälle yhdessä tietokoneessa käytettäväksi suunnattu tuote), ja ESX Server on tehty tukemaan useampaa yhtäaikaista virtuaalikonetta, Remote Consolea ja skriptausta. GSX Serveriä voidaan ajaa sekä Windows, että Linux-käyttöjärjestelmissä, ja se tukee enintään 64 virtuaalikonetta per GSX-palvelin. ESX Server puolestaan tukee virtuaalista SMP:tä (Symmetric multiprocessing) ja 80 VCPU:ta (Virtual Central Processing Unit) per ESX-palvelin. ESX Serveriä käytettäessä on mahdollista käyttää VMotion-tekniikkaa. Molemmat tukevat 3.6GB muistia per virtuaalikone. (VMWare Inc. 2006, 0.3, 1.6, 1.10.)

VMWaren GSX Server tarvitsee alustakseen joko Linux tai Windows käyttöjärjestelmän, jonka päälle ohjelmisto asennetaan. ESX Server puolestaan asennetaan suoraan laitteistoon ja ESX Server sisältää itsessään käyttöjärjestelmän (joka perustuu osittain RedHat Linuxiin), jossa virtuaalikoneita ajetaan. Koska ESX Server asennetaan suoraan laitteistoon ilman erillistä käyttöjärjestelmää, on ESX:n tukema laitteistokanta rajoitetumpi kuin GSX Serverin tapauksessa. GSX Server puolestaan mahdollistaa laajemman laitteistotuen, koska GSX Serverin toiminta perustuu isäntäkoneen käyttöjärjestelmään, ja näin ollen virtuaalikoneet voivat käyttää kaikkia isäntäkoneen näkemiä laiteresursseja. Teknisen rakenteensa (kuvattu kuviossa 3) vuoksi ESX Serverillä saavutetaan 83 - 98% suorituskyky natiiviin laitteistoon verrattuna, kun sekä Workstation että GSX jäävät 70 - 90% suorituskyvyn tasolle. GSX Server -tapauksessa kyseinen ohjelmisto tarjoaa VMM-kerroksen virtuaalikoneille, kun taas ESX Server -tapauksessa sen tekee VMkernel.



KUVIO 3. VMWare arkkitehtuuri. (VMWare Inc. 2006, 1.6 - 1.9.)

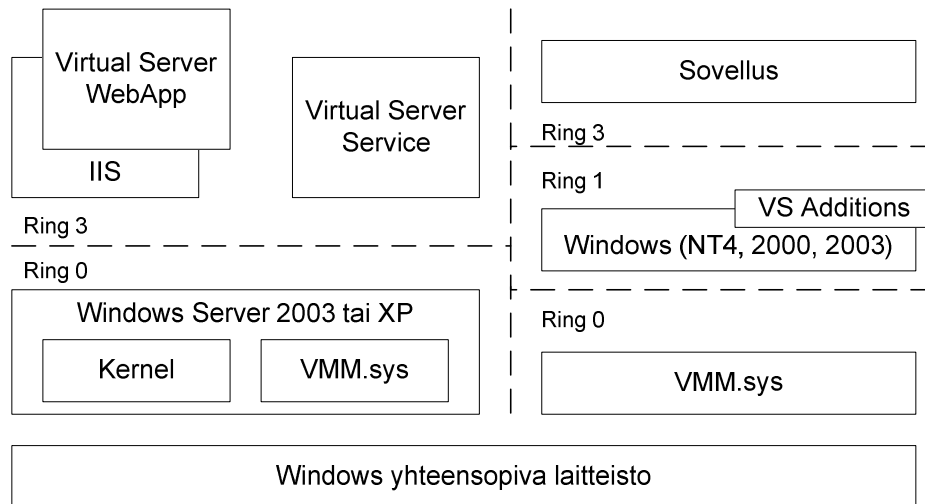
VMWare GSX Server tarvitsee toimiakseen sekä kernel-tason että user-tason komponentit. Virtuaalikoneen tarvitessa resursseja isäntäkoneen VMM-kerros herättää VMWare sovelluksen isäntäkoneessa, joka puolestaan varaa laiteresurssit. Tämä toimintatapa aiheuttaa kaksi ”ylimääräistä” kontekstin vaihtoa joka kerran, kun virtuaalikone haluaa kirjoittaa tietojansa kiintolevylle tai käyttää verkkoa. ESX Serverissä virtuaalikoneet käsittelevät koko laitteistoa suoraan, ja näin ollen virtuaalikoneiden toiminta on tehokkaampaa. ESX Serverin rajattu laitteistotuki mah-

dollistaa laitteiston suoran käsittelyn, mutta myös laiteviat voivat olla ESX:lle kohtalokkaampia. (VMWare Inc. 2006, 1.8 - 1.9.)

VMWare-virtuaalikoneiden hallinta tapahtuu VMWare VirtualCenterin avulla keskitetysti. VirtualCenterillä voidaan hallita sekä GSX Serverissä, että ESX Serverissä olevia virtuaalikoneita. Hallintaominaisuuksiin kuuluu virtuaalikoneiden monitoroinnin, käynnistyksen, sammutuksen ja siirtelyn lisäksi myös virtuaalikoneiden konsoliyhteydet. VirtualCenterin ominaisuuksiin kuuluu myös VMotion. VMotionin avulla virtuaalikoneita voidaan siirtää ESX-järjestelmässä isäntäkoneelta toiselle ilman, että virtuaalikone sammutetaan tai pysäytetään siirron ajaksi. VMotionin toiminnan edellytyksenä on isäntäkoneiden kytkeytyminen samaan SAN (Storage Area Network) järjestelmään, sekä erillinen gigabitin taustaverkko ESX-palvelimien välille. Virtuaalikoneiden siirto laitteelta toiselle onnistuu synkronoimalla käynnissä olevan virtuaalikoneen tila uudelle ESX Serverille taustaverkon välityksellä. Synkronoinnin jälkeen virtuaalikone pysäytetään vanhalla ESX Serverillä ja herätetään käyntiin uudella. (VMWare Inc. 2006, 1.11.)

3.2.2 Microsoft Virtual Server

Microsoft Virtual Server on Microsoftin laitteistovirtualisointiin kehittämä ratkaisu. Microsoft Virtual Server on kehitetty Microsoft Virtual PC:stä, jonka alunperin on kehittänyt Microsoftin nykyisin omistama Connectic niminen yhtiö. Microsoft Virtual Server toteuttaa laitteistovirtualisointia tarjoamalla virtualisoinnin toiminnan kannalta tärkeän VMM-kerroksen. Toiminta perustuu luotettavaan laitteistoalustaan, siihen asennettuun käyttöjärjestelmään ja käyttöjärjestelmään asennettuun Virtual Server 2005 -ohjelmistoon. Käyttöjärjestelmään asennetaan VMM.sys laiteohjain kommunikoidaan virtuaalikoneiden VMM.sys:ien kanssa ja Virtual Server -palvelu virtualisoitujen palvelimien ajamiseksi. Isäntäpalvelimeen voidaan asentaa myös IIS-palvelu, jolloin samaan palvelimeen saadaan käyttöön myös Virtual Server Web Administrator -sivusto. Microsoft Virtual Server 2005:n arkkitehtuuri on kuvattu kuviossa 4, jossa vasen puoli on Virtual Server isäntäkone ja oikea puoli kuvaa virtuaalikonetta. Molemmat koneet toimivat samalla fyysisellä laitealustalla ja virtuaalikoneita voi olla käynnissä samanaikaisesti enemmänkin kuin yksi. (ProTrainIT Oy 2006, 1.5, 1.8, 2.2.)



KUVIO 4. Microsoft Virtual Server 2005 R2 arkkitehtuuri.

Virtual Server 2005 -ohjelmistoa voidaan käyttää sekä Windows Server 2003:ssa (R2 tai SP1 sekä Standard, Enterprise, Datacenter, Standard x64 tai Enterprise x64 -versio) että Windows XP:ssä. Windows XP asennusta tulisi käyttää ainoastaan testaukseen, ja tuotannon ratkaisut tulee toteuttaa Windows Server 2003 -käyttäjärjestelmällä. Virtual Server on optimoitu 2-8 prosessorille, mutta Virtual Server voi skaalautua jopa 32 prosessoriin ja 64GB muistiin asti. Virtuaalisina käyttöjärjestelminä tuettuna ovat Windows x86 -käyttöjärjestelmien (NT4.0:sta Windows Server 2003:een sekä Windows XP SP2) lisäksi myös Linux-käyttöjärjestelmät. Virtual Server 2005 kuitenkin rajoittaa virtuaalikoneen näkemän prosessorimäärän yhteen ja käytettävissä olevan muistin 3,6GB:en. Myöskään 64-bittisiä käyttöjärjestelmiä ei voida ajaa virtuaalikoneissa. (ProTrainIT Oy 2006, 2.3, 2.5, 2.7 - 2.8.)

Virtual Server tai Web Administrator eivät vaadi toimiakseen laitteistolta juuriakaan muistia tai levytilaa, mutta laitteistoa suunniteltaessa tulee huomioida virtuaalikoneiden tarvitseman muistin ja levytilan määrän lisäksi myös laitteiston korvattavuus toisella laitteella esim. laiterikon sattuessa. Suunnitteluesimerkiksi otettakoon tilanne, jossa ajetaan 10 virtuaalikoneessa Windows Server 2003 palvelimia, joiden muistin tarve on 512MB ja kiintolevytilan tarve 10GB. Tästä voidaan suoraan laskea virtuaalikoneiden tarvitsema muistinmäärä 5GB ja levytila 100GB.

Vielä tällaisella kokoonpanolla ei voida lähteä tilaamaan isäntäkonetta, vaan on huomioitava myös isännän muistin ja kiintolevyn tarpeet, jotka useimmissa tapauksissa ovat muistin osalta 512MB ja kiintolevyn osalta 6GB. (ProTrainIT Oy 2006, 2.2, 2.6.)

Virtual Server 2005 hallinta tapahtuu Virtual Server Administration Website -sivuston välityksellä selainta käyttämällä. Tämän sivuston avulla voidaan hallita useampaa Virtual Server -isäntäkonetta, eikä Administration Website -sivuston asentaminen itse isäntäkoneeseen ole välttämätöntä. Tyypillisimmässä käyttötapauksessa sivusto kuitenkin asennetaan isäntäpalvelimeen. Administration Website hallintasivusto vaatii käyttäjältään tunnistuksen, ja oletusasetuksilla ainoastaan isäntäkoneen pääkäyttäjät voivat hallita Virtual Services -palvelua. Tunnistautumisen jälkeen pääkäyttäjät pääsevät muokkaamaan sekä isäntäkoneen asetuksia että virtuaalikoneiden resursseja tai luomaan kokonaan uusia virtuaalikoneita. Virtuaalikoneiden käyttöä varten on olemassa myös Virtual Machine Remote Control -ohjelma, jolla saadaan suora yhteys virtuaalikoneeseen. (ProTrainIT Oy 2006, 4.2 - 4.6.)

3.2.3 Virtuozzo

SWSofin valmistaman Virtuozzon virtualisointi ei ole laitteistovirtualisointia vaan käyttöjärjestelmätason virtualisointia. Virtuozzo jakaa isäntäkäyttöjärjestelmän ja -laitteiston osiin eristetyiksi virtuaaliympäristöiksi. Nämä virtuaaliympäristöt ovat siirrettäviä kerroksia isäntäkäyttöjärjestelmän päällä, ja ne aiheuttavat vain vähän ylimääräistä kuormaa palvelimelle. Mahdollisimman pienen ylimääräisen kuorman vuoksi yhden fyysisen käyttöjärjestelmän päällä voidaan ajaa jopa sataa virtuaaliympäristöä. Virtuaaliympäristön suunnittelussa on huomioitu tarve siirtää palvelua tuottava sovellus toiselle laitteelle, ja näin ollen Virtuozzoa käytettäessä on mahdollista siirtää virtuaaliympäristö palvelimelta toiselle hyvin lyhyellä huoltokatkolla. (SWSof 2008, 5 – 6.)

Virtuozzon luomalla virtuaaliympäristöllä on omat prosessinsa, tiedostonsa, käyttäjänsä, pääkäyttäjätunnuksensa, täysi verkkoyhteys, järjestelmäkirjastot ja kaikki muut, mitä virtuaaliympäristö tarvitsee toimiaksensa kuin aito palvelin. Virtuozzo

käyttää vain yhtä käyttöjärjestelmää ja luo virtuaaliympäristöistä linkin takaisin isäntäkäyttöjärjestelmään. Linkkien ansiosta virtuaaliympäristön koko pysyy hyvin pienenä ja ympäristö tehokkaana. Tehokkuutta lisää myös rakenne, jossa itse käyttöjärjestelmää on käynnissä vain yksi kappale. (SWSOft 2008, 6.)

Virtuaaliympäristöt ovat suojattu ja erotettu toisistaan useammalla tietoturvakerroksella eikä yhden virtuaaliympäristön tapahtumat vaikuta toisiin ympäristöihin. Virtuozzon Kernel Service Abstraction Layer välittää järjestelmäpyynnöt käyttöjärjestelmän ytimelle ja estää virtuaaliympäristöä kaatamasta koko palvelinta. Virtuozzo File System puolestaan takaa, etteivät virtuaaliympäristöt voi sotkea toisensa tai isäntäkäyttöjärjestelmän tiedostoja. Copy-on-write tekniikalla muutetusta tiedostosta luodaan oma kopio virtuaaliympäristön sisälle, ja näin virtuaalinen ympäristö voi tallentaa näennäisiä muutoksia käyttöjärjestelmään. (SWSOft 2008, 6.)

Virtuozzon dynaamisen resurssienjaon ansiosta CPU, muisti, verkkoliikenteen kapasiteetti, levytila ja muut järjestelmän resurssit voidaan jakaa virtuaaliympäristöjen kesken. Resursseja voidaan taata virtuaaliympäristölle tietyn verran, tai niiden käyttöä voidaan rajoittaa määrittelemällä suurin sallittu arvo. Virtuaaliympäristöt voivat myös käyttää palvelimen joutilaana olevat resurssit tarvittaessa. Resurssien jakoa voidaan muokata vaikuttamatta käynnissä olevan virtuaaliympäristöön. (SWSOft 2008, 6.)

Virtuozzon hallinta tapahtuu joko komentoriviltä, graafisesta käyttöliittymästä (Virtuozzo Management Center) tai verkkosivulta toimivasta Virtuozzo Control Centeristä. Hallintatyökaluilla voidaan provisoida, seurata, varmuuskopioida, palauttaa, analysoida, etsiä virheitä, korjata ja siirtää virtuaaliympäristöjä. Näistä tavoista etenkin komentorivityökalulla voidaan tehdä ajastettuja ja automaattisia tehtäviä virtuaaliympäristöjen hallinnoimiseksi. Virtuaaliympäristöjen ylläpitäjät voivat käyttää selainpohjaista Virtuozzo Power Panelia omien virtuaaliympäristöjensä käynnistämiseen, pysäyttämiseen, varmistamiseen ja muihin tärkeisiin toimenpiteisiin ilman Virtuozzo pääkäyttäjää. (SWSOft 2008, 7.)

3.2.4 Ratkaisujen kustannukset ja vertailu tilanteeseen ilman virtualisointia

Laitteistovirtualisoinnin järjestelmien kilpailun kiristyessä ohjelmistovalmistajat ovat tuoneet tarjolle myös ilmaisia vaihtoehtoja tuotteistaan. Nämä tuotteet ovat usein rajoitettuja ominaisuuksiltaan, ja rajoitusten vuoksi suuremmat yritykset tarvitsevatkin kaupallisten tuotteiden ominaisuuksia.

Esitellyistä ratkaisuista sekä VMWare GSX että Microsoft Virtual Server toimivat käyttöjärjestelmän päällä, joten laitteistokustannusten lisäksi ratkaisun kustannuksia aiheuttaa myös käyttöjärjestelmälisenssit. VMWaren ohjelmisto on saatavissa myös Linux -käyttöjärjestelmälle, joten kustannuksissa on mahdollista säästää käyttöjärjestelmälisenssien suhteen, jos alustana käytetään ilmaista käyttöjärjestelmän jakeluversiota. Itse virtualisointijärjestelmät ovat saatavana maksuttomana ohjelmistovalmistajilta.

VMWare ESX on maksullinen ohjelmisto, joka tarvitsee toimiakseen sovellusvalmistajan hyväksymän laitteistokokoonpanon lisäksi sovelluslisenssit. Ratkaisun kustannuksia laskettaessa tulee huomioida myös tarve Virtual Centerille, joka käytännössä lisää kustannuksia yhden lisäpalvelimen ja käyttöjärjestelmälisenssin verran.

Virtuozzo ei ole varsinainen laitteistovirtualisointijärjestelmä, vaan järjestelmä luo virtuaaliympäristöt suoraan isäntäkäyttöjärjestelmästä. Virtuozzoa käytettäessä käyttöjärjestelmä virtuaalikoneissa on siis sama kuin isäntäkoneessa ja esimerkiksi uusien käyttöjärjestelmien testaaminen ei ole mahdollista. Järjestelmän kustannukset muodostuvat laitteiston ja käyttöjärjestelmän lisäksi Virtuozzon lisenssistä.

Laitteistojen virtualisoinnilla saavutetaan samoja etuja käytettävästä virtualisointijärjestelmästä riippumatta. Yleisiksi eduiksi voidaan katsoa järjestelmän vieraskäyttöjärjestelmälle luoma abstrakti laitteistokerros ja tämän ominaisuuden tuoma mahdollisuus siirtää virtualisoitu laite isäntäkoneelta toiselle ilman muutoksia asiakasjärjestelmään. Suurin etu tällä saavutetaan laitteiston ikääntyessä ja vikaantuessa, jolloin vastaavaa fyysistä laitetta ei enää ole saatavilla, eikä palvelun siirtäminen vikaantuneesta laitteesta uuteen ole mahdollista esim. käyttöjärjestelmä-

rajoitusten vuoksi. Siirrettävyyden lisäksi virtuaalisista palvelimista saadaan helposti kopio testausta varten joko kopioimalla palvelimen virtuaalikiintolevyt ja asetustiedostot tai käyttämällä virtualisointijärjestelmän valokuvaominaisuutta, jolloin itse virtualisointijärjestelmä säilyttää valokuvaushetken tilanteen palvelimesta. Koska palvelimia voidaan käsitellä kuin tiedostoja kiintolevyllä, voidaan luoda palvelimesta joko yleinen pohja tai jotain tiettyä toistettavaa palvelua varten esiasennettu virtuaalipalvelin, jota kopioidaan uuden palvelimen tai palvelun tarpeiden mukaisesti.

Kehittyneimmissä laitteistovirtualisointiratkaisuissa virtuaalipalvelimia voidaan siirtää käynnissä olevana keskeytyksettä isäntälaitteelta toiselle. Tällaisen ominaisuuden ansiosta virtuaalipalvelimille saadaan vikasietoisuutta, koska virtuaalipalvelimet voidaan siirtää isäntälaitteen vikaantuessa toiselle isäntälaitteelle, ilman ylläpitäjän toimenpiteitä.

3.3 Ohjelmistovirtualisointi

3.3.1 Altiris – Software Virtualization Solution

Altiris Software Virtualization Solution (lyhyemmin SVS) on Altiris:n ohjelmistovirtualisointijärjestelmä, jonka toiminta perustuu kerrosajatteluun. SVS:ia voidaan käyttää verkkoympäristön lisäksi myös yksittäisenä asennuksena, jolloin saadaan käyttöön ohjelmistovirtualisoinnin edut myös pienemmissä toteutuksissa kuten esim. kotona. SVS:ssa sovellukset ja sovellusten data sijoitetaan VSP:ihin (Virtual Software Package) ja näin ollen auttaa poistamaan sovelluksenhallinnan ongelmia. SVS:n avulla voidaan

- aktivoida tai deaktivoida sovelluksia
- palauttaa rikkoutunut sovellus toimivaan tilaan ilman vaikutuksia toisten sovellusten toimintaan (tai tarvittaessa luoda ajastettuja ohjelmapalautuksia)
- poistaa sovellusten välisiä epäyhteensopivuuksia (VSP sisältää esim. sovelluksen tarvitseman version DLL tiedostosta ja näin ollen sovelluksella on käytössään juuri tarvitsemansa versio)

- yksinkertaistaa sovellusten versionhallintaa.

(Altiris 2006, 1.)

SVS:n ydinkomponentti on Software Virtualization Agent. Agent täytyy olla asennettuna käyttäjän työasemaan sekä ns. base machineen. Base machine on työasema tai palvelin, joka ei ole tuotantokäytössä ja jossa on täysin päivitetty käyttöjärjestelmä ja hyvin vähän tai mieluummin ei mitään muita ohjelmia. Base machinea käytetään VSP:iden luomiseen asennetuista sovelluksista. Näin VSP:en saadaan kaikki sovelluksen käyttämiseen tarvittavat tiedostot, prosessit ja asetukset. VSP:n luonnin jälkeen SVS järjestää ne automaattisesti kerroksiksi (virtuaalisiksi sovelluksiksi). Järjestelmän ylläpitäjät voivat muokata kerroksissa olevien sovellusten asetuksia SVS:n hallintatyökaluilla. (Altiris 2006, 2.)

Sovelluksen virtualisointi voidaan tehdä joko yhden sovelluksen virtualisointina, globaalina muutosten kaappauksena, tai manuaalisesti tyhjästä VSP:sta. Yhden sovelluksen virtualisoinnin tapauksessa SVS:lle määritellään asennettavan sovelluksen asennusohjelman sijainti ja suoritetaan asennus. SVS tunnistaa asennusohjelman sulkeutumisen ja lopettaa muutosten tarkkailun automaattisesti. Globaalissa kaappauksessa sekä muutosten tarkkailun aloitus että lopetus tehdään manuaalisesti. Toinen merkittävä ero näillä kahdella menetelmällä on tarkkailtavien muutosten määrä ja vaikutusalue, joka yhden sovelluksen tapauksessa rajoittuu pelkästään asennusohjelman ja sen aliprosessien tekemiin muutoksiin, mutta globaalissa kaappauksessa kaikki järjestelmässä tapahtuvat muutokset kirjataan virtuaalisen sovelluksen muutoksiksi. Globaalista kaappauksesta on hyötyä tapauksissa, jossa yhteen VSP:en halutaan virtualisoida useampi sovellus, jotka ovat riippuvaisia toisistaan tai vaikuttavat suuresti toistensa toimintaan. Kolmas vaihtoehto on luoda VSP tyhjästä, jolloin halutut tiedostot kopioidaan manuaalisesti kerrokseen. Tätä tyhjän VSP:n tapausta voidaan käyttää esimerkiksi talon omien sovellusten, joista ei välttämättä ole olemassakaan asennustiedostoja, virtualisointiin. (Altiris 2006, 11 – 12.)

Jotta virtualisoituja sovelluksia voidaan käyttää muissakin tietokoneissa, täytyy niistä luoda arkistotiedosto siirtoa varten. SVS järjestelmässä tätä siirtotiedostoa kutsutaan VSA:ksi (Virtual Software Package Archive), joka on käytännössä ZIP

pakattu tiedosto. VSA:n sisältöä voidaan siis tarkastella ZIP tiedostomuotoa tukevilla pakkausohjelmilla. Arkistotiedoston luonnin jälkeen tiedosto siirretään asiakas-koneille käyttäen yrityksen tietojärjestelmään soveltuvinta tapaa. Tällainen menetelmä voi olla esimerkiksi sähköinen sovellustenjakelujärjestelmä, sähköposti tai CD-levy, riippuen käyttötarkoituksesta ja työntekijöiden teknisistä taidoista. Altiris tarjoaa jakeluun ohjelmistonhallintajärjestelmänsä Altiris Notification Serveriä, jonka avulla myös sovellusten päivitykset voidaan jakaa helposti ja keskitetysti. VSA:n jakelun jälkeen sovellus tuodaan SVS-järjestelmään jotain SVS:n hallintatyökalua käyttäen. Hallintatyökaluna voidaan käyttää joko graafista Software Virtualization Adminia, tai komentoriviltä käytettävää svscmd.exe:ä. Tuonnin yhteydessä SVS purkaa VSA:n uudeksi sovelluserrokseksi. Vielä ennen sovelluksen käyttöä uusi kerros on aktivoitava käyttöön. Aktivointi tapahtuu samoilla hallintatyökaluilla kuin muukin SVS:n hallinta. Sovelluksen käyttöön saattaminen ei siis vaadi sovelluksen uudelleenasettamista asiakasjärjestelmään. Kerroksen aktivoinnin jälkeen sovellusta voidaan käyttää aivan kuin sovellus olisi asennettu paikallisesti tietokoneelle. (Altiris 2006, 2 – 4.)

Software Virtualization Agentin toiminta perustuu Altiris:n kehittämään tiedostojärjestelmän suodatinajuriin, fslx.sys:iin. Tiedostojärjestelmän suodatinajuri sijaitsee tiedostojärjestelmän ja I/O (Input/Output) alijärjestelmän välissä, samalla tavoin kuin virustorjuntaohjelmistojen vastaava komponentti. Sijaintinsa ansiosta ajuri näkee jokaisen I/O alijärjestelmän kautta tulevan pyynnön, ja voi näin ollen kaapata ja uudelleenohjata pyynnön virtuaalisen sovelluksen tarpeen mukaisesti toiseen sijaintiin. (Altiris 2006, 5 – 6.)

Kerroksilla on viisi erilaista tilaa, jotka vaikuttavat fslx.sys:n tapaan käsitellä käyttöjärjestelmän lähettämiä pyyntöjä, ja ne on lueteltuna taulukossa 1.

TAULUKKO 1. Kerrosten tilojen vaikutus. (Altiris 2006, 7.)

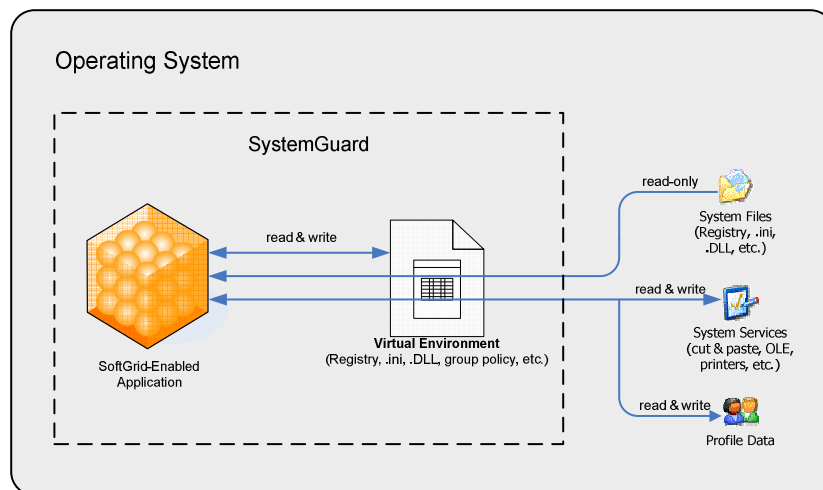
Tila	Määritelmä
Activated	Kerroksen sisältö on näkyvissä käyttöjärjestelmälle ja loppukäyttäjälle.
Deactivated	Kerroksen sisältö on näkyvissä vain fslx.sys ajurille.
Deleted	Kerros on merkitty poistettavaksi, mutta kerros poistetaan vasta, kun fslx.sys voi olla varma ettei kerroksen poistaminen vaikuta käynnissä oleviin prosesseihin.
Exported	Kerroksen sisältämät tiedot on pakattu VSA:ksi.
Imported	VSA:n sisältö on purettu ja kerros on valmis aktivoitavaksi.

Jokainen SVS:n kerros jakautuu kirjoitussuojattuun ja kirjoitettavaan alikerrokseen. Nimensä mukaisesti kirjoitussuojattu alikerros on suojattu käyttäjän tekemiltä muutoksilta, ja vain järjestelmän ylläpitäjät voivat muokata alikerroksen sisältöä. Käyttäjän tekemät muutokset tallennetaan kirjoitettavaan alikerrokseen ja näin sovellusten asetusten muokkaaminen on mahdollista. Kun kerros jaetaan näihin kahteen osaan, on sovelluksen palauttaminen alkuperäiseen tilaan helppoa. Käytännössä kirjoitettava alikerros korvataan uudella, tyhjällä kirjoitettavalla alikerroksella, ja kaikki sovellukseen tehdyt muutokset katoavat. Sovelluksen tiedostojen lisäksi nämä alikerrokset sisältävät myös sovelluksen merkinnät käyttöjärjestelmän rekisteriin. SVS:ssa sovelluksille voidaan määrittää lista tiedostoista, joita ei haluta tallentaa kirjoitettavaan alikerrokseen ja sitä kautta tuhota esim. sovelluksen korjauksen yhteydessä. Näin esim. Microsoft Wordin tilanteessa haluttaisiin poissulkea Word-dokumentit, jolloin ne tallentuvat todelliseen tiedostojärjestelmään virtuaalisen kirjoitettavan alikerroksen sijaan. (Altiris 2006, 7, 12.)

3.3.2 Microsoft SoftGrid

SoftGrid on Microsoftin ohjelmistovirtualisointijärjestelmä. SoftGridillä virtualisointi ohjelma luo virtualisointivaiheessa itselleen virtuaalisen SystemGuard ym-

päristön, joka estää ohjelmaa tekemästä muutoksia itse käyttöjärjestelmään ja rekisteriin. SystemGuard kuitenkin antaa ohjelmalle mahdollisuuden tallentaa omat rekisteriarvonsa ja muutoksensa ohjelman omaan virtuaaliseen ympäristöön, joka on eristetty todellisesta käyttöjärjestelmästä. Tätä menetelmää hyödyntäen ohjelma voi siis tehdä tarvitsemansa muutokset järjestelmään, jolloin ohjelman toiminta ei estetä, mutta SystemGuard kaappaa nämä muutokset ohjelman ja käyttöjärjestelmän välistä ja tekee ne virtuaaliseen ympäristöön aidon ympäristön sijaan. Virtuaalinen ympäristö pitää aidon käyttöjärjestelmäympäristön puhtaana virtuaalisten ohjelmien tekemiltä muutoksilta kuitenkin estämättä niitä toimimasta ja tallentamasta tietojaan. Jos virtuaalinen ohjelma ei pääsisi ollenkaan ulos eristysympäristöstään, jäisivät myös ohjelmilla luodut dokumentit ohjelman omaan hiekkalaatikkoon (kuvio 5).



KUVIO 5. SoftGrid SystemGuard ympäristö. (Sean ym. 2006, 23 – 24.)

Kirjoitushetkellä SoftGrid client on saatavilla seuraaville käyttöjärjestelmille:

- Windows 2000 Professional
- Windows 2000 Server ja Advanced Server
- Windows XP Professional
- Windows 2003 Server (Standard, Enterprise, Web ja Data Center Edition)
- Windows Vista.

SoftGrid for Terminal Servers on saatavilla seuraaville käyttöjärjestelmille:

- Windows 2000 Server ja Advanced Server
- Windows 2003 Server (Standard, Enterprise, Web ja Data Center Edition).

SoftGrid for Terminal Servers tarvitsee toimiakseen toimivan ja konfiguroidun terminaali palvelun Application Modeen. (Sean ym. 2006, 40; Microsoft 2008.)

SoftGrid -järjestelmä koostuu seuraavista osista:

SoftGrid Sequencer

SoftGrid Sequencer -työasema on SoftGrid-järjestelmän avainkomponentteja.

Sequencer on ohjattujen toimintojen avulla toimiva työkalu, joka uudelleenpake-toi ohjelman lähetettäväksi reaaliaikaisesti ja ajettavaksi eristetyssä ympäristössä. Sovellusta sekvensoitaessa, eli virtualisoitaessa, SoftGrid Sequencer luo sovelluk-selle neljä tiedostoa: SoftGrid-enabled sovelluksen tiedoston (SFT), OSD (Open Software Description) tiedoston, ikonitiedoston (ICO) ja projektitiedoston (SPRJ). Näistä tiedostoista SFT-tiedosto sisältää virtualisoidun sovelluksen datan, jonka Sequencer on pakannut lohkoihin suoratoistoa varten. SFT-tiedoston täytyy sijaita jokaisella SGVAS:llä (SoftGrid Virtual Application Server), joka kyseistä sovel-lusta jakelee. SPRJ-tiedostossa on tallennettuna sekvensointiprojektiin liittyviä asetuksia, kuten Sequencerin erikoisasetukset kyseiselle sovellukselle, ja viittaus jokaiseen sekvensoinnista syntyneeseen OSD-tiedostoon. ICO-tiedosto sisältää sovelluksen kuvakkeen datan, ja tämä kyseinen kuvake näytetään työpöydällä sovelluksen pikakuvakkeena. OSD-tiedosto sisältää tiedon ohjelman käynnistyk-sestä, kuten esimerkiksi vähimmäisvaatimukset käyttöjärjestelmän ja SoftGrid Clientin osalta sekä mahdolliset ennen tai jälkeen ohjelman suorituksen suoritet-tavat komentojonot eli skriptit. Näistä tiedostoista SPRJ ja OSD ovat XML-muotoisia tiedostoja, ja etenkin OSD tiedostot voivat tarvita muutoksia vielä vir-tualisoinnin jälkeenkin. (Sean ym. 2006, 61, 313.)

SGVAS

SoftGrid Virtual Application Server on koko SoftGrid-infrastruktuurin toiminnan kannalta kriittisin komponentti. SGVAS käsittelee käyttäjien pyyntöjä sovelluksen

dataan ja siirtää tämän datan suoratoistotekniikkaa käyttäen tunnistetuille ja hyväksytyille käyttäjille heidän tarpeidensa mukaisesti. SGVAS-palvelu lukee konfiguraatietonsa Data Storesta käynnistymisensä yhteydessä ja alkaa toimia konfiguraationsa mukaisesti. Käyttäjän ottaessa yhteyttä SGVAS:iin SGVAS vertaa SoftGrid Clientin lähettämää tunnistuslipukkeen sisältämää ryhmätietoa Data Storessa olevaan ryhmätietoon ohjelmien osalta ja tämän tiedon perusteella SGVAS tietää, mihin ohjelmiin pyynnön lähettäneellä käyttäjällä on käyttöoikeus. SGVAS lähettää tunnistetulle ja hyväksytylle käyttäjälle tämän käytössä oleviin ohjelmien OSD- ja ICO-tiedostojen sijainnin, josta SoftGrid client kopioi kyseiset tiedostot paikalliseen järjestelmään. Jokaisen ohjelmakäynnistyksen yhteydessä SoftGrid Client on yhteydessä SGVAS:iin, jolloin SGVAS varmistaa Data Storesta käyttäjällä olevan vielä oikeus käyttää kyseistä sovellusta, ja jos lisenssinhallinta on käytössä, onko vapaita lisenssejä vielä olemassa. SGVAS siirtää tarvittavat data-lohkot SoftGrid Clientille, jos käyttäjällä on edellisten tarkistusten perusteella mahdollisuus käynnistää sovellus. (Sean ym. 2006, 59.)

SoftGrid Management Console

SoftGrid Management Console on MMC:n (Microsoft Management Console) laajennus. Ohjelmaa järjestelmään tuotaessa Sequencerin luomat pikakuvake ym. määritelmät luetaan tiedostopalvelimelle kopioiduista tiedostoista, ja näitä esimääriteltyjä asetuksia voidaan vielä muokata ennen ohjelman käyttöönottoa. Management Console yhdistyy Management Web Serviceen käyttäen kommunikointiin .NET Remotingin avulla HTTP- (Hypertext Transfer Protocol) tai HTTPS (Hypertext Transfer Protocol Secure)-protokollia. Management Console voi sijaita joko samalla, tai eri tietokoneella kuin Management Web Service. (Sean ym. 2006, 37, 57.)

SoftGrid Management Web Service

SoftGrid Management Web Service toimii SoftGrid Management Consolen ja Data Storen välissä. Näin Management Console ei pääse suoraan lukemaan tai kirjoittamaan Data Storeen, vaan kaikki Data Storen käsittely tapahtuu Management Web Servicen välityksellä. Järjestelmän ylläpitäjän ottaessa yhteyttä Mana-

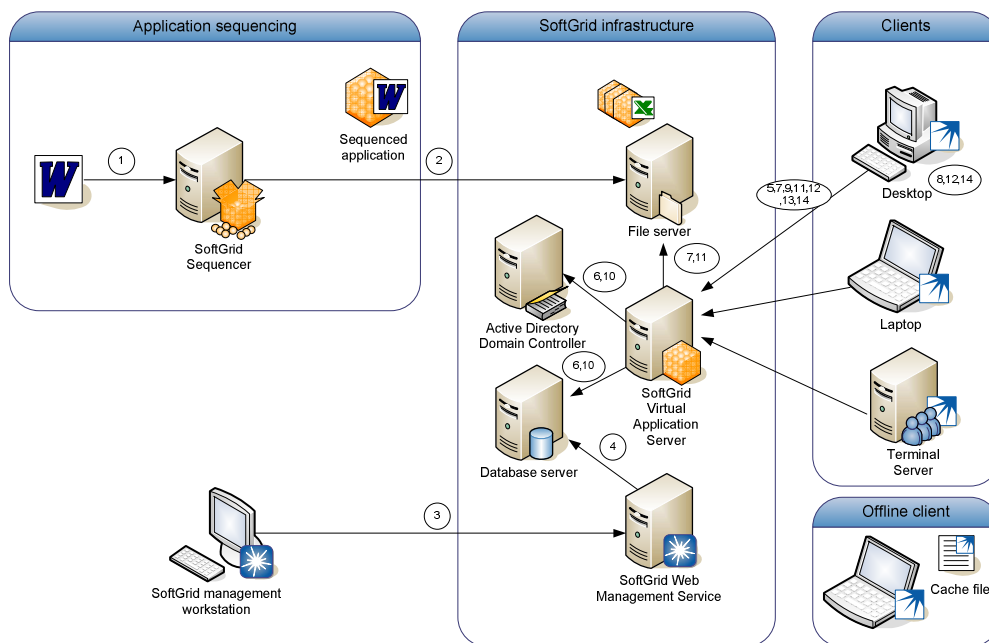
gement Consoleella Management Web Siteen, ylläpitäjän käyttäjätunnus ja oikeudet varmistetaan AD:n (Active Directory) käyttäjäryhmistä ADSI (Active Directory Service Interface) yhteydellä. (Sean ym. 2006, 57.)

Data Store

Data Store toimii SoftGrid infrastruktuurin tietovarastona, jonka tarkoituksena on tallentaa, seurata ja tarjota kaikki järjestelmän toiminnan kannalta tarvittava informaatio. Tämä informaatio sisältää Provider Policyt, palvelinryhmät, palvelimien konfiguraation ja Account Authority -tiedon. Data Store sisältää tiedot myös sovelluksista ja sovelluksiin liittyvistä käyttöoikeuksista, FTA:sta (File Type Association) ja lisensoinnista. Data Store voi myös tallentaa käyttötiedot seurantaan ja raportointia varten. Data Store voidaan asentaa joko Microsoft SQL 2000- tai 2005- tai MSDE (Microsoft SQL Server 2000 Desktop Engine)-alustalle. Yleisissä SoftGrid ympäristöissä samaa Data Storea käyttää yksi tai useampi SoftGrid Virtual Application Server. (Sean ym. 2006, 58 – 59.)

SoftGrid Client

SoftGrid Client on komponentti, joka hallitsee työasemassa sovellusten julkaisua käyttäjän näkyville, hallitsee yhteyksiä SGVAS:eihin sekä virtuaalisia SystemGuard-ympäristöjä. Käyttäjän kirjautuessa työasemaan standardia Windows autentikointia käyttäen, SoftGrid Client kaappaa Local Security alijärjestelmän luoman tunnistuslipukkeen, joka sisältää käyttäjän turvatunnisteen ja kaikkien ryhmien turvatunnisteet, joihin käyttäjä kuuluu. SoftGrid Client kommunikoi kahdella eri protokollalla riippuen suoritettavasta toiminnosta. SGVAS:n kanssa SoftGrid Client kommunikoi RTSP- (Real-Time Streaming Protocol) tai RTSPS (Real-Time Streaming Protocol Secure) -protokollalla. Yhteydenoton jälkeen yhteys jatkuu RTP- (Real-Time Transport Protocol) ja RTCP (Real-Time Transport Control Protocol) -protokollia hyödyntäen. (Sean ym. 2006, 59 – 60.)



KUVIO 6. SoftGrid ympäristön toiminta.

Kuviossa 6 on kuvattu SoftGrid ympäristön toiminta ja sen eri osien liittyminen toisiinsa. Alla on selitetty kuviossa esiintyvä sovelluksen matka asennuspaketista aina loppukäyttäjän työpöydälle asti.

1. Sovellus sekvensoidaan, eli virtualisoidaan SoftGrid Sequencerilla.

SoftGrid Sequencer on työasema, jolla sovelluksista luodaan virtuaalisia sovelluspaketteja. Sovellusta virtualisoidessa käydään läpi kolme päävaihetta, ja näiden vaiheiden lopputuloksena tavallisesta ohjelmasta on tullut virtualisoitu ohjelmapaketti. Sequencer tallentaa virtualisoidusta ohjelmasta tarvittavat tiedostot, joiden avulla sovellus saadaan tuotua SoftGrid järjestelmään.

2. Sekvensoitu ohjelmapaketti kopioidaan tiedostopalvelimelle.
3. Uusi ohjelma määritellään käyttöön tietyille käyttäjäryhmälle.

Ennen kun tiedostopalvelimelle kopioitua virtualisoitua sovellusta voidaan käyttää, on järjestelmän ylläpitäjän tuotava sovellus SoftGrid-järjestelmään käyttämällä SoftGrid Management Consolea. Tässä vaiheessa määritellään myös käyttöoikeudet ohjelmaan tietyille AD:n ryhmille.

Vain määriteltyihin ryhmiin kuuluvat käyttäjät voivat nähdä SoftGridin julkaisemat pikakuvakkeet ja käynnistää virtualisoidun sovelluksen.

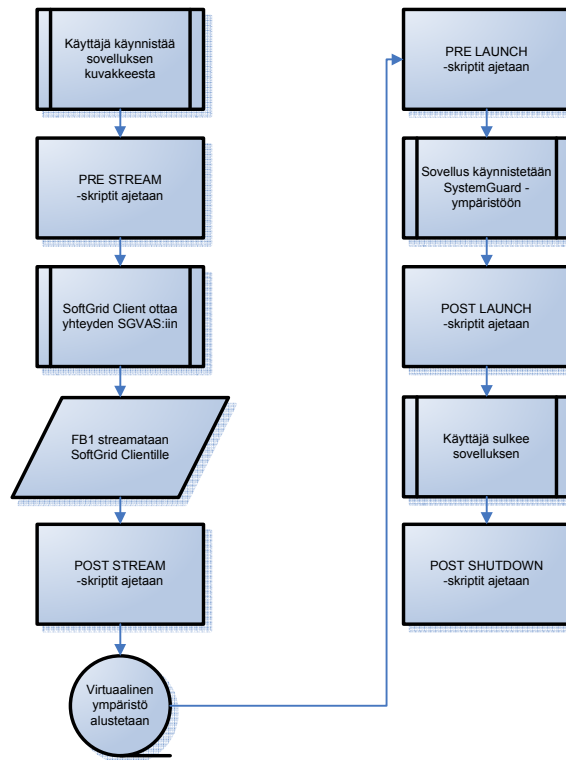
4. SoftGrid Web Management Service tallentaa konfiguraatietiedon tietokantaan.
5. SoftGrid Client pyytää palvelimelta listaa saatavilla olevista sovelluksista.
6. SGVAS vertaa työaseman lähettämää käyttäjätunnusta AD:n käyttäjäryhmiin ja tietokannassa oleviin ohjelmatietoihin.
7. SGVAS palauttaa käyttäjälle .OSD ja .ICO tiedostot kyseiselle käyttäjälle ja käyttöjärjestelmälle saatavilla olevista ohjelmista.
8. SoftGrid Client luo ja/tai päivittää pikakuvakkeet ja FTA:t työasemalla. Pikakuvakkeiden sijainnit ovat määritelty ohjelman OSD-tiedostossa.
9. Käyttäjä käynnistää sovelluksen työasemalla. Työasema lähettää pyynnön ohjelman käynnistyksessä SGVAS:lle.
10. SGVAS vertaa työaseman lähettämiä käyttäjätietoja kyseiseen sovelluksen käyttöoikeuksiin ja AD:n käyttäjäryhmiin.
11. SGVAS käyttää suoratoistotekniikkaa siirtääkseen ohjelman käynnistykseen tarvittavat ohjelmakomponentit työasemalle käyttäen RTSP-protokollaa.
12. Ohjelma käynnistyy työasemalla, ja SoftGrid Client lähettää tiedon ohjelman käynnistyksessä palvelimelle ja tallentaa tapahtuman omaan lokiinsa. Ohjelman paikallisesti tallennetut muutokset ladataan ohjelman omaan virtuaaliympäristöön.
13. SoftGrid Client pyytää tarvittaessa lisää sovelluksen osia SGVAS:lta, joka siirtää ne clientille suoratoistotekniikalla.
14. Ohjelma suljetaan, tiedot ohjelman sulkemisesta tallennetaan lokiin ja lähetetään SGVAS:lle. Ohjelman virtuaaliympäristö tallennetaan muutokseen paikallisen tietokoneen kiintolevyille ohjelman seuraavaa käynnistystä varten. Clientin välimuistissa olevat tiedot säilyvät offline –käyttöä ja seuraavaa käynnistystä varten.

(Sean ym. 2006, 36 – 45.)

Virtualisoitujen sovellusten päivittämiseen SoftGrid 4.0 tuo uutena ominaisuutena Active Upgrade -ominaisuuden. Ennen tätä uutta ominaisuutta, ylläpitäjän tuli varmistua, ettei yksikään käyttäjä käyttänyt sovellusta, kun sovellus päivitettiin.

Nyt virtuaalinen sovellus voidaan päivittää palvelimelle häiritsemättä käyttäjiä. Palvelin lisää saman ohjelmistopakettin tietoihin uuden versionumeron ja uuden SFT-tiedoston sijainnin. Näin auki olevat sovellukset voivat jatkaa toimintaansa keskeytymättä ja käyttää vanhan sovellusversion tiedostoja. SoftGrid Clientin hakiessa palvelimelta uusia sovellusten tietoja uusi sovelluksen versionumero ja SFT-tiedoston sijainti lähetetään Clientille. Seuraava sovelluksen käynnistyskerta käynnistää uudesta paketista päivitetyn sovelluksen. (Sean ym. 2006, 177.)

OSD-tiedostoja muokkaamalla SoftGrid:lle voidaan määrittää parametreja sovelluksen käynnistämistä tai virtuaaliympäristö alustamista varten. OSD-tiedostoon voidaan myös kirjoittaa sovelluksen toiminnan tai vian selvityksen kannalta tärkeitä skriptejä. Skriptejä voidaan ajaa eri kohdissa sovelluksen käynnistämistä (visualisoitu kuviossa 7). Usean käynnistyskohdan lisäksi voidaan määrittellä, halutaanko skripti suorittaa suojatussa tilassa SystemGuard-ympäristössä, vai sen ulkopuolella. Tällaisilla OSD -tiedoston muutoksilla voidaan esim. yhdistää sovelluksen kannalta välttämättömät verkkolevyt tai käynnistää komentokehote sovelluksen vianselvityksen auttamiseksi.



KUVIO 7. Skriptien suoritusjärjestys. (Sean ym. 2006, 313 – 316.)

3.3.3 Citrix Application Streaming

Citrix Application Streaming on osa Citrix Presentation Server -arkkitehtuuria. Application Streaming mahdollistaa ohjelmien ajamisen loppukäyttäjän työasemalla ilman ohjelman asentamista. Ajettava sovellus siirretään työasemalle tai palvelimelle suoratoistotekniikan avulla, ohjelmakomponentti kerrallaan ja tarpeen mukaan. Suoratoistettu ohjelma ajetaan eristyksistä muista sovelluksista ja käyttäjärjestelmästä Citrixin Isolation Environment -ympäristössä, ja tämän ominaisuuden ansiosta ohjelma ei pääse häiritsemään käyttäjärjestelmän tai muiden sovellusten toimintaa. (Citrix Systems Inc. 2007, 729.)

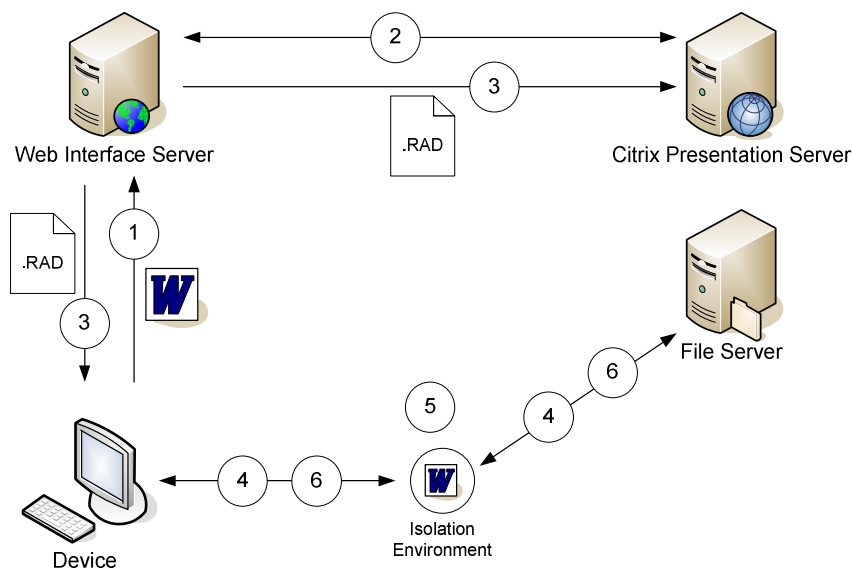
Citrix Application Streaming -ympäristö koostuu seuraavista komponenteista tavanomaisen Citrix Presentation Server ympäristön lisäksi:

- Citrix Streaming Profiler
- Citrix Streaming Client

Näiden komponenttien lisäksi Application Streaming lisenssien tulee olla asennettuna lisenssipalvelimelle. (Citrix Systems Inc. 2007, 731 – 732.)

Citrix Streaming Profilerilla luodaan suoratoistettavien ohjelmien profiili. Järjestelmän ylläpitäjä luo haluamastaan sovelluksesta, tai sovellusryhmästä paketin käyttäen Citrix Streaming Profileria ja määrittelee profiilin, jonka käyttäjärjestelmä, service pack -taso, levyasematunnukset ja kieli vastaavat asiakaslaitteiden järjestelmätietoja. Jos sovellusta on jaettava useammalle erilaiselle alustalle, voidaan sovelluksesta tehdä useampia profiileja erilaisilla järjestelmätiedoilla. Profiiloinnin jälkeen sovelluksia voidaan päivittää ja resursseja lisätä profiiliin tarpeen mukaan. Itse Citrix Streaming Profiler tulee asentaa samalle käyttäjärjestelmälle ja service pack -tasolle, kuin ympäristön vanhin käytössä oleva asiakaslaite. Yhteensopivuuden varmistamiseksi jokaista käytössä olevaa käyttäjärjestelmää varten tulisi olla erillinen Streaming Profiler. Tämän lisäksi Streaming Profileriin ei tule asentaa muuta kuin järjestelmän toiminnan kannalta välttämättömät sovellukset. (Citrix Systems Inc. 2007, 731 – 740.)

Citrix Streaming Client paikallistaa suoratoistettavan sovelluksen, valmistelee eristysympäristön sovellukselle ja siirtää tarvittavat sovelluksen osat. Streaming Client on pakollinen osa Citrix Application Streaming -järjestelmää ja Streaming Client asentuu oletuksena kaikille Citrix Presentation Server 4.5 -palvelimille, joiden lisäksi Streaming Client voidaan asentaa myös Windows 2000, XP, Vista ja Windows Server 2003 käyttäjärjestelmiin. (Citrix Systems Inc. 2007, 731.)



KUVIO 8. Citrix Application Streaming kommunikointi.

Citrix Application Streamingia käytettäessä sovelluksen käynnistys etenee kuvion 8 mukaisesti:

1. Käyttäjä käynnistää sovelluksen käyttäen Program Neighborhood Agentia tai Web Interface -sivustoa.
2. Sovelluksen käynnistyspyyntö välitetään Web Interfacelta Presentation Serverille. Web Interface luo .RAD tiedoston, josta selviää sovelluksen sijainti tiedostopalvelimella.
3. Web Interface lähettää .RAD tiedoston Citrix Streaming Clientille
4. Citrix Streaming Client käynnistää eristysympäristön ja suoratoistaa sovelluksen .RAD tiedoston osoittamasta sijainnista .
5. Eristysympäristö muokataan sovellusprofiilin mukaiseksi, ja sovellus käynnistetään eristysympäristön sisälle.
6. Citrix Streaming Client lataa tarvittaessa sovelluksen tarvitsemia tiedostoja.

(Citrix Systems Inc. 2007, 733 – 734.)

Uutta sovellusta profiloidessa käynnistetään Streaming Profiler ja luodaan uusi profiili. Tähän uuteen profiiliin asennetaan virtualisoitava sovellus kertomalla

Streaming Profilerille käynnistettävän sovelluksen asennusohjelma. Streaming Profiler käynnistää asennusohjelman eristysympäristön sisälle ja tallentaa kaikki asennusohjelman järjestelmään tekemät muutokset. Luotuun profiiliin tallennetaan tiedostojärjestelmään ja rekisteriin tehdyt muutokset sekä sovelluksen eristysympäristön ja profiilin tiedot. Profiili tallennetaan tiedostopalvelimelle, josta profiili voidaan julkaista sovelluksen loppukäyttäjille. (Citrix Systems Inc. 2007, 741.)

Sovelluksen päivittäminen profiiliin tapahtuu Streaming Profilerin avulla. Päivityksen yhteydessä paketin sisäistä sovelluksen versionumeroa kasvatetaan, ja päivitettyt kohteet tallennetaan uusiksi tiedostoiksi profiiliin. Useamman version tallentaminen samaan profiiliin mahdollistaa sovelluksen vanhan version käyttämisen, ja tämän vuoksi sovelluksien käyttöön ei tule käyttökatkoa päivityksen yhteydessä. Uusi sovelluksen käynnistyspyyntö ohjataan kuitenkin uuteen tiedostoversioon, ja näin vanha versio jää automaattisesti pois käytöstä. Vanha sovellusversio voidaan poistaa, kun sovellusta ei enää käytetä. Sovellusversion poistaminen tapahtuu poistamalla sovellusversion .CAB tiedosto tiedostopalvelimelta. (Citrix Systems Inc. 2007, 778 – 780.)

Sovellukset julkaistaan loppukäyttäjille Citrix Management Console - hallintasovelluksen avulla. Sovellusta julkaistaessa määritellään, kuinka sovellusta käytetään: ajetaanko sitä palvelimella vai käyttäjän työasemalla. Sovellusta julkaistaessa valitaan myös sovelluksen käyttäjät, eli käytännössä käyttäjäryhmä, jossa olevat käyttäjät saavat käynnistää sovelluksen. Pikakuvakkeen kuvake, nimi ja sijainti määritellään keskitetysti hallintasovelluksella. Program Neighborhood Agentia käytettäessä sovelluksen pikakuvake voi sijaita Windowsin käynnistä - valikossa. Sovellusta julkaistaessa voidaan määritellä myös, onko sovellus käytettävissä ilman verkkoyhteyttä, ja jos on, niin ketkä voivat käyttää sovellusta näin. (Citrix Systems Inc. 2007, 786 – 791.)

3.3.4 Ratkaisujen kustannukset ja vertailu tilanteeseen ilman virtualisointia

Ohjelmistovirtualisoinnin kustannukset muodostuvat ympäristön tarvitsemasta yhdestä tai useammasta palvelimesta sekä työasemasta, jolla varsinainen sovel-

luksen virtualisointi tehdään. Etenkin virtualisointityöaseman kohdalla kannattaa hyödyntää laitteistovirtualisointia, jos suinkin mahdollista, koska työasema on palautettava puhtaaseen tilaan virtualisoinnin jälkeen. Palvelinlaitteiden lisäksi palvelimiin tarvitaan käyttöjärjestelmälisenssit ja mahdolliset virtualisointijärjestelmien lisenssit. SoftGrid:n osalta virtualisointiin tarvittavan infrastruktuurin sovelluslisenssit kuuluvat SoftGrid asiakasohjelmalisenssien hintaan, kun Citrix Application Streaming tarvitsee Citrix infrastruktuurin lisensoinnin lisäksi erilliset Citrix Application Streaming lisenssit. Lisenssien lisäksi virtualisointi aiheuttaa kustannuksia myös henkilöstön koulutuksen osalta. Loppukäyttäjille sovellusvirtualisoinnin aiheuttamat muutokset ovat hyvin kosmeettisia, ja näin ollen koulutustarve onkin pelkästään ylläpitohenkilöstöllä.

Sovelluksen virtualisoinnin aiheuttama työmäärä on yksittäisen sovelluksen työmäärää suurempi, koska sovellusta asennettaessa tulee huomioida sovelluksen toiminta ja vaatimukset ulkoisten ohjelmakomponenttien suhteen. Tämä lisää työn kustannusten osuutta, mutta koska asennustyötä ei tarvitse toistaa jokaisen yksittäisen asennuksen myötä, tulevat virtualisoinnin säästöt esille asennuskustannuksissa muutaman saman sovelluksen käyttöönoton tapauksessa.

Asennuksen ja asennustapahtumien toiston vähenemisen lisäksi ohjelmistovirtualisoinnin avulla voidaan ratkaista yhteensopivuusongelmia, joissa jotkin kaksi sovellusta eivät voi toimia tai olla asennettuna samanaikaisesti yhdelle tietokoneelle. Virtuaaliset eristysympäristöt estävät sovelluksia näkemästä toisiaan ja tarjoavat sovelluksille juuri niiden tarvitsemat ohjelmistokirjastot, ja näin ollen käyttöjärjestelmän tai toisen sovelluksen päivittäminen ei vaikuta virtualisoidun sovelluksen toimintaan. Koska molemmat SoftGrid ja Citrix Application Streaming sisältävät sovelluksen jakelutavan, voidaan myös sovellusten päivittäminen tehdä keskitetysti ja näin ollen jaella testattu ja toimiva päivitetty sovellus loppukäyttäjille. Uuden version käyttöönotto tapahtuu automaattisesti sovelluksen seuraavalla käynnistyskerralla. Ylläpitäjien kannalta versionhallinta helpottuu, eikä loppukäyttäjien tarvitse kärsiä sovellusversion päivityksen aiheuttamasta katkoksesta sovelluksen tai tietokoneen käytössä.

4 NYKYISET VIRTUALISOINTIRATKAISUT

4.1 Liiketoimintaa edistävät ratkaisut

Kaikki työssä esitellyt ratkaisut ovat suunniteltuja edistämään liiketoimintaa, ja näin ollen ne voivatkin lisätä tuottavuutta vähentämällä ylläpitokustannuksia sekä nopeuttamalla sovellusten jakelua ja hallintaa. Etäkäyttöratkaisut tarjoavat mahdollisuuden käyttää yrityksen sovelluksia ja tietokantayhteyksiä toimittaessa yrityksen verkon ulkopuolelta. Etäkäytöllä voidaan myös käyttää sovelluksia, joita ei ole mahdollista tai järkevää asentaa paikallisesti käyttäjän tietokoneille tai joiden virtualisointi sovellustasolla ei ole mahdollista tai järkevää.

Laitteistovirtualisointia käytettäessä voidaan hyödyntää valmiita käyttöjärjestelmäpohjia eli esim. tiettyä palvelinsovellusta varten asennettua järjestelmää, josta luodaan kopio asetuksineen ja tietoineen uudeksi palvelimeksi. Tällä tavoin saadaan säästettyä huomattavasti aikaa uusien järjestelmiä rakennettaessa, koska itse peruskäyttöjärjestelmän rutiininomainen asennus ja asennuksen tarvitsema aika jää pois.

Laitteistovirtualisoinnilla voidaan säästää myös esim. sovellustestauksessa tai ongelmanratkaisutilanteissa. Sovellustestausympäristöä varten ei tarvitse hankkia useita palvelimia eikä näin ollen niiden tarvitsema tila tai sähkönsyöttö ole ongelmallista. Sekä testauksessa että ongelmanratkaisussa on usein hyödyllistä päästä takaisin tiettyyn tilanteeseen ja yrittää toista ratkaisua tai asetusta. Virtuaalisten työasemien tai palvelinten kanssa koko käyttöjärjestelmästä voidaan ottaa tiedototatasolla kopio, joka voidaan palauttaa tarvittaessa entisen päälle tai hyödyntää useimmissa laitteistovirtualisointijärjestelmissä olevaa valokuvaominaisuutta, joka mahdollistaa tallennettuun tilanteeseen palaamisen.

4.2 Yleishyödylliset ratkaisut

Jokapäiväisessä tietokoneen käytössä, yritysmaailman ulkopuolella ohjelmistovirtualisointi tai laitteistovirtualisointi eivät toistaiseksi tarjoa peruskäyttäjälle lisäar-

voa. Harrastajat ja opiskelijat puolestaan voivat hyötyä laitteistovirtualisoinnin tuomasta mahdollisuudesta testata esim. uusia käyttöjärjestelmä versioita tai, käyttää kokonaan toista käyttöjärjestelmää ja sen mukanaan tuomia sovelluksia ja palveluita samanaikaisesti yhdellä fyysisellä tietokoneella.

ASP-tekniikalla toteutetut ratkaisut, kuten esim. selaimella toimiva sähköposti, tuovat palvelut loppukäyttäjien käyttöön käytettävästä päätelaitteesta riippumatta. Tällaista selainpohjaista palvelua käytettäessä palvelun käyttäjällä ei tarvitse olla tavallista käyttöoikeutta enempää oikeuksia työasemaan, koska työasemalle ei tarvitse asentaa lisäkomponentteja palvelua varten. Tunnettuna esimerkkinä tällaisesta sähköpostista voidaan mainita Microsoftin Hotmail.

4.3 Viihdekäyttö

Toistaiseksi ohjelmistovirtualisointia ei ole hyödynnetty viihdekäytössä, mutta virtualisoinnin avulla olisi mahdollista toteuttaa kuukausimaksuperustainen palvelu, jonka asiakassovellus jaeltaisiin suoratoistotekniikkaa hyödyntäen ja virtualisoitaisiin työasemassa. Tällä tavalla virtualisoidun sovelluksen sovellustiedostoihin pääsy olisi hankalampaa sovelluksen muokkaamisen kannalta (esim. tietokonepelit), ja sovelluksen käynnistäminen voitaisiin estää kokonaan, jos palvelun tilaus on lopetettu. Lopetettaessa palvelun käyttö, voitaisiin myös poistaa sovelluksen paikallisessa välimuistissa olevat komponentit ilman käyttöjärjestelmän tai muiden komponenttien rikkoutumisen vaaraa.

ASP mallilla voidaan toteuttaa viihdekäyttöä ajatellen esim. pikaviestinpalveluita, kuten MSN Messenger. Selaimella toimivan pikaviestinpalvelun käyttöä varten ei enää tarvita erikseen asennettavaa sovellusta, eikä palvelu myöskään ole käyttöjärjestelmäriippuvainen.

5 TAPAUS DNA OY

5.1 Yleinen kuvaus toteutuksesta

DNA Oy / Etelä-Suomen tietoverkossa on muutamia satoja työasemia, joista hie-
man alle puolet on kannettavia tietokoneita. Työasemia käytetään pääsääntöisesti
työnantajan tiloissa omilla työpisteillä, mutta myös muualla tapahtuvaa työtä teh-
dään jonkin verran. Tällainen ympäristö asettaa vaatimuksia ja luo haasteita tieto-
järjestelmiä ylläpitävälle organisaatiolle, sekä itse tietojärjestelmille. Työasemat
ovat pääsääntöisesti henkilökohtaisia, mutta joitain työasemia käyttää tietty ryhmä
erillisillä yhteiskäyttötunnuksilla.

Henkilöstön tietokoneilla käytössä on vaihteleva lukumäärä erinäisiä työnteolle
välttämättömiä ohjelmia, joiden asennus, ylläpito ja hallinta ovat haasteellinen
tehtävä yrityksen tietojärjestelmien ylläpidolle.

Koska yrityksen tietoverkko ja tietojärjestelmät ovat olleet käytössä jo vuosia en-
nen uusien tekniikoiden käyttöönottoa, eikä varsinaisen tuotantoympäristön rin-
nalle ole taloudellisesti tai teknisesti järkevää rakentaa toista yhtä suurta infra-
struktuuria pelkästään uusien tekniikoiden käyttöönoton vuoksi, on tekniikoiden
käyttöönotto tehty porrastetusti ja täydentämään olemassa olevaa tietojärjestel-
mää. Itse ohjelmien virtualisointi tehdään tarvelähtöisesti, jolloin tarpeettomien
ohjelmien virtualisointityöltä vältytään ja näin ollen yrityksen IT-
henkilöstöresurssit voidaan kohdistaa muihin projekteihin. Vaikka tarvelähtöinen
virtualisointi vähentää itse virtualisointityön tarvetta, hidastaa tarvelähtöisyys
myös uusien virtualisoitavien ohjelmien käyttöönottoa.

Microsoft SoftGrid:n toiminnallinen testaus tehtiin laboratorioympäristössä, jossa
käytössä oli kolme virtuaalista palvelinta ja kaksi virtuaalista työasemaa. Virtu-
alisointiympäristönä käytettiin olemassa olevaa Microsoft Virtual Server 2005 -
alustaa. Laboratorion palvelimien käyttöjärjestelmänä oli Windows Server 2003 ja
työasemien Windows XP. Palvelimista yksi toimi domain controllerina laborato-
rion AD:lle, toinen Microsoft Terminal Server etäkäyttöpalvelimenä ja kolmas
SoftGrid Virtual Application Serverinä. Työasemista toinen toimi virtualisoinnin

ja virtualisoitujen sovellusten testialustana ja toinen SoftGrid Sequencer työasemana. SoftGrid –järjestelmän tarvitsema tietokanta asennettiin SoftGrid Virtual Application Serverille asennettuun Microsoft SQL Server Desktop Engine tietokantapalveluun, koska testiympäristöstä ei haluttu mahdollistaa pääsyä tuotantojärjestelmään.

Järjestelmän testivaiheessa virtualisoitiin onnistuneesti muutamia työasemalla käytettäviä perussovelluksia ja useampi versio Java virtuaalikoneesta. Testatuista sovelluksista etenkin Javan virtualisoinnin helppous ja useamman Java version samanaikainen suorittaminen eri selainikkunoissa antoivat järjestelmästä vakuuttavan kuvan. Erinäisten testien perusteella SoftGrid vaikutti hyvin kiinnostavalta järjestelmältä, jolla oli täysi mahdollisuus ratkaista käsillä oleva ongelma.

Ohjelmistovirtualisointijärjestelmäksi valittiinkin siis Microsoftin SoftGrid. Valintapäätökseen vaikuttivat yrityksen positiiviset kokemukset ja asenteet Microsoftin järjestelmiä ja tukipalveluita kohtaan, lisenssien kohtuullinen hinnoittelu sekä tietojärjestelmän painottuminen melkein täysin Microsoftin tuotteilla toteutetuksi. Tällaisessa tapauksessa tuen saaminen ongelmatilanteisiin olisi luonnollisesti helppoa, koska yksi yritys toimittaa suuren osan ohjelmistoista, eikä yrityksen tarvitse käyttää omia IT-henkilöstöresurssejaan eri järjestelmien yhteensopivuusongelmien selvittämiseen järjestelmiä tuottavien osapuolien välille, vaan useimmiten voidaan toimia vain yhden ohjelmatoimittajan kanssa. Ohjelmistovirtualisointi on myös kustannustehokkaampi tapa päästä yhteensopivuusongelmien aiheuttamista haitoista kuin etäkäyttö. Etäkäytön lisenssikustannukset nousisivat huomattavasti ohjelmistovirtualisoinnin lisenssikustannuksia suuremmiksi etenkin Citrixin tapauksessa, jolloin tarvittaisiin sekä Terminal Server-, että Citrix-lisenssit.

Ohjelmistovirtualisointijärjestelmän suunnittelussa huomioitiin järjestelmän skaalautuvuus nykyistä suuremmalle käyttäjämäärälle ja laajempaan ohjelmistokirjoon. Järjestelmään suunniteltiin heti alusta mahdollisuus toteuttaa vikasietoisuutta ja kuormantasausta käyttämällä ohjelmapakettien ja asiakaslaitteiden konfiguraatiossa palvelimen nimen sijaan yleistä alias-nimeä, joka voidaan myöhemmin muuttaa osoittamaan kuormantasaajaan ja näin ollen saada järjestelmään sekä vi-

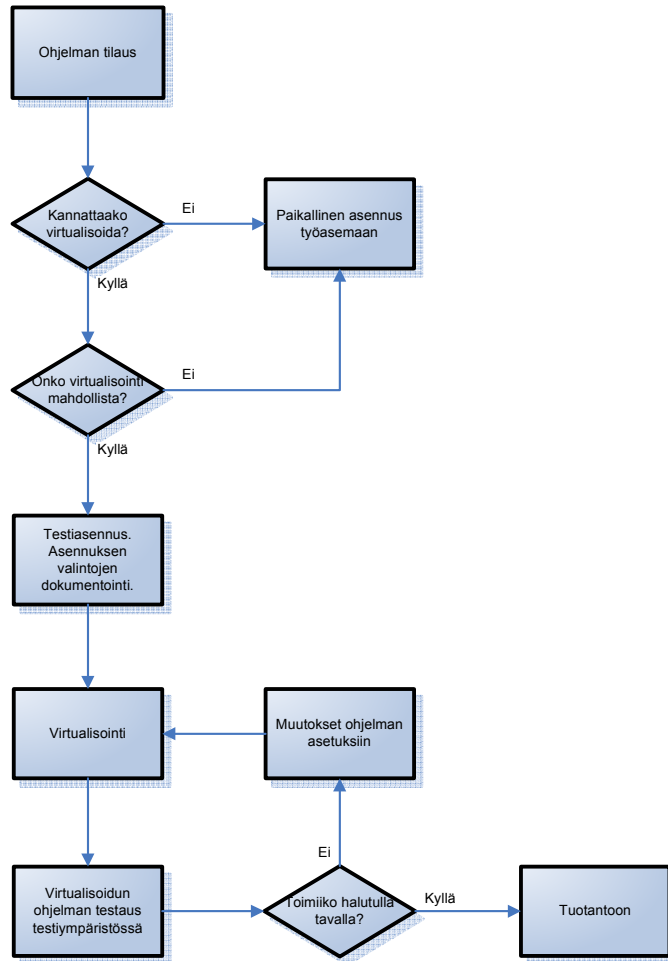
kasietoisuutta että jakaa kuormaa eri palvelimien kesken. Toteutusvaiheessa kyseisille toiminnoille ei kuitenkaan ollut tarvetta, ja ohjelmistovirtualisointipalvelin toteutettiin yhdellä fyysisellä palvelinlaitteella, kuitenkin huomioiden mahdollinen tuleva tehon ja muistin tarve. Kuorman tasauksessa on mahdollista käyttää joko erillistä kuormantasainta, tai Microsoft Windowsin omaa NLB (Network Load Balancing) tekniikkaa, joka mahdollistaa tässä tapauksessa saman toiminnan kuin erillinen kuormantasainlaite.

Varsinaiseksi SoftGrid VAS -palvelinlaitteeksi valittiin testikäytössä ollut blade -palvelin. Palvelimen käyttöjärjestelmäksi asennettiin Windows Server 2003. Palvelinohjelmiston käyttämäksi tietokannaksi valittiin yrityksen tietokantapalvelimelle luotu tietokanta. Näin ratkaisun tietokannan varmuuskopiointi ja ylläpito hoituu keskitetysti muiden tietokantojen yhteydessä. Tietojen tallennusvarastoksi levyjärjestelmään luotiin tallennusalue, joka jaettiin verkkoon palvelimen käyttöön. Tällä tavoin mahdollistettiin useamman SoftGrid VAS:n liittäminen samoihin virtuaalisten sovellusten tietoihin. Järjestelmän toimintavarmuutta voitaisiin lisätä kahdentamalla kaikki järjestelmän komponentit ja hajauttamalla järjestelmän toinen puoli toiseen konesaliin, mutta kustannusten kannalta tämä ei ollut vielä perusteltua.

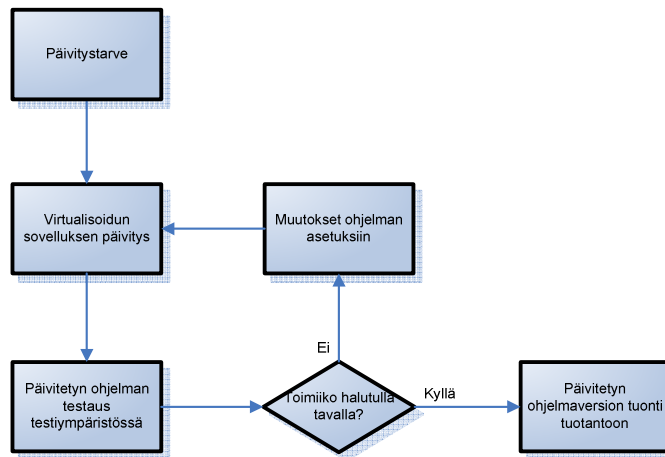
Työn aikana SoftGrid palvelimen ja asiakasohjelmistojen versiot päivittyivät, ja näin ollen myös järjestelmä päivitettiin. Työasemien osalta päivityksen asennus tehtiin samalla tavalla kuin ensimmäinen asennus. Palvelimenkaan osalta päivityksessä ei ollut ongelmia, vaan koko päivitysprosessi sujui asennusohjeen mukaisesti. Palvelimen päivitys tehtiin toimistoaikana, koska palvelua ei vielä nähty kriittiseksi liiketoiminnan kannalta.

Uuden sovelluksen virtualisointiprosessi ja virtualisoidun sovelluksen päivitysprosessi on suunniteltu etenemään kuvioissa 9 ja 10 kuvatulla tavalla. Testaamalla sovellusta tai sovelluspäivitystä erillisessä testiympäristössä vältetään virheellisesti virtualisoitujen tai päivitettyjen sovellusten mahdollisesti aiheuttamat ongelmatilanteet. Ongelmatilanne voi syntyä esim. kun sovelluksen luullaan olevan päivitetty onnistuneesti SoftGrid Sequencerilla, ja sovelluspaketti on tuotu päivityksenä järjestelmään, mutta sovellus ei todellisuudessa toimikaan odotetulla tavalla.

Näin loppukäyttäjät voivat ladata tietokoneilleen toimimattoman päivitysversion sovelluksesta, ja työnteko voi estyä.



KUVIO 9. Uuden virtuaalisen ohjelman tilausprosessi.



KUVIO 10. Virtuaalisen ohjelman päivitystilausprosessi.

SoftGrid-asiakasohjelmien asennus yrityksen työasemille toteutettiin uusien työasemien osalta lisäämällä asiakasohjelmisto vakioidun työaseman asennuspakettiin. Vanhojen jo käytössä olevien työasemien kohdalla asiakasohjelman jakelu tehtiin käyttäen yrityksessä jo käytössä olevaa ESD-järjestelmää, Microsoft SMS:ää (Systems Management Server). Asiakasohjelman asetukset määriteltiin asennuksen yhteydessä komentoriviltä, koska virallista AD:n kautta jaeltavaa asetuspohjaa ei ollut saatavilla. Asiakasohjelmiston käyttämäksi levyasematunnukseksi valittiin V, jotta muutoksia jo käytössä oleviin levyasematunnuksiin ei tarvitsisi tehdä. Välimuistin määrää kasvatettiin 2GB:stä 4GB:iin.

SoftGrid järjestelmän hyötyjen saattamiseksi laajempaan käyttöön, hankittiin työasemalicenssien lisäksi SoftGrid for Terminal Servers -lisenssi. Näin ohjelmistovirtualisoinnin etuja voidaan käyttää hyväksi myös yrityksen etätyöjärjestelmää käytettäessä. Lisenssi hankittiin varsinaisen käyttöönoton jälkeen, ja virtualisoidut, etäkäytössä tarvittavat sovellukset, muokattiin ja testattiin toimivaksi etäkäyttöympäristössä.

Laitteistovirtualisoinnin osalta yrityksessä on ollut käytössä Microsoft Virtual Server 2005. Palvelimien virtualisoinnista on ollut hyviä kokemuksia, mutta edellinen virtualisointiympäristö ei enää pystynyt tarjoamaan kaikkia haluttuja omi-

naisuuksia, ja esim. VLAN:ien (Virtual LAN) käyttö virtuaalisten palvelimien osalta ei ollut mahdollista. Kyseinen ongelma johtui palvelimen ajurien ja Microsoft Virtual Serverin ohjelmakomponentin sijainnista samalla tasolla ja näin ollen estäen ajurin VLAN-toiminnallisuuden.

Laitteistovirtualisointia varten valittiin VMWare ESX -järjestelmä sen tarjoamien teknisten etujen vuoksi verrattuna muihin virtualisointijärjestelmiin. Ratkaisevat edut olivat järjestelmän VLAN-tuki, mahdollisuus ottaa useita valokuvia virtuaalilaitteista ja järjestelmän VMotion -ominaisuus, joka tarjoaa vikasietoisuutta virtualisoituihin palvelimiin.

VMWare ESX -järjestelmää varten hankittiin ensimmäisessä vaiheessa yksi palvelin. Palvelimen paikallisia kiintolevyjä käytetään vain itse VMWare ESX:n tallentamiseen, ja kaikki virtuaaliset palvelimet sijaitsevat verkkotallennusjärjestelmässä, josta ESX-järjestelmä käyttää niitä NFS (Network File System) -tiedostojärjestelmän avulla. Virtuaaliympäristön hallintaa varten erilliseen palvelimeen asennettiin VMWare Virtual Center. Tähän tehtävään yrityksestä löytyi vapaana oleva palvelinlaite, joten uutta laitetta ei ollut tarpeellista tilata.

Toisessa vaiheessa järjestelmään lisättiin toinen palvelin tasaamaan järjestelmän kuormaa normaalitilanteessa sekä tuomaan vikasietoisuutta laitteistovian sattuessa. ESX-palvelimista luotiin High Availability-klusteri, ja kuorman tasaus säädettiin automaattiseksi. Näin järjestelmän annettiin itse päättää, milloin jokin virtuaalikone on järkevää siirtää fyysiseltä palvelimelta toiselle. Toisen palvelinlaitteen lisäksi järjestelmään lisättiin toinen tallennusalue levyjärjestelmästä, jotta virtuaalipalvelimille voitaisiin provisioida levyjärjestelmästä levytilaa entistä vapaammin.

Käytössä olevia, Microsoft Virtual Server -ympäristössä toimivia virtuaalikoneita siirrettiin ESX-ympäristöön, jotta virtuaalikoneet olisivat hallittavissa keskitetymmin eivätkä hajautettuna usealle yksittäiselle palvelimelle. Samalla myös vanhoja fyysisiä palvelimia virtualisoitiin laitekaappitilan vapauttamiseksi ja ikääntyvän laitteiston laitevioilta säästymiseksi. Kaikki siirrettävät tai konvertoitavat palvelimet olivat Windows -palvelimia, joten työssä voitiin hyödyntää VMWaren

Converter -sovellusta. Converter asentaa virtualisoitavaan kohdejärjestelmään agenttisovelluksen, jonka avulla koko järjestelmä kaikkine tietoineen on mahdollista kopioida ESX-järjestelmään virtuaalipalvelimeksi.

5.2 Ratkaisujen kustannukset ja vertailu tilanteeseen ilman virtualisointia

Ohjelmistovirtualisointijärjestelmän osalta kustannukset muodostuvat palvelinlaitteesta, levyjärjestelmän tallennuskapasiteetista, käyttöjärjestelmälisenssistä ja SoftGrid-lisensseistä. SoftGrid-työasemalisenssit ovat osa Microsoft Desktop Optimization Pack:iä ja SoftGrid for Terminal Servers on hankittava erikseen. Desktop Optimization Packin kustannukset muodostuvat kuukausittaisesta maksusta työasemien määrän mukaan. SoftGrid Serverin lisenssi puolestaan kuuluu asiakasohjelmien lisensseihin. Järjestelmän pystytys- ja ylläpitokustannuksien lisäksi kustannuksia aiheuttavat myös ylläpitohenkilöstön koulutus järjestelmän käyttöön ja sovellusten virtualisointityö.

Laitteistovirtualisoinnin järjestelmän osalta kustannukset muodostuvat VMWare ESX -palvelinlaitteistosta, -lisensseistä, tallennuskapasiteetista, Virtual Center palvelinlaitteesta ja sen käyttöjärjestelmästä sekä henkilöstön kouluttamisesta ja opastamisesta järjestelmän käyttöön. Microsoft Virtual Serverin kustannukset muodostuvat palvelinlaitteesta, isäntäkäyttöjärjestelmästä ja tallennuskapasiteetista. Itse virtualisointijärjestelmien lisäksi kustannuksia aiheuttavat virtuaalisten palvelinten käyttöjärjestelmälisenssit, mutta nämä kustannukset muodostuisivat ilman virtualisointiakin.

Ohjelmistovirtualisoinnin suoranaiset loppukäyttäjälle näkyvät muutokset ovat sovelluksen käyttöönottoajassa ja sovelluksen käynnistyksessä. SoftGrid asiakassovellus tuo käyttöjärjestelmän tehtäväpalkkiin kuvakkeensa ja antaa tietoa sovelluksen latautumisen, käynnistymisen ja sammumisen yhteydessä. Kuviossa 11 näkyy SoftGridin sinivalkoinen kuvake ja ilmoitukset, kuinka järjestelmä lataa, käynnistää ja sammuttaa sovelluksen. Näiden lisäksi samalla tavoin järjestelmä ilmoittaa, jos sovelluksen käynnistämisessä on ilmennyt ongelma, eikä sovellusta voida käynnistää.



KUVIO 11. SoftGridin ilmoitukset sovelluksen suorituksesta.

Sovellusten käyttöönottoaika asiakkaan tarpeen ilmaisemisesta ja sovelluksen asennuksen hyväksynnästä väheni tunneista tai päivistä muutamiin minuutteihin jo virtualisoitujen sovellusten kohdalla. Virtualisoimattomien sovellusten tapauksessa sovelluksen toimitusaika pysyi pääasiassa ennallaan. Tällaisissa tapauksissa virtualisoidun sovelluksen testaajana toimi sovelluksen loppukäyttäjä, jolla useimmissa tapauksissa on jo valmiiksi tuntemus sovelluksen toiminnasta, ja näin ollen mahdollisuus toimia testaajana. Vaikeasti virtualisoitavien sovellusten tilanteessa sovellus voidaan edelleen asentaa paikallisesti työasemaan ja jatkaa virtualisoidun sovelluksen testaamista ja muokkaamista tarpeen mukaan seuraavaa sovelluksen tilausta varten.

Laitteistovirtualisoinnin osalta muutokset eivät näy palveluiden loppukäyttäjien suuntaan. Teknisen henkilöstön osalta palvelimien hallinta helpottuu hallintatyökalujen avulla, joilla virtuaalisen palvelimen ominaisuuksia voidaan muokata, ja palvelinta voidaan käyttää aivan kuin istuttaisiin palvelimen näppäimistön ääressä. Vaikka esim. palvelimen muistin määrän lisääminen ei onnistukaan palvelinta sammuttamatta, on se silti nopeampi toimenpide virtuaalisena kuin fyysisenä toimenpiteenä, ja tämän vuoksi käyttöjatkos palvelussa on huomattavasti lyhyempi.

Uusien palvelimien ja etenkin testipalvelimien osalta käyttöönottokynnys laski huomattavasti, koska testaamista varten uutta laitetta ei tarvinnut enää hankkia. Virtualisointijärjestelmästä provisioitiin testaamista varten tarvittava määrä resursseja ja käyttöjärjestelmä plus sovellukset asennettiin virtuaaliseen palveli-

meen. Tiettyjen käyttöjärjestelmien osalta edes ominaisuuksien määrittelyä ja käyttöjärjestelmän asennustyötä ei ollut tarpeen tehdä, koska voitiin hyödyntää esiasennettua ja määriteltyä palvelin pohjaa.

5.3 Käyttäjäpalaute

Ohjelmistovirtualisoinnin osalta varsinaista palautekyselyä ei loppukäyttäjille tai ylläpitäjille tehty. Palautetta kuitenkin tuli molemmilta tahoilta, ja pääasiassa palaute oli positiivista. Laitteistovirtualisointi ei vaikuttanut palvelun loppukäyttäjille merkittäväällä tavalla, minkä vuoksi laitteistovirtualisointia ei ole huomioitu palautteessa.

Loppukäyttäjät olivat tyytyväisiä sovellusten toimitusnopeuteen ja siihen, ettei heidän tarvinnut keskeyttää työntekoaan tai sopia erikseen aikaa sovelluksen asennukselle. Lisäksi sovellusongelmien nopeasta korjaamisesta tuli positiivista palautetta. Kenenkään mielestä asiakasohjelmiston antamat ilmoitukset sovelluksen käynnistyksestä eivät olleet häiritseviä, vaikka kyselyä ilmoituksista tulikin muutamalta henkilöltä. Sovelluksen käynnistymisen hitaudesta raportoi muutama käyttäjä, mutta hitaus ei näissä tapauksissa johtunut ohjelmistovirtualisoinnista. Järjestelmien tuki ja ylläpito antoivat positiivista palautetta sovellusten toimittamisen helppoudesta. Kun sovellus voidaan toimittaa loppukäyttäjän käyttöön lisäämällä tämä sopivaan käyttäjäryhmään, ei tarvitse sopia asennusajankohtia eikä toistaa rutiininmukaista sovellusasennusta, ja näin ollen aikaa jää muihin työtehtäviin enemmän. Sovellusongelmien tilanteessa raskasta vianselvitystä ei välttämättä tarvitse tehdä, koska sovellus on helppo palauttaa ennalta määrättyyn tilaan ja asetuksiin. Osa sovellusten ongelmista on kuitenkin hankalampi selvittää virtualisointia käytettäessä, koska esimerkiksi sovelluksen tiedostoja ei päästä katsomaan tai muokkaamaan suoraan tiedostojärjestelmän kautta ilman erillisiä toimenpiteitä.

5.4 Ratkaisulla saavutetut hyödyt

Suurimpia saavutettuja ohjelmistovirtualisoinnin etuja on sovellusten jakelun helppous ja yhteensopivuusongelmien huomattava väheneminen. Uusia sovelluk-

sia tai jopa verkosta käytettäviä järjestelmiä voidaan jakaa loppukäyttäjän työpöydälle kuvakkeiksi, ja näistä kuvakkeista käynnistämällä selaimessa on automaattisesti käytössä oikeat lisäsovellukset ja/tai Java -ympäristön versio. Sovelluksen saattaminen loppukäyttäjän käyttöön ei enää vie tunteja tai päiviä aikataulujen yhteensovittamisen vuoksi, vaan sovellus saadaan käyttöön sekunneissa tai minuuteissa sovelluksen tarpeen hyväksynnästä. Sovelluksen jakelun helpottuessa näin merkittävästi säästetään aikaa sekä työaseman loppukäyttäjältä että järjestelmätukihenkilöiltä.

Laitteistovirtualisoinnin osalta VMWare ESX:n avulla voidaan tarjota vikasietoisia palvelinalustoja omiin tai asiakkaiden tarpeisiin. Vikasietoisuus ei siis enää vaadi kahta fyysistä laitetta jokaista palvelinta kohden, vaan palvelut voidaan keskittää VMWare-ympäristöön ja tuoda vikasietoisuus automaattisen migraation avulla. Näin palvelun itsessään ei tarvitse tukea kahdentamista, eikä sen suhteen tarvitse tehdä monimutkaisia toimenpiteitä. Tietenkään tällainen kahdennus ei kahdenna itse palvelua ja esim. virtuaalikoneen käyttöjärjestelmän huolto tuo kuitenkin käyttökatkoksen palveluun. Automaattisen migraation lisäksi virtuaalikoneet voidaan siirtää alustalta toiselle myös manuaalisesti, jolloin toinen virtuaalipalvelin voidaan ajaa alas päivitystä tai huoltoa varten ilman, että palvelualustassa olevissa virtuaalikoneissa toimivat palvelut häiriintyisivät tai että palveluihin tulisi katkoksia.

Vikasietoisuuden lisäksi laitteistovirtualisoinnin avulla saavutetaan huomattava nopeusetu uusien palvelimien käyttöönotossa, kun perinteinen käyttöjärjestelmän asennus voidaan korvata palvelimen kopioinnilla valmiista pohjasta. Tätä tekniikkaa hyödyntäen voidaan luoda pohja käyttöjärjestelmästä ja hyödyntää tätä valmiiksi asennettua kokonaisuutta uusien palvelimien pohjana.

Virtualisointi voi olla ratkaisuna myös tilanteisiin, joissa palvelu on toteutettu jollekin vanhalle käyttöjärjestelmälle (esim. Windows NT 4), eikä tällaista käyttöjärjestelmää enää voida asentaa uusiin laitteisiin laitetuen ja ajurien puuttumisen vuoksi. Tällöin palvelua tuottava tietokone voidaan konvertoida virtuaalikoneeksi, ja palvelua tuottaneesta vanhasta ja helposti vikaantuvasta laitteesta voidaan luopua.

5.5 Mahdolliset tuotekonseptit

Ohjelmistovirtualisoinnin hyödyntäminen asiakastuotteissa on hankalaa pääasiassa lisenssikäytäntöjen vuoksi. Sekä itse virtualisointijärjestelmän että virtualisoidun sovellusten lisenssien pitäisi olla vuokrattavissa loppuasiakkaalle. Tämän vuoksi ohjelmistovirtualisointia voitaisiinkin myydä vain konsulttipalveluna asiakkaan omaan tietojärjestelmään. Suurin hyöty ohjelmistovirtualisoinnista saadaan tilanteessa, jossa virtualisointia myyvä yritys ylläpitää asiakkaan tietojärjestelmiä. Kyseisiä palveluita pienille ja keskisuurille yritysasiakkaille tuottavia yrityksiä vastaan kilpailtaessa, ohjelmistovirtualisointi toisi uusia lähestymistapoja sovellusten hallinnan kannalta.

Laitteistovirtualisointi puolestaan tuo huomattavasti ohjelmistovirtualisointia helpommat mahdollisuudet hyödyntää virtualisointitekniikkaa asiakastoteutuksissa. Kevyttä palvelintarvetta varten asiakkaalle voidaan vuokrata virtuaalipalvelin fyysisen palvelimen sijaan. Virtuaalisen palvelimen tapauksessa asiakas ei luultavasti edes huomaa eroa fyysiseen laitteeseen paitsi huomattavasti nopeampina toimitusaikoina. Laitetta vuokraavan osapuolen edut virtuaalipalvelimen tapauksessa muodostuvat säästöistä laitetilassa, sähkönkulutuksessa, laitteiden ja sähkönsyötöiden asennuksissa, kaapeloinneissa sekä virtuaalisten palvelimien resurssien provisioinnissa.

ASP-tekniikalla toteutetut palvelut ovat perinteisempiä tuotteina myytäviä palveluita. Tietoliikenneoperaattorit tarjoavat asiakkailleen palveluna yleensä vähintään selaimella toimivan sähköpostin. Yrityksille pelkkä selainsähköposti ei välttämättä kuitenkaan riitä, vaan palveluun halutaan lisätä ehkäpä kalenteri, yrityksen yhteystiedot, intranetsivusto tai vaikkapa yrityksen oma sisäinen pikaviestinpalvelu. Tällaisia palveluita voidaan toteuttaa erinäisinä tuotteina, jotka sisältävät kaikki tai osan edellä mainituista ominaisuuksista. Palvelun loppuasiakas käyttää tilaamia palveluita selaimella tai palvelun käyttöön tarkoitettulla asiakasohjelmistolla. Loppuasiakkaalle palvelu toimii kuin se olisi asiakkaan omassa verkossa ja omilla laitteilla, vaikka samoja palveluita tuotetaan samoilla laitteilla useammalle asiakkaalle. Asiakkaat eriytetään toisistaan palvelun turvaominaisuuksien avulla.

Eristyksen vaatimien toiminnallisuuden ja lisenssikäyttöjen vuoksi kaikkia asiakkaiden toivomia palveluita ei kuitenkaan voida aina tuottaa ASP-mallilla.

6 TULOKSET

Opinnäytetyölle asetetut tavoitteet saavutettiin, ja sovellusten yhteensopivuusongelmat etenkin verkkosovellusten Java vaatimusten osalta poistuivat. Sovellusten jakelu loppukäyttäjille jo virtualisoitujen sovellusten osalta nopeutui huomattavasti, koska järjestelmätukihenkilön ei enää tarvinnut keskeyttää sovelluksen loppukäyttäjän työntekoa ja asentaa sovellusta käsin, tai vaihtoehtoisesti loppukäyttäjän ei tarvinnut odottaa sovelluksen asentumista käytössä olevasta sähköisestä ohjelmistonjakelujärjestelmästä. Käytännössä virtualisoidun sovelluksen asennusajan voidaan ajatella olevan muutamia sekunteja, koska sovellus saadaan loppukäyttäjän käyttöön lisäämällä hänet sovelluksen mahdollistavaan AD:n käyttäjäryhmään ja opastamalla sovelluslistan päivittäminen työasemalta.

Ohjelmistovirtualisoinnin tuoma sovellusten korjausominaisuus (eli sovelluspaketin tilan palauttaminen) poisti useammassa tapauksessa tarpeen keskeyttää työntekoa sovelluksen uudelleenasetuksen vuoksi. Vaikka virtualisointi tuo yritykselle huomattavat säästöt, ei se kuitenkaan ole täysin ongelmatonta. Heikosti testattu virtualisoitu sovellus voi toimia näennäisesti täysin oikein useimmissa tilanteissa, mutta joissain tilanteissa toimintahäiriöitä voidaan havaita. Työn aikana tällainen vikatilanne tuli vastaan erään sovelluksen asetuksien tallentumisongelmana, jolloin sovelluksessa käytettävää tulostinta ei voitu tallentaa sovelluksen käyttämään asetustiedostoon. Vikatilanne ratkaistiin siirtämällä sovelluksen asetustiedosto virtuaalisen sovelluspaketin ulkopuolelle, jonka jälkeen tallennus tiedostoon onnistui.

Laitteistovirtualisoinnin osalta edellisen järjestelmän rajoituksista päästiin virtualisointijärjestelmän vaihdon yhteydessä. Samalla uusi järjestelmä toi mukanaan ominaisuuksia, joita vanhassa järjestelmässä ei ollut. Uusia ominaisuuksia otettiin käyttöön tarpeen mukaan, ja näiden ominaisuuksien myötä esim. virtuaalikoneiden hallinnan hajauttaminen sitä tarvitseville on entistä helpompaa.

7 ANALYYSI JA JOHTOPÄÄTÖKSET

Ohjelmistovirtualisointi tarjoaa yrityksille mahdollisuuden säästää sovellusten hallintakustannuksissa ja tarjota työntekijöilleen heidän työssään tarvitsemat sovellukset nopeasti ja työntekoa keskeyttämättä. Ongelmatilanteista selviäminen virtualisoinnin avulla on entistä joustavampaa ja nopeampaa, koska sovellusta ei tarvitse asentaa uudelleen, vaikka itse sovellus vikaantuisi. Sovellusten mahdollisen jakelun ja helpon viankorjauksen lisäksi sovelluksia virtualisoimalla voidaan selvittää tilanteista, joissa sovellukset tai sovellusversiot eivät perinteisesti toimisi samaan aikaan asennettuina. Suurin hyöty käyttäjäperustaisesta sovellusjakelusta (esim. Microsoft SoftGrid) on suurissa ympäristöissä, joissa käyttäjät eivät välttämättä jatkuvasti työskentele samalla työasemalla tai joissa uusia käyttäjiä ja työasemia tulee lisää nopeaan tahtiin. Tällaisissa tilanteissa käyttäjän tarvitsemat sovellukset voidaan määrittää jo käyttäjätunnusta luotaessa, ja ne ovat käytettävissä heti ensimmäistä kertaa työasemalle kirjaututtaessa, oli työasema mikä tahansa. Nykyiset ohjelmistovirtualisointijärjestelmät ovat toimivia, ja niillä voidaan saavuttaa edellä mainittuja hyötyjä.

Laitteistovirtualisoinnista on hyötyä yrityksille, joilla on paljon matalalla kuormituksella olevia palvelimia. Keskittämällä tällaisia palvelimia virtuaalisiksi palvelimiksi voidaan saada huomattaviakin säästöjä tilojen ja sähkönkulutuksen kannalta. Laitteistojen virtualisointi tuo mukanaan myös muita hyvin käytännöllisiä mahdollisuuksia, joita yrityksissä voidaan ottaa käyttöön tarpeen mukaan. Etenkin testiympäristöissä virtualisointia kannattaa hyödyntää, koska tällaisissa tapauksissa palvelinten kuormat ovat aina alhaisia, ja koko testiympäristö saadaan luotua hetkessä monistamalla asennetusta palvelimesta tarvittava määrä kopioita. Tämänhetkiset laitteistovirtualisointiratkaisut tarjoavat siis fyysiseen laitteeseen nähden etuja matalakuormituksellisissa ja vähän resursseja vaativissa palvelimissa. Virtualisointiratkaisuja on saatavilla useita, ja ne toimivat pääsääntöisesti hyvin. Laitteistovirtualisoinnin ominaisuusvaatimusten ollessa pienet ovat virtualisointiympäristön perustuskustannuksetkin pienet, koska sekä Microsoft Virtual Server, että VMWare GSX tarjoavat erinomaisen mahdollisuuden hyödyntää virtualisointitekniikkaa minimaalisilla kustannuksilla.

8 TULEVAISUUS

Microsoft SoftGrid 4.5 version myötä ohjelmiston nimi muutetaan Microsoft Application Virtualization:ksi. Tärkeimpinä uusina ominaisuuksina 4.5 versioon on tulossa mm:

- sovelluspakettien automaattinen lataaminen asiakaslaitteelle
- SoftGrid asiakasovelluksen kieliversiot
- uusia sovellustenjakelutapoja
- SoftGrid pakettien dynaaminen yhdistäminen keskenään.

Näiden lisäksi SoftGrid-virtualisointia tulee olemaan mahdollista käyttää irrallisena ilman suoratoisto-ominaisuutta, jolloin sovellusten jakelu työasemille täytyy hoitaa muita jakelukanavia käyttäen, esim. ESD-järjestelmällä tai CD-levyllä.

Windows Server 2008 Terminal Services tuo edellisiin versioihin uutena ominaisuutena Citrix Presentation Serveristä tutun ominaisuuden, jolla ajettavan sovelluksen ikkuna tuodaan palvelimelta loppukäyttäjän työpöydälle koko palvelimen työpöydän sijaan poistamatta kuitenkaan mahdollisuutta tuoda edelleen koko työpöytä käyttöön. Sovellusten käynnistäminen etäpalvelimelta voidaan tehdä tiedostotyypimäärityksen, pikakuvakkeen tai verkkosivulla olevan linkin avulla. Tämä toiminnallisuus tekee Windows Server 2008:n Terminal Servicesistä mielenkiintoisen haastajan Citrix Presentation Serverille etäkäytön osalta. Jos tarvittavat ominaisuudet ja hallintamahdollisuudet löytyvät jo pelkästä käyttöjärjestelmän palvelusta, voi olla vaikea perustella lisäsovelluksen kustannuksia.

Windows Server 2008 tuo mukanaan käyttöjärjestelmään sisäänrakennetun uuden laitteistovirtualisointiratkaisun, Hyper-V:n. Hyper-V:n avainominaisuuksiin kuuluu mm:

- uusi ja kehittynyt arkkitehtuuri, joka perustuu 64-bittiseen mikrokernelissä sijaitsevaan VMM:iin
- laaja käyttöjärjestelmätuki mukaan lukien 32- ja 64-bit versiot Windowsista ja Linuxista
- SMP-tuki mahdollistaa useamman suorittimen provisioinnin virtuaaliselle palvelimelle

- tehokkaampi isäntäpalvelimen resurssien käsittely
- virtuaalipalvelimien valokuvaominaisuus
- nopea virtuaalipalvelimen migraatio isäntäpalvelimelta toiselle.

Jos Microsoft lunastaa lupauksensa Hyper-V:n ominaisuuksista, tekee Hyper-V Windows Server 2008:sta kilpailijan VMWare ESX:lle. Windows Server 2008 -laitevaatimusten lisäksi Hyper-V tarvitsee toimiakseen laitteistolta rautatason (Intel VT tai AMD-V) virtualisointituen. Kevyet järjestelmävaatimukset lisäävätkin Hyper-V:n kiinnostavuutta ESX:ään verrattuna.

Virtualisointi sekä laitteistojen että ohjelmistojen osalta on kehittynyt viime vuosina nopeasti. Etenkin Microsoftin ajatus rakentaa virtualisointi osaksi Windows Server 2008:aa, kertoo ohjelmistojätin positiivisen näkemyksen virtualisoinnin tulevaisuudesta. Tulevaisuus tarjoaakin hyvin mielenkiintoisia mahdollisuuksia hyödyntää virtualisointitekniikoita jokapäiväisessä tietokoneen käytössä. Käyttäjien kasvavat tarpeet ja vaatimukset järjestelmien ominaisuuksille tulevat muokkaamaan järjestelmistä monipuolisia tai hyvin muihin järjestelmiin integroituvia osakokonaisuuksia.

LÄHTEET

- Altiris 2006. Altiris Software Virtualization Solution 2.0 White Paper [verkkojulkaisu]. Altiris [viitattu 8.1.2008]. Saatavilla: www.altiris.com
- Brain. 2008. How ASPs Work [verkkojulkaisu]. How Stuff Works Inc. [viitattu 12.2.2008]. Saatavilla: www.howstuffworks.com
- Brown. 2005. To Install or Not Install: Solving the Application Management Nightmare Part II [verkkojulkaisu]. DABCC Inc [viitattu 9.1.2008]. Saatavissa: <http://www.dabcc.com/documents/Part II - To Install or Not Install - Rev 2.pdf>
- Citrix Systems Inc. 2007. Citrix Presentation Server 4.5: Administration
- Lukkari J. 2007. Sähkösyöppö konesali. Tekniikka & Talous 28/2007, 16 - 17.
- Microsoft 2005. Technical Overview of Terminal Services [verkkojulkaisu]. Microsoft [viitattu 8.1.2008]. Saatavilla: www.microsoft.com
- Microsoft 2007. Remote Desktop Protocol [verkkojulkaisu]. Microsoft [viitattu 15.1.2008]. Saatavissa: <http://msdn2.microsoft.com/en-us/library/aa383015.aspx>
- Microsoft 2008. Microsoft SoftGrid 4.2 release notes and updates [verkkojulkaisu]. Microsoft [viitattu 3.1.2008]. Saatavissa: <http://support.microsoft.com/kb/941408>
- ProTrainIT Oy. 2006. Virtual Server 2005 R2 ProLAB
- Sean, Donahue, Flanagan, Kelly, Kinney, Yuen, Lanoix, Kim, Richardson. 2006. SoftGrid 4.0 Administration Training Guide
- SWSOFT 2008. An Introduction to OS Virtualization and Virtuozzo [verkkojulkaisu]. SWSOFT [viitattu 29.1.2008]. Saatavilla: www.parallels.com/en/products/virtuozzo/whitepapers/
- The Linux Information Project 2006. The X Window System: A Brief Introduction [verkkojulkaisu]. The Linux Information Project [viitattu 17.1.2008]. Saatavilla: <http://www.linfo.org/x.html>
- TIEKE Tietoyhteiskunnan kehittämiskeskus ry 2008. Käsiteluettelo [verkkojulkaisu]. TIEKE Tietoyhteiskunnan kehittämiskeskus ry [viitattu 12.2.2008]. Saatavilla: http://www.tieke.fi/verkkokaveri/teemat/tietotekniikkahankinnat/ratkaisuvaihtoehdotja_ohjelmiston/kasiteluettelo/

Torikka M. 2007. Virtualisointi viherryttää konesaleja. Tekniikka & Talous
28/2007, 18.

VMWare Inc. 2006. Virtual Infrastructure with ESX Server and VirtualCenter
Student Guide

Wikipedia 2008a. Secure Shell [verkkajulkaisu]. Wikipedia [viitattu 12.2.2008].
Saatavilla: http://en.wikipedia.org/wiki/Secure_Shell

Wikipedia 2008b. Virtualization [verkkajulkaisu]. Wikipedia [viitattu 8.1.2008].
Saatavissa: <http://en.wikipedia.org/wiki/Virtualization>