



- OPINNÄYTETYÖ - AMMATTIKORKEAKOULUTUTKINTO
TEKNIIKAN JA LIIKENTEEN ALA

VIRTUALISOINTIALUSTO- JEN VERTAILU

TEKIJÄ/T: Antti Sirviö

Koulutusala Tekniikan ja liikenteen ala	
Koulutusohjelma Tietotekniikan koulutusohjelma	
Työn tekijä(t) Antti Sirviö	
Työn nimi Virtualisointialustojen vertailu	
Päiväys 23.5.2017	Sivumäärä/Liitteet 21/1
Ohjaaja(t) Pekka Granroth	
Toimeksiantaja/Yhteistyökumppani(t) Itä-Suomen yliopisto / Bioinformatiikan keskus	
<p>Tiivistelmä</p> <p>Opinnäytetyön tarkoituksena oli tehdä virtualisointialustojen vertailu toimeksiantona Itä-Suomen yliopistolle, Bioinformatiikan keskukselle. Vertailussa mitattiin suorituskykyeroja natiivin ja virtualisoidun käyttöjärjestelmän välillä käyttäen Phoronix-Test-Suite testiohjelmistoa. Toimeksiantajan pyynnöstä vertailussa olevat virtualisointialustat olivat WMWare ESXi, Microsoft Hyper-V, Linux KVM, Citrix XenServer sekä Oracle Virtualbox.</p> <p>Jokaiselle virtualisoidulle tietokoneelle varattiin 16 Gb keskusmuistia ja 4 prosessorin säiettä. Käyttöjärjestelmänä toimi CentOS 7. Virtualisoidut tietokoneet olivat 1, 2 ja 4 yhdistelminä.</p> <p>Testiympäristö koostui kahdesta tietokoneesta. Toinen tietokoneista toimi virtualisointialustana, ja toinen hallintatietokoneena sekä tulosten käsittelytietokoneena. Suorituskykytestin komentoskripti syötettiin yhtäaikaista virtuaalisille tietokoneille SSH/ClusterSSH – ohjelmistolla, joka toimi Screenin päällä.</p> <p>Phoronix-Test-Suiten sisältämät testit sekä ennalta valmiiksi määritetty paikallinen R-testi suoritettiin batch-ajona, ja jokainen testi suoritettiin kolme kertaa. Tulokset yhdistettiin yhdeksi csv-tiedostoksi Pythonin avulla. Tulokset luovutettiin Itä-Suomen yliopistolle, Bioinformatiikan laitokselle.</p>	
Avainsanat Virtualisointi, vertailu, virtualisointialusta, ESXi, KVM, Hyper-V, XenServer, Virtualbox	

Field of Study Natural Sciences			
Degree Programme Degree Programme in Information Technology			
Author(s) Antti Sirviö			
Title of Thesis Comparison of virtualization platforms			
Date	23.5.2017	Pages/Appendices	21/1
Supervisor(s) Mr. Pekka Granroth / Lecturer			
Client Organisation /Partners University of Eastern Finland / Bioinformatics center			
<p>Abstract</p> <p>The purpose of this thesis was to do a comparison of virtualization platforms for University of Eastern Finland, Bioinformatics Center. In the comparison it was measured performance differences between native versus virtualized operating system with Phoronix-Test-Suite benchmark. As per request of University of Eastern Finland, selected virtualization platforms were WMWare ESXi, Microsoft Hyper-V, Linux KVM, Citrix XenServer and Oracle Virtualbox.</p> <p>For each virtualized computer (guest) was allocated 16 Gb of memory and 4 processor threads. Operating system used was CentOS 7. The combination of virtualized computers were 1, 2 and 4.</p> <p>The test environment consisted from two computers. Other computer acted as a host for virtualization platform, and other one was the management computer and test result placeholder. Command for test-script was set simultaneously on all virtualized computers were inserted via SSH/ClusterSSH, that were run on top of Screen.</p> <p>Built-in performance tests from Phoronix-Test-Suite and pre-defined local test for R were run as a batch, and every test were run three times total. Test results were combined to a single csv-file with Python. Detailed test results were given to the University of Eastern Finland, Bioinformatics Center.</p>			
Keywords Virtualization, comparison, virtualization platforms, ESXi, Hyper-V, KVM, XenServer, Virtualbox			

SISÄLTÖ

1	JOHDANTO	6
2	VIRTUALISOINTI.....	7
2.1	Mitä virtualisointi on.....	7
2.2	Virtualisoinnin nykytilanne (ja historiaa)	7
2.3	Virtualisoinnin edut	8
3	VIRTUALISOINTIALUSTAT	10
3.1	Virtualisointialustojen tyypit.....	10
3.2	VMWare ESXi.....	11
3.3	Microsoft Hyper-V	11
3.4	Linux KVM	11
3.5	Citrix Xenserver	12
3.6	Oracle Virtualbox	12
4	TESTIYMPÄRISTÖ JA TESTIEN SUORITTAMINEN	13
4.1	Testiympäristö.....	13
4.2	Testiohjelman Phoronix-Test-Suite	14
4.3	Kaappausmedian luonti	15
4.4	Testien suorittaminen.....	15
5	TULOKSET	17
5.1	Saadut tulokset.....	17
6	YHTEENVETO.....	20
	LÄHTEET	21
	LIITE 1: SAADUT TULOKSET KAAVIOINA.....	22

Työssä käytetyt lyhenteet ja termit

Host	Fyysinen tietokone, toimii virtualisointialustana
Guest	Looginen, virtualisoitu tietokone
Hypervisor	Virtualisointialusta, termi
VMM	Virtual Machine Monitor, Hypervisor
SSH	Secure Shell, etäyhteys
R	Ohjelmointikieli
Allokointi	Resurssien jakamista
Emulointi	Toisen laitteen tai järjestelmän jäljitystä
Simulointi	Tapahtuman jäljennystä
Kahdentaminen	Toissijainen varajärjestelmä
Kernel	Ydin (käyttöjärjestelmän)
CentOS 7	Käyttöjärjestelmä, Linux
CSV-tiedosto	Comma Separated Value (CSV), tiedostomuoto
Python	Ohjelmointikieli

1 JOHDANTO

Virtualisointi on yleistynyt huomattavasti viimeisen 20 vuoden aikana. Eräänä syynä virtualisointitekniikan yleistymiseen etenkin palvelinten kohdalla on se, että virtualisointi mahdollistaa eri ohjelmistojen sekä ohjelmistoversioiden käytön kehitys-, testaus- sekä tuotantoympäristöissä. Lisäksi virtualisointi vähentää laiteinvestointitarpeita ja lisää palvelinlaitteiston käyttöastetta. (Dittner & Rule 2007, 5-8)

Virtualisointi on eräänlainen tekniikka, joka mahdollistaa useamman loogisen käyttöjärjestelmän suorittamisen yhdellä fyysisellä tietokoneella. Virtuaaliset käyttöjärjestelmät eivät ole niin sanottuja fyysisiä koneita, vaan toimivat tietokoneen ja käyttöjärjestelmän välissä olevalla virtualisointialustalla (hypervisor). Virtualisointia voidaan myös jollain tavalla verrata emulointiin (Chisnall 2008, 24).

Tämän opinnäytetyön tarkoituksena on verrata natiivin käyttöjärjestelmän ja virtualisointialustojen suorituskykyeroja testiohjelmistolla (Phoronix-test-suite). Kun tulokset natiivista käyttöjärjestelmästä on saatu, asennetaan testiympäristön tietokoneelle virtualisointialusta, jonka päälle kyseinen käyttöjärjestelmä asennetaan ja ajetaan testiohjelmat uudestaan.

Opinnäytetyö toteutetaan toimeksiantona Itä-Suomen yliopistolle, bioinformatiikan keskukselle. Opinnäytetyön toimeksiantajan pyynnöstä tyyppin 1 virtualisointialustoina vertailussa mukana ovat VMWare ESXi, Microsoft Hyper-V, Linux KVM sekä Citrix XenServer. Lisäksi tyyppin 2 virtualisointialustoista otetaan vertailuun mukaan Oracle Virtualbox. Alkuperäisen suunnitelmaan kuuluva VMWare Workstation vertailualusta jäi pois toimeksiantajan asettaman aikarajan johdosta.

2 VIRTUALISOINTI

Tässä kappaleessa tarkastellaan mitä virtualisointi on, nykytilannetta sekä etuja. Teorieettisessä tarkastelussa hyödynnetään virtualisoinnin käsikirjoja (hands-on guide), koska tieteellisiä julkaisuja virtualisoinnista on vähän. Suomessa Markku Tikkanen (2008) on tehnyt kirjallisuuskatsauksen palvelimien virtualisoinnista ja Petri Siiskonen (2011) vertaillut joitakin virtualisointitekniikoita. Tieteellisten julkaisujen vähyyks voinee selittyä osaltaan valmistajien omilla tutkimuksilla, mitkä eivät ole julkisia.

2.1 Mitä virtualisointi on

Virtualisointi voidaan määritellä siten, että tietokoneen resurssit (Host) jaetaan yhdeksi tai useammaksi suorittavaksi tietokoneeksi (Guest) käyttämällä yhtä tai useampaa virtualisointitekniologiaa. Virtualisoidut tietokoneet (Guest) kohtelevat niille allokoituja resursseja ikään kuin todellisina, ja virtualisointialustan tehtävänä on varmistaa tämän illuusion saumattomuus. Hypervisorin (VMM) tehtävä on lisäksi rajoittaa virtualisoitujen tietokoneiden (Guest) saatuutta fyysisen raudan resursseihin virtualisoitujen tietokoneiden (Guest) välillä. Virtualisointi tunnetaan esimerkiksi termeillä ajan jakaminen, emulointi, osittain- tai kokonaan simulointi. (Dittner & Rule 2007, 6-7; Matthews, Dow, Deshane, Hu, Bongio, Wilbur & Johnson 2008, 15.)

Virtualisoinnin määritelmä täyttyy, mikäli osa seuraavista argumenteista pitää paikkaansa (Dittner & Rule 2007, 2, 16):

- Lisää erottelevan kerroksen sovelluksen ja laitteiston väliin
- Vähentää kustannuksia ja monimutkaisuutta
- Tarjoaa mahdollisuuden rajata tietokoneen resursseja parantamalla luotettavuutta ja turvallisuutta
- Parantaa palvelun tasoa sekä palvelunlaatua (Quality of Service, QoS)
- IT infrastruktuurin tarve vastaa paremmin yrityksen tavoitteisiin
- Poistaa päällekkäisyyksiä sekä nostaa käyttöastetta IT infrastruktuurissa

2.2 Virtualisoinnin nykytilanne (ja historiaa)

Jo vuosikymmen sitten virtualisointia kohtaan alkaneen lisääntyneen kaupallisen kiinnostuksen vuoksi on hyvä tarkastella virtualisoinnin taustaa. Palvelimien virtualisointi on tyypillisin ja ensimmäinen muoto virtualisoinnista. Teknologiana virtualisointi on ollut olemassa 1960-luvulta lähtien. Tietotekniikan maailman tietoisuuteen se tuli ajan jakamisena sekä virtuaalisena muistina Atlas- sekä M44/44X- projektien myötä. (Rule & Dittner 2007, 31.)

Virtualisointitekniikan kehittymisen myötä IBM kehitti varhaisen arkkitehtuurin hypervisorista (VMM). Hypervisorit kiinnostivat tietokonetutkijoita niiden ”virhekestävyyden” (robustness) ja vakauden (stability) vuoksi. Hypervisorit mahdollistivat usean virtualisoidun tietokoneen (guest) käyttämisen yhtäaikaaisesti, eikä yhden virtuaalisen tietokoneen kaatuminen vaikuttanut muiden virtuaalisten tietokoneiden suoriutumiseen. Juuri tästä syystä uusia käyttöjärjestelmiä sekä kokeellista koodia suoritetaan vakaan koodin ohella virtualisoidussa ympäristössä. (Matthews, Dow, Deshane, Hu, Bongio, Wilbur & Johnson 2008, 29.)

Ensimmäinen kaupallisesti saatavilla oleva virtualisointiin suunniteltu tietokonejärjestelmä oli IBM System/370 (Matthews, Dow, Deshane, Hu, Bongio, Wilbur & Johnson 2008, 29). Monet System/370 järjestelmän sekä sen onnistuneiden versiokehitysten (iteraatiot) ominaisuudet ja suunnittelun elementit ovat edelleen käytössä nykypäivän virtualisointitekniikoissa (Rule & Dittner 2007, 31).

1990 – luvulta alkaen trendinä on ollut siirtyminen palvelinsaleihin (data center) keskitettyjen palvelinratkaisujen sijaan. Suureksi osaksi tämä on johtunut siitä, että palvelimien käyttöaste sille määritetyillä resursseilla oli tätä ennen hyvin vähäistä (”server sprawl”) ja sovelluksien ja sovellusalojen määrän kasvaessa palvelimien tarve kasvaa eksponentiaalisesti. Keskitetyt palvelimet nähtiin liian kalliina ratkaisuna hankittavaksi ja ylläpidettäväksi. (Matthews, Dow, Deshane, Hu, Bongio, Wilbur & Johnson 2008, 7.) Tästä syystä virtualisointi on otettu laajalti käyttöön palvelinratkaisujen toteutuksen yhteydessä.

2.3 Virtualisoinnin edut

Yrityksen koosta huolimatta useimmat tietokoneella tapahtuvat toiminnot soveltuvat tehtäväksi virtualisointitekniikoilla. Jotkut toiminnot kuitenkin ovat riittävän suuria, että niille on perusteltua varata oma laskennallinen kapasiteetti. Osa toiminnoista puolestaan on niin suuria, että tarvitsevat keskitetyn kapasiteettiratkaisun suoriutuakseen sille määritetystä kuormituksesta. (Rule & Dittner 2007, 26.)

Yleisesti ottaen virtualisointitekniologia muuttaa sekä laitteistotarpeen pääkeskustietokoneesta keski- tai alhaisen hintaluokan tietokoneisiin että joko keskitetystä tai hajautetusta infrastruktuurirakenteesta näiden yhdistelmäksi. Toisin sanoen suuret tietokoneet voidaan jakaa useaksi pienemmäksi yksiköksi, mikä mahdollistaa loogisen paikallisen hajautuksen samalla, kun hyödyntää laitteiston fyysistä keskittämistä. Eräs virtualisoinnin huomattavimmista eduista on näin ollen fyysisten palvelimien määrän vähentyminen ja olemassa olevien palvelimien käyttöasteen kasvattaminen. (Rule & Dittner 2007, 32.)

Muita virtualisoinnin etuja ovat esimerkiksi yhteensovittaminen (consolidation), luotettavuus (reliability) ja turvallisuus (security). Näitä virtualisoinnin etuja voidaan hyödyntää infrastruktuurin strategioinnissa, kuten esimerkiksi teknologian uudistamisessa, liiketoiminnan jatkuvuuden turvaamisessa ja odottamattomasta tapahtumasta palautumisessa, virtuaalisissa työ-

asemissa sekä testilaboratorioissa ja ohjelmistomäärittelyjen hallinnassa. Virtualisointia hyödynnetään kasvavassa määrin myös esimerkiksi tiedon varastoinnissa, verkkoteknologiassa sekä sovellusteknologioissa. (Rule & Dittner 2007, 26, 32, 33.)

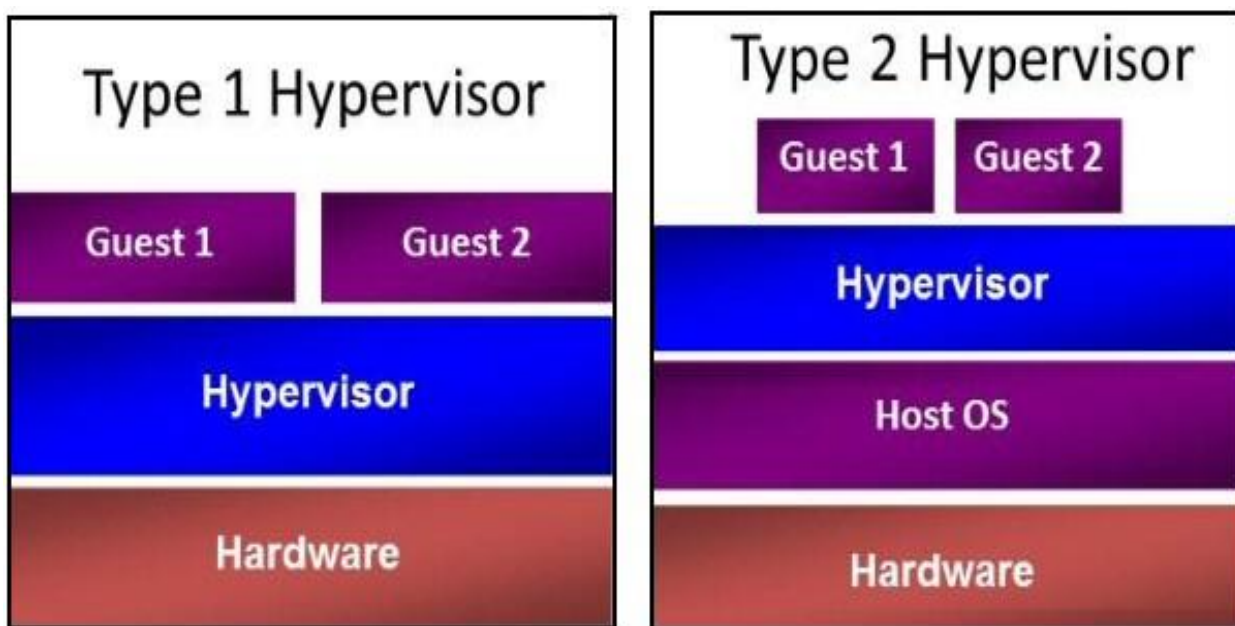
Yritykset ovat valmiita investoimaan palveluidensa luotettavuuteen varmistaakseen liiketoiminnan kannalta kriittisten sovellusten saatavuuden, ja usein tekevätkin niin. Sijoittamalla palveluiden sekä ohjelmiston kahdentamiseen varmistetaan palveluiden jatkuva saatavuus ja liiketoiminnan kyky toimia. Palveluiden kahdentaminen on tunnetusti kallista. Virtualisoinnin avulla palvelut voidaan eriyttää yhteen tai useampaan osaan ja näin varmistaa palvelun saatavuus pienemmillä kustannuksilla. (Rule & Dittner 2007, 14–15.)

Palveluiden eriyttäminen virtualisoinnin avulla myös vähentää koko järjestelmän alttiutta tietoturvahkalle. Mikäli yksi tai useampi tietokoneen osio tulee murretuksi, sen pääsy muuhun järjestelmään voidaan eriyttämisen avulla estää. Tämän lisäksi osion toimintakykyä voidaan supistaa heikentämällä osion kapasiteettia, esimerkiksi vähentämällä prosessorin tehoa, saatavuutta kovalevyyn tai pysäyttämällä järjestelmä kokonaan. Kyseinen toiminto olisi vaikeaa, ellei mahdotonta, tehdä, mikäli murrettu osio toimisi suoraan fyysisen laitteiston päällä. (Rule & Dittner 2007, 15–16.)

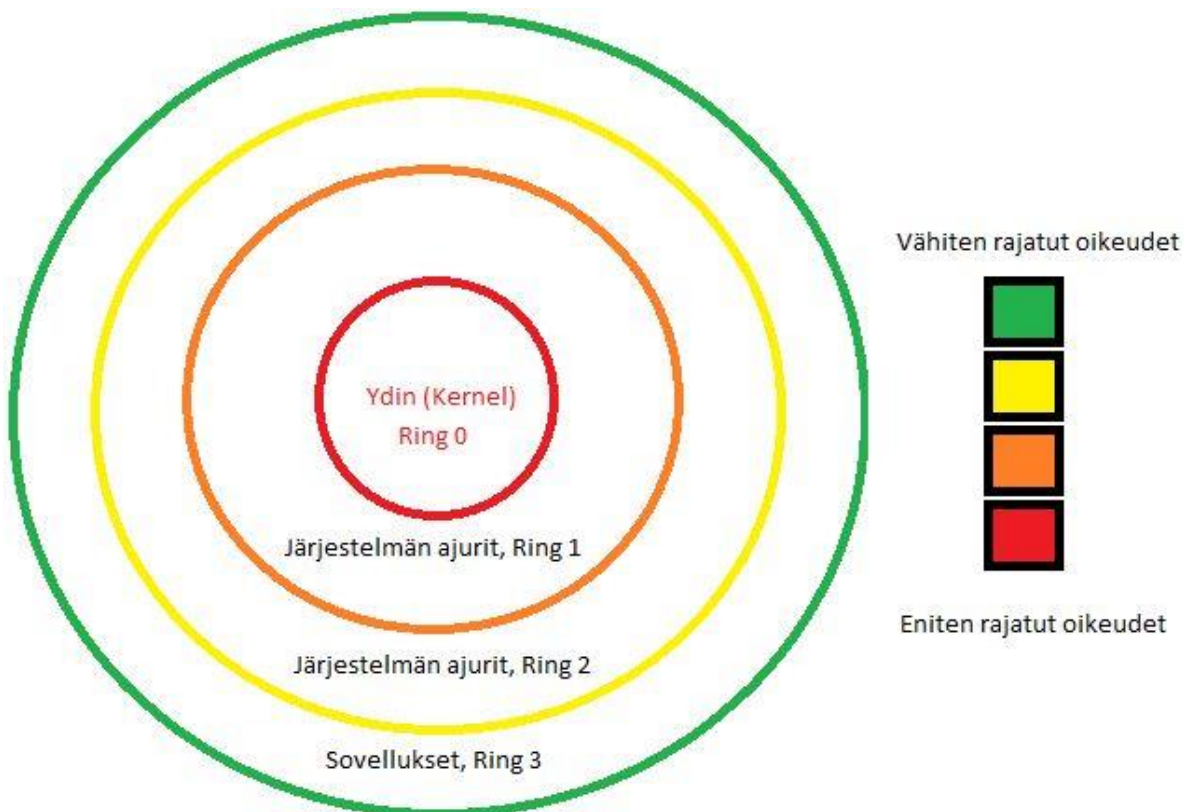
3 VIRTUALISOINTIALUSTAT

Tyyppin 1 virtualisointialustoina vertailussa mukana ovat VMWare ESXi, Microsoft Hyper-V, Linux KVM sekä Citrix XenServer. Lisäksi tyyppin 2 virtualisointialustoista otetaan vertailuun mukaan Oracle Virtualbox.

3.1 Virtualisointialustojen tyypit



Kuva 1. Virtualisointialustojen tyypit.



Kuva 2. Virtualisoinnin ja suojausten kerrokset (Rings).

Tyyppin 1 virtualisointialusta (VMM, Host) toimii suoraan laitteiston päällä (Ring 0) samaan tapaan kuin käyttöjärjestelmä (ks. kuva 2). Virtualisoidut tietokoneet (Guest) toimivat sitten VMM:n päällä (Ring 1), kerrosta ylempänä laitteiston rautatasoa. Tämä virtualisointitekniikka mahdollistaa todellisen virtuaalitietokoneen eriyttämisen omaksi kokonaisuudeksi laitteiston "raudasta" (hardware). (Rule & Dittner 2007, 18, 32.)

Tyyppin 2 virtualisointialusta (VMM, Host) puolestaan toimii käyttöjärjestelmän päällä (Ring 3). Virtualisoidut tietokoneet (Guest) toimivat tyyppin 2 virtualisointialustan päällä. Koska x86 – arkkitehtuurissa ei ole Ring 3 yläpuolelle määritettyä kerrosta, tyyppin 2 virtualisoidut tietokoneet toimivat mahdollisimman kaukana fyysisestä "raudasta" (hardware). Vaikutukset heijastuvat suorituskykyyn, koska käyttöjärjestelmän pyynnön täytyy kulkea jokaisen kerroksen läpi. (Rule & Dittner 2007, 18, 32.) Kyseinen seikka havaittiin myös osassa tämän opinnäytetyön tulosten osa-alueista.

3.2 VMWare ESXi

Perustettu 1998 oli ensimmäinen yritys, joka tarjosi kaupallista virtualisointiohjelmistoa x86:lle, ja nykyisin tarjoaa kattavan virtualisointiratkaisujen tuotevalikoiman x86-pohjaisille palvelimille ja työasemille. VMWarella on tyyppin 1 – virtualisointituote ESX Server. Yhdessä VMWare ESX Serverin kanssa Hypervisor asemoituu virtualisoidun työaseman ja laitteiston väliin abstraktina kerroksena. VMWare Workstationin kanssa Hypervisor toimii isäntätietokoneena kuten Windowsiin tai Linuxiin, asennettuna sovelluksena. VMWare mahdollistaa muokkaamattoman virtualisoidun käyttöjärjestelmän suorittamisen kääntämällä lennosta x86 komennot, joita ei voida virtualisoida. VMWare ESX Server ja Workstation ovat kaupallisia tuotteita. On hyvä huomioida, että VMWare voi suorittaa virtualisoituja työasemia paravirtualisoituna käyttäen WMI:tä. (Matthews, Dow, Deshane, Hu, Bongio, Wilbur & Johnson 2008, 33.)

3.3 Microsoft Hyper-V

Kaupallisille markkinoille myöhään tulleen tuotteenä voidaan pitää Hyper-V:tä. Hyper-V virtualisointialusta on saatavilla sekä itsenäisenä tuotteena, että Windows Server ominaisuutena (Alkaen versiosta 2008). Hyper-V kääntää virtualisoidun tietokoneen kernel moodin koodin ja todellisen x86 koodin käyttäjätason suorittavaksi koodiksi. Virtualisoidun tietokoneen (guest) koodi suoritetaan natiivisti. Virtualisoidun tietokoneen (guest) pyynnöt voidaan lukita joissakin olosuhteissa suorituskykyyn lisäämiseksi tai isäntätietokoneen integroimiseksi. (Matthews, Dow, Deshane, Hu, Bongio, Wilbur & Johnson 2008, 33.)

3.4 Linux KVM

Linux KVM (Kernel Virtual Machine) on toinen uusi tulokas virtualisoinnin "kentälle". KVM on tyyppin 1 – virtualisointialusta joka tuli esille 2.6.20 kernelin kehitytyön yhteydessä. KVM on sovitus Linux Kernelille, joka tekee Linuxista Hypervisorin "lisäosana" olevan KVM moduulin asennuksen yhteydessä. Jokainen KVM virtualisoitu tietokone (guest) suoritetaan "käyttäjä-

avaruudessa”. Tämä lähestymistapa saa näyttämän virtualisoidun tietokoneen (guest) tavallisena prosessina suorittavalle isäntätietokoneelle (host). Siksi KVM:llä on heikompi kyky eriyttää virtualisoidut tietokoneet toisistaan kuin muilla lähestymistavoilla. KVM:n avulla on mahdollista lähettää komennot jokaiselle virtualisoidulle tietokoneelle samalla tavalla kuin jokaisen eri käyttäjän prosessille voidaan tehdä. Saavuttaakseen kyseisen toiminnon, KVM on esitellyt uuden suoritustavan, joka on etäinen Linux käyttöjärjestelmän tavallisiin moodeihin (Kernel / Käyttäjä) nähden. Kyseinen moodi tunnetaan nimellä Guest mode. Guest moodilla on oma käyttäjä- ja kernel-moodi. (Matthews, Dow, Deshane, Hu, Bongio, Wilbur & Johnson 2008, 33.)

3.5 Citrix XenServer

Xen Hypervisor luotiin Cambridgen yliopiston tietokonelaboratoriossa osana Xenoserver-projektia vuonna 2001. Xenoserver-projektin tavoitteena oli mahdollistaa koodia suorittavia alustojen saatavuus ympäri maailmaa. Laitteiston laaja hyötykäyttö haluttiin varmistaa, ja tähän tarkoitukseen sopivin vaihtoehto oli Hypervisor, joka mahdollistaa usean yhtäaikaisen x86-arkkitehtuurin palvelimen. Xenoserver-projektin yhteydessä luotiin Xen ytimeksi jokaiselle Xenoserver-noodille. (Matthews, Dow, Deshane, Hu, Bongio, Wilbur & Johnson 2008, 30–31.)

3.6 Oracle Virtualbox

Oracle Virtualbox on alustariippumaton virtualisointiohjelmisto, joka mahdollistaa usean käyttöjärjestelmän suorittamisen samanaikaisesti. Virtualbox soveltuu esimerkiksi testaamiseen, kehittämiseen, esittelyyn sekä ohjelmistojen julkaisemiseen usealle alustalle yhden tietokoneen avulla. Virtualbox luotiin vastaamaan markkinoiden tarpeeseen. Virtualbox on kevyt ja helppokäyttöinen, suorituskykyinen virtualisointialusta ja se tukee laajaa skaalaa käyttöjärjestelmiä. (<http://www.oracle.com/us/tech-nologies/virtualization/oracle-vm-virtualbox-overview-2981353.pdf>)

4 TESTIYMPÄRISTÖ JA TESTIEN SUORITTAMINEN

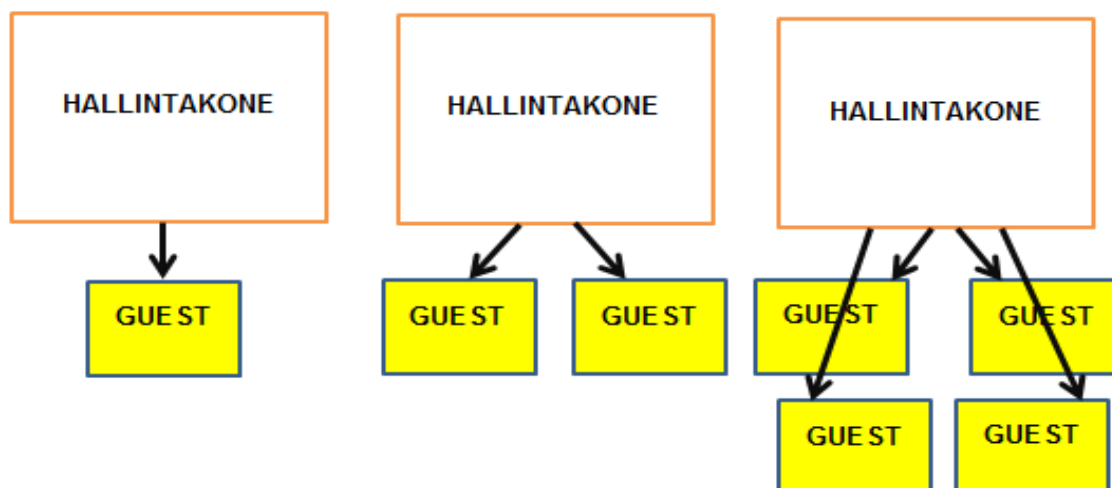
Testiympäristöä varten oli varattu kaksi tietokonetta, joista suorituskykyisempi tietokone valittiin testien virtualisointialustaksi ja toinen virtuaalietokoneiden hallintaa sekä tulosten tallennusta varten. Tietokoneista suorituskykyisempi valikoitui virtualisointialustaksi, koska suurempi keskusmuisti mahdollistaa useamman yhtäaikaisen virtuaalikoneen.

Alkuperäisenä suunnitelmana oli suorittaa vertailu useamman testiohjelmiston / testiympäristön yhdistelmänä (mm. IOBlazer, SysBench, Phoronix test suite). Tarkempi perehtyminen eri testausohjelmistoihin osoitti, että testaus kannattaa suorittaa yhden testausohjelmistokokonaisuuden alla, jotta tulokset ovat keskenään yhdenmukaisia. Testiohjelmistojen kokonaisuudeksi valittiin Phoronix test suite, joka sisältää useita valmiita testiohjelmistoja itsessään. Kyseiseen testiympäristöön on mahdollista lisätä omia testejä ja suorittaa kyseiset testit normaalisti testiympäristön muiden ennaltamääritettyjen testien ohella.

4.1 Testiympäristö

Testiympäristöä varten oli varattu kaksi tietokonetta. Toista tietokonetta käytettiin virtualisointialustatietokoneena (host). Siihen asennettiin kunkin virtualisointialustan ohjelmisto sekä virtuaalietokoneet (guest) kyseisen virtualisointialustan ohjelmiston ”päälle”.

Toista tietokonetta käytettiin hallintatietokoneena. Tätä tietokonetta käytettiin virtuaalisten tietokoneiden hallintaan SSH-yhteydellä ja 2/4 – virtuaalietokoneen yhdistelmää ClusterSSH – ohjelmistolla. ClusterSSH – ohjelmisto mahdollistaa yhden komennon syöttämisen usealle yhteydessä olevalle tietokoneelle samanaikaisesti (ks. kuva 3). Hallintatietokoneelle asennettiin Screen-apuohjelmisto SSH- / ClusterSSH-yhteyden tukemiseksi. Screen-apuohjelmisto varmisti testausten suorittamisen pitämällä yhteyden auki mahdollisesta terminaalin kaatumisesta huolimatta.



Kuva 3. Virtuaalietokoneiden hallinta-asetelma.

Jokaiselle virtualisoidulle loogiselle tietokoneelle (guest) allokoitiin 16 Gb muistia sekä 4 prosessorin säiettä. Virtuaalisten tietokoneiden käyttöjärjestelmäksi asennetaan CentOS 7. Koska XenServer sekä Hyper-V eivät suostuneet käynnistämään neljää yhtäaikaista virtuaalitietokoneetta, fyysisen (host) tietokoneen keskusmuistin määrä lisättiin 128 Gb:iin. ESXi, KVM sekä Virtualbox alustoilla fyysisessä (host) tietokoneessa oli 64Gb keskusmuistia. (Ks. taulukko 1)

Proessori	Xeon E5-1620 @ 3.6 GHz
Emolevy	D3348-A2x (Intel® C612 Chipset)
Muisti	64 Gb @ 1600 MHz (128 Gb XenServer / Hyper-V)
Näytönohjain	Geforce 770 GTX
Kovalevy	Toshiba 256 Gb SSD

Taulukko 1. Fyysisen tietokoneen (host) tekniset tiedot.

4.2 Testiohjelman Phoronix-Test-Suite

Alkuperäisenä suunnitelmana oli käyttää IOBlazer, SysBench, Phoronix Test Suite sekä wPrime testiohjelmistoina. Hyvin pian ohjelmistoihin perehtymisen jälkeen havaittiin, että testiohjelmistojen tiedostomuodot eivät ole yhteensopivia keskenään. Toisin sanoen kaikkien testiohjelmistojen käyttäminen olisi edellyttänyt usean apuohjelman luomisen tulosten yhtenäistämiseksi. Apuohjelmistojen luominen ei ollut tarkoituksenmukaista tämän opinnäytetyön tekemisen näkökulmasta. Tästä johtuen testiohjelmistoksi valikoitui testiohjelmisto Phoronix-test-suite.

Phoronix-test-suite on kattavin testaukseen ja tiettyjen kriteerien mukaiseseen vertailuun tarkoitettu testausalusta. Phoronix-test-suite mahdollistaa testauksen automatisoinnin; testin asennuksen, suorittamisen sekä raportoinnin. Kaikki Phoronix-test-suiteen testit ovat tarkoitettu helposti toistettaviksi, helppokäyttöiseksi sekä tukevat testauksen automatisointia. Phoronix-Test-Suite antaa tulokset xml – tiedostoina.

Phoronix-test-suiteessa on valmiina yli 200 testiohjelmaa, jotka ovat avoimesti saatavilla Open-Benchmarking.org integraation kautta. Tutkimukseen valikoitui muistin, suorittimen, kovalevyn, verkon sekä järjestelmän suorituskykyä mittaavat testiohjelmat (ks. taulukko 2). Kyseiset testiohjelmat mittaavat kokonaisuudessaan oman testauskohteen osa-alueita, ja niitä käytetään yleisimmin tietokoneen suorituskyvyn mittauksissa.

Prossessorin suorituskykytesteistä esimerkiksi HMMER mittaa sekvenssihaun kestoa tietokannoista sekunteina.

Muistin suorituskykytesteistä esimerkiksi ramspeed sisältää 4 osiota: Copy, Scale, Add ja Triad. Copy kopioi tiedon yhdestä paikasta toiseen ($A = B$). Scale muokkaa tietoa vakiolla ennen siirtoa toiseen paikkaan ($A = m*B$). Add yhdistää kaksi muistipaikan tietoa keskenään ennen

siirtoa ($A = B + C$). Triad on yhdistelmä Scalesta ja Addista ($A = m*B + C$). Tulos on Mb/s muodossa.

Kovalevyn suorituskykytesteistä esimerkiksi compilebench luo, rakentaa, päivittää ja lukee kernel puun. Tulos on Mb/s muodossa.

Network-loopback testaa verkkokortin loopback osoitetta siirtämällä 10Gb dataa, ja mittaa ajan sekunneissa.

Toimeksiantajan pyynnöstä yhtenä testiohjelmanä käytettiin R:llä tehtyä valmista ohjelmistokokonaisuutta, jolle tehtiin paikallinen testiprofiili Phoronix-test-suiteen sisälle.

Testiohjelma	Kohde
Ramspeed	Muisti
Stream	Muisti
Cachebench	Muisti
Hmmer	Prosessori
Mafft	Prosessori
MrBayes	Prosessori
Sample-Program (Pi)	Prosessori
FS-Mark	Kovalevy
Compilebench	Kovalevy
Unpack-Linux	Kovalevy
Network-loopback	Verkko
R (valmis script)	Järjestelmä

Taulukko 2. Testiohjelma sekä testattava osa.

4.3 Kaappausmedian luonti

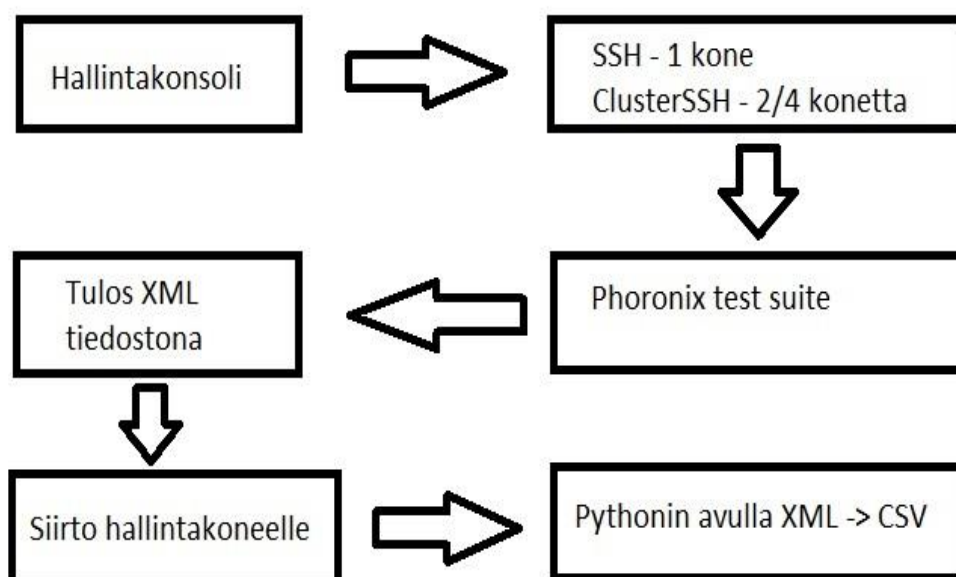
Alkuperäisenä suunnitelmana oli kaapata natiivi käyttöjärjestelmä testiohjelmistoinen, mitä käytettäisiin vertauskohteena virtualisointialustojen tuloksien vertailussa. Kaappausmedian luonti ei kuitenkaan toiminut, eivätkä virtualisointialustojen tiedostomuodot olisi tukeneet yhtä kaappausmediaa. Sen sijaan kaikille virtualisointialustoille asennettiin CentOS 7.0 – käyttöjärjestelmä ”puhtaana” asennuksena. Tämän jälkeen asennettiin Phoronix-Test-Suite sekä Phoronix-test-suiteen vaatimat käyttöjärjestelmän lisäosat jokaiselle virtualisointialustalle ”vakioidulla” komennoilla.

4.4 Testien suorittaminen

Phoronix-test-suiteen valmiit ohjelmistot sekä R-osuus suoritettiin kokonaisuudessa batch-ajona. Batch-ajo eli eräajo tarkoittaa testausten suorittamista ennalta määritettyjen arvojen mukaisesti, sekä tuloksen tarkastelua. Pääsääntöisesti testaukset sujuivat hyvin. Ennen varsi-

naisia testiajoja piti kuitenkin ratkaista seuraavat virtualisointialustoissa olleet haasteet: i) HyperV – virtualisointialusta ymmärsi käytetyn SSD-kovalevyn vasta valmistajan omalla ajurilla, ii) XenServer ei tukenut suoraa dynaamista muistinkäsittelyä, ja iii) ESXi asennuslevyyn piti asentaa verkkokortin ajuri lisänä, jotta asennus meni läpi.

VirtualBox alustalla testiohjelmiston suoritus neljän virtuaalisen tietokoneen yhdistelmällä ei onnistunut. Kyseinen testi suoritettiin kaksi kertaa, joista kummallakin kerralla yksi virtuaalisesta tietokoneesta jäi ilman tulosta. Tämän vuoksi VirtualBoxin neljän virtuaalisen tietokoneen yhdistelmän tulokset eivät ole vertailukelpoisia, eivätkä näin ollen ole mukana tulosten analysoinnissa. Sen sijaan yhden sekä kahden virtuaalisen tietokoneen testiajot onnistuivat VirtualBox alustalla ja ovat mukana vertailussa.



Kuva 4. Testiajojen prosessikaavio.

Phoronix-test-suitein batch-ajo suoritettiin tulosten saamista varten yhteensä 44 kertaa, jokainen testivariaatio suoritettiin kolme (3) kertaa ja niistä laskettiin keskiarvo. Tämän jälkeen Phoronix-Test-Suitein batch-ajojen tulokset siirrettiin hallintatietokoneelle. Koska Phoronix-test-suite antaa tulokset XML tiedostona, ja toimeksiantajan pyynnöstä tulokset haluttiin luettavassa muodossa, luotiin koodi, joka muuttaa XML tuloksen CSV muotoon Pythonin avulla. (Ks. kuva 4)

Testiohjelmiston suorituskomento annettiin kaikille virtuaalietokoneille yhtä aikaa. Kaikki virtuaalietokoneet suorittivat testit samassa lineaarisesti etenevässä järjestyksessä, mutta eivät samanaikaisessa syklistä. Tämä vaikuttaa testitulosten vertailukelpoisuuteen.

5 TULOKSET

Tämän opinnäytetyön tarkoituksena oli verrata natiivin käyttöjärjestelmän ja virtualisointialustojen suorituskykyeroja. Suorituskykyeroa mitattiin Phoronix-Test-Suite testiohjelmiston avulla. Testeissä mitattiin muistin, prosessorin, kovalevyn, verkon sekä koko järjestelmän nopeutta. Tulokset luokiteltiin kunkin osa-alueen mukaisesti. Tulokset arvioitiin urheilussa käytetyllä järjestysperusteella, ja tasatilanteessa käytettiin suhteellista tilastointia arviointiasteikossa.

Opinnäytetyötä varten kootun testirympäristön tausta-ajatuksena oli sen havainnointi, missä rasitusasteessa fyysinen tietokone (host) pystyy suorittamaan testiohjelmiston kohtuullisen ajan sisällä (ks. kuva 3, Virtuaalitetokoneiden hallinta-asetelma). Näin ollen tulokset eriytettiin kolmeksi eri kategoriaksi: yhden-, kahden- sekä neljän virtuaalitetokoneen perusteella. Natiivitetokone jätettiin mukaan ainoastaan yhden virtuaalitetokoneen suorituskykytaulukon, koska natiivitetokone on vertailukelpoinen ainoastaan yhden virtuaalitetokoneen kategorias-
sa.

Huomioitavaa suorituskykytulosten arvioinnissa on i) testiajojen suhteellisen pieni määrä, sekä ii) testit suoritettiin samassa lineaarisesti etenevässä järjestyksessä, mutta eivät samanaikaisessa syklissä.

5.1 Saadut tulokset

Tulokset esitetään kategorioittain. Kategorioina on virtuaalitetokoneiden lukumäärä. Yksittäiset testaukset yhden-, kahden- sekä neljän virtuaalitetokoneen osilta ovat kaavioina opinnäytetyön liitteinä (liite 1).

	1	2	3	4	5	6
Kovalevy	Natiivi	VirtualBox	KVM	Esx	XenServer	Hyper-V
Muisti	Natiivi	Esx	XenServer	VirtualBox	Hyper-V	KVM
Verkko	KVM	Natiivi	Esx	XenServer	Hyper-V	VirtualBox
Prosessori	Natiivi	Esx	XenServer	KVM	VirtualBox	Hyper-V
Järjestelmä	Natiivi	XenServer	Esx	KVM	Hyper-V	VirtualBox

Taulukko 3. Yhden virtuaalitetokoneen osa-aluekohtainen tulostaulukko.

Yksittäisten suorituskykytestien tasolla virtualisointialustat suoritettiin natiivitetokoneeseen nähden prosentuaalisesti laajasti vaihdellen. Yhden virtuaalitetokoneen kategoriassa kovalevytesteissä virtualisointialustat suoritettiin parhaimmillaan 225 % suhteessa natiivitetokoneen suorituskykyyn, ja heikoimmillaan 31 % virtualisointialustat suorituskykyyn verrattuna. Muistin osalta parhaimmillaan 121 % ja huonoimmillaan 54 %. Verkon testissä suorituskykyerot olivat parhaimmillaan 114 % ja huonoimmillaan 21 %. Prosessorin testissä suorituskykyerot olivat parhaimmillaan 103 % ja huonoimmillaan 15 %. Lähimpänä natiivitetokoneen suo-

rituskykyä virtualisointialustat olivat koko järjestelmää mittavassa R-testissä; parhaimmillaan 99 % ja huonoimmillaan 94 % natiiviin verrattuna.

Vaikka yksittäiset testitulokset natiivin ja virtualisointialustojen välillä vaihtelivat suuresti, kokonaisuutena tarkasteltuna natiivi suoriutui suurimmassa osassa suorituskykymittauksia parhaiten. Taulukkoon 3 on koottu tuloslista, jossa on natiivin ja virtualisointialustojen suorituskykyerot järjestettynä parhaasta huonoimpaan osa-alueittain. Virtualisointialustoja ei laitettu "paras – huonoin" järjestykseen, koska suorituskykyerot testatuilla osa-alueilla vaihtelevat suuresti.

	1	2	3	4	5
Kovalevy	KVM	VirtualBox	Esx	Hyper-V	XenServer
Muisti	VirtualBox	KVM	XenServer	Esx	Hyper-V
Verkko	KVM	XenServer	Esx	Hyper-V	VirtualBox
Prosessori	Esx	KVM	XenServer	VirtualBox	Hyper-V
Järjestelmä	KVM	Hyper-V	Esx	XenServer	VirtualBox

Taulukko 4. Kahden virtuaalitietokoneen osa-aluekohtainen tulostaulukko.

Kahden virtuaalitietokoneen kategoriassa kovalevytesteissä virtualisointialustat suoriutuivat parhaimmillaan 151 % suhteessa natiivitietokoneen suorituskykyyn, ja heikoimmillaan 32 % virtualisointialustat suorituskykyyn verrattuna. Muistin osalta parhaimmillaan 118 % ja huonoimmillaan 28 %. Verkon testissä suorituskykyerot olivat parhaimmillaan 117 % ja huonoimmillaan 21 %. Prosessorin testissä suorituskykyerot olivat parhaimmillaan 102 % ja huonoimmillaan 21 %. Kuten myös yhden virtuaalitietokoneen testissä, pienimmät erot suorituskyvyssä olivat koko järjestelmää mittaavalla R-testillä; parhaimmillaan 98 % ja huonoimmillaan 87 % natiiviin tietokoneen tulokseen verrattuna.

	1	2	3	4
Kovalevy	Esx	Hyper-V	XenServer	KVM
Muisti	KVM	XenServer	Esx	Hyper-V
Verkko	KVM	XenServer	Esx	Hyper-V
Prosessori	XenServer	KVM	Esx	Hyper-V
Järjestelmä	Hyper-V	Esx	KVM	XenServer

Taulukko 5. Neljän virtuaalitietokoneen osa-aluekohtainen tulostaulukko.

Neljän virtuaalitietokoneen kategoriassa kovalevytesteissä virtualisointialustat suoriutuivat parhaimmillaan 102 % suhteessa natiivitietokoneen suorituskykyyn, ja heikoimmillaan 11 % virtualisointialustat suorituskykyyn verrattuna. Muistin osalta parhaimmillaan 108 % ja huonoimmillaan 39 %. Verkon testissä suorituskykyerot olivat parhaimmillaan 92 % ja huonoimmillaan 49 %. Prosessorin testissä suorituskykyerot olivat parhaimmillaan 100 % ja huonoim-

millaan 20 %. Koko järjestelmää mittaavalla R-testillä suorituskykyero oli parhaimmillaan 94 % ja huonoimmillaan 68 % natiiviin tietokoneen tulokseen verrattuna.

6 YHTEENVETO

Opinnäytetyö toteutettiin toimeksiantona Itä-Suomen yliopistolle, bioinformatiikan keskukselle. Opinnäytetyössä verrattiin natiivin käyttöjärjestelmän ja virtualisointialustojen suorituskykyeroja Phoronix-test-suite -testiohjelmistolla. Opinnäytetyön toimeksiantajan pyynnöstä tyyppin 1 virtualisointialustoina vertailussa mukana olivat VMWare ESXi, Microsoft Hyper-V, Linux KVM sekä Citrix XenServer. Lisäksi tyyppin 2 virtualisointialustoista otettiin vertailuun mukaan Oracle Virtualbox.

Virtualisointialustojen mallinnointiin käytettiin toista testiympäristön tietokonetta. Jokaiselle virtualisoidulle loogiselle tietokoneelle (guest) allokoitiin 16 Gb muistia sekä 4 prosessorin säiettä. Virtuaalisten tietokoneiden käyttöjärjestelmäksi asennettiin CentOS 7. Virtuaalitetokoneiden lukumääräksi valikoitui toimeksiantajan pyynnöstä yhden-, kahden- sekä neljän yhtäaikaisen virtuaalitetokoneen yhdistelmä.

Toista testiympäristön tietokonetta käytettiin hallinnointitietokoneena, johon asennettua Screen-apuohjelmistoa käytettiin SSH- / ClusterSSH-yhteyden muodostamiseksi virtuaalitetokoneille. Opinnäytetyötä tehtäessä havaittiin, että käytetty Screen-apuohjelmisto piti yhteyden virtuaalitetokoneille avonaisena, vaikka etäyhteyden konsoli itsessään oli kaatunut. ClusterSSH – ohjelmisto puolestaan mahdollisti yhden komennon syöttämisen usealle yhteydessä olevalle tietokoneelle samanaikaisesti.

Phoronix-test-suiteen valmiit testiohjelmit sekä R-osuus suoritettiin kokonaisuudessa batch-ajona, ja testiohjelmisto suoritettiin kolme kertaa jokaisen virtuaalitetokoneyhdistelmän variantilla. Suorituskykytestit suoritettiin kolme kertaa, koska haluttiin lisätä tulosten luotettavuutta. Saadut tulokset tulokset muokattiin toimeksiantajan pyynnöstä Pythonin avulla yhdeksi csv-tiedostoksi, joka sisälsi kaikki testitulokset. Näin testitulokset ovat luettavassa muodossa.

Toimeksiantaja voi hyödyntää tässä opinnäytetyössä saatuja testituloksia valitessaan virtualisointialustaa heitä eniten hyödyttävän osa-alueen perusteella. Jatkokehitystä ajatellen hyötyä voisi olla myös tämän opinnäytetyön ulkopuolelle jääneen näytönohjaimen suorituskykytestauksella. Tämän lisäksi voisi ottaa huomioon myös ulkopuolelle jääneen VMWare Workstation virtualisointialustan, sekä mahdollisesti Docker – tekniikalla toteutetut virtualisointialustat.

LÄHTEET

Chisnall, David, 2008. The Definite Guide to the Xen Hypervisor. Boston, MA: Prentice Hall

Dittner, Rogier & Rule, David, 2007. Best Damn Server Virtualization Book Period. Burlington, MA. Syngress Publishing Inc.

Matthews, Jeanna N., Dow, Eli M., Deshane, Todd, Hu, Wenjin, Bongio, Jeremy, Wilbur, Patrick F. , Johnson, Brendan, 2008. Running Xen: Hands-On Guide To The Art of Virtualization. Boston, MA: Prentice Hall

Siiskonen, P. 2011. VMWare, ESX-Server, Hyper-V ja Virtuoggo virtualisointitekniikoiden vertailu. Kuopio: Itä-Suomen Yliopisto.

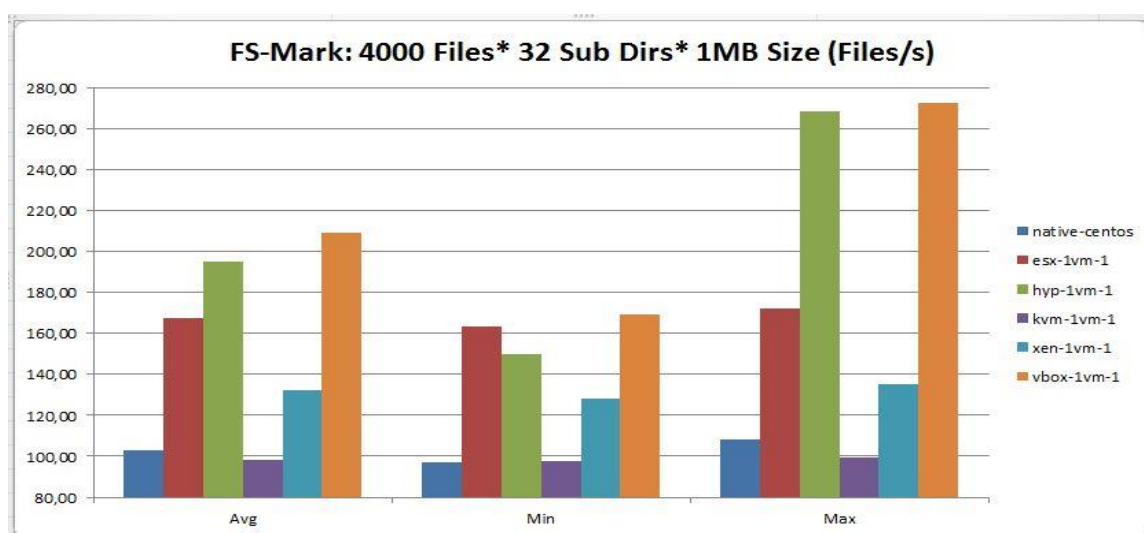
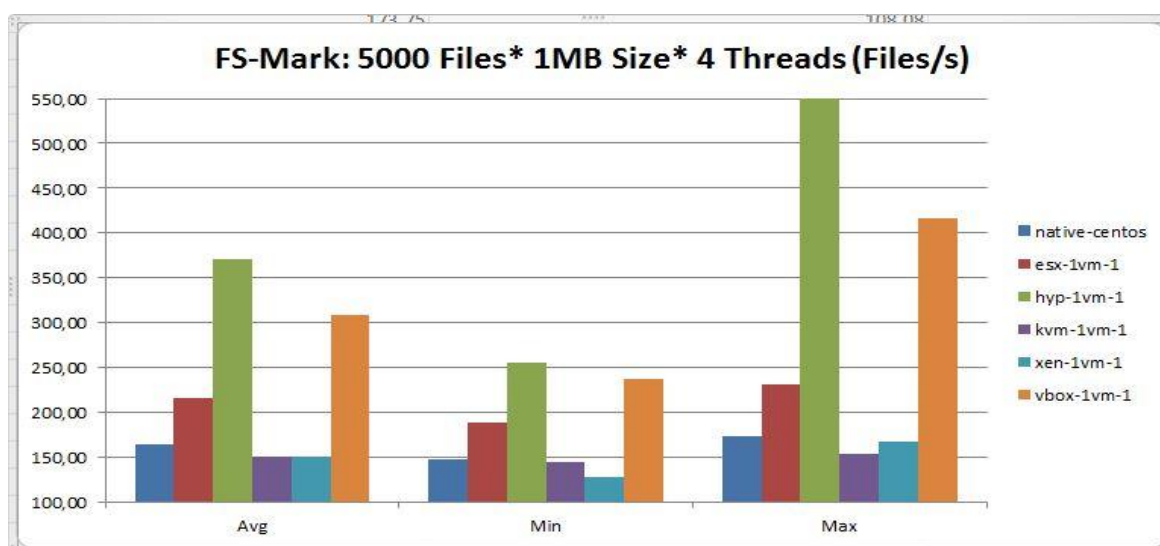
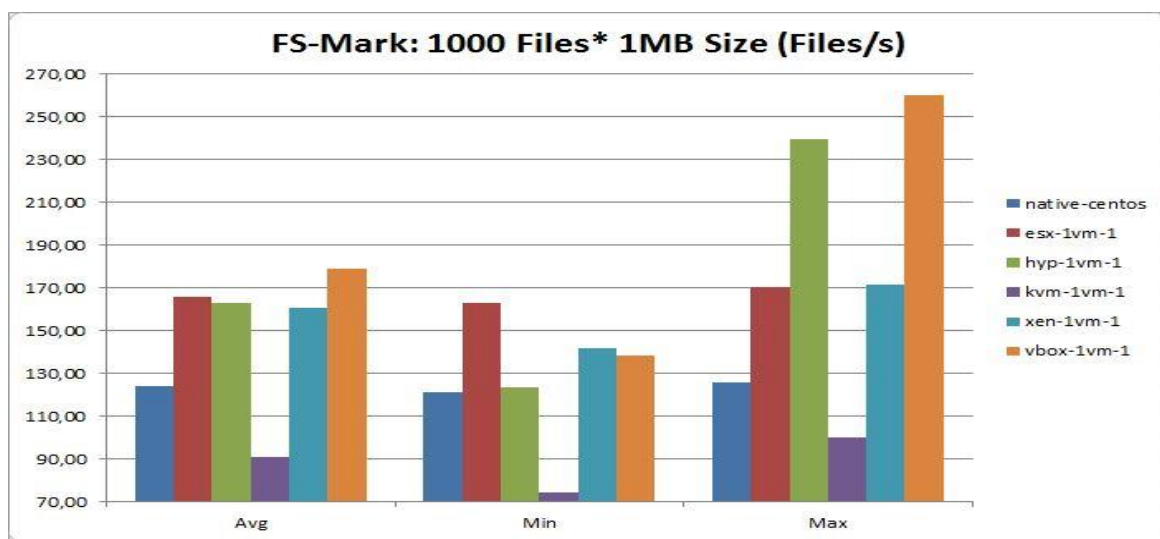
Tikkanen, M. 2008. Palvelimien virtualisointi. Kuopio: Kuopion Yliopisto.

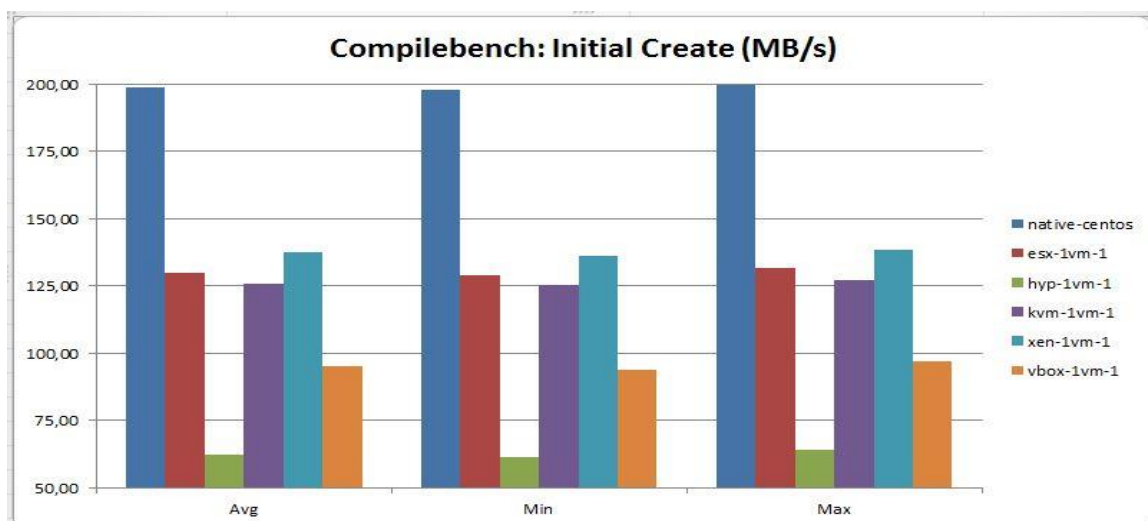
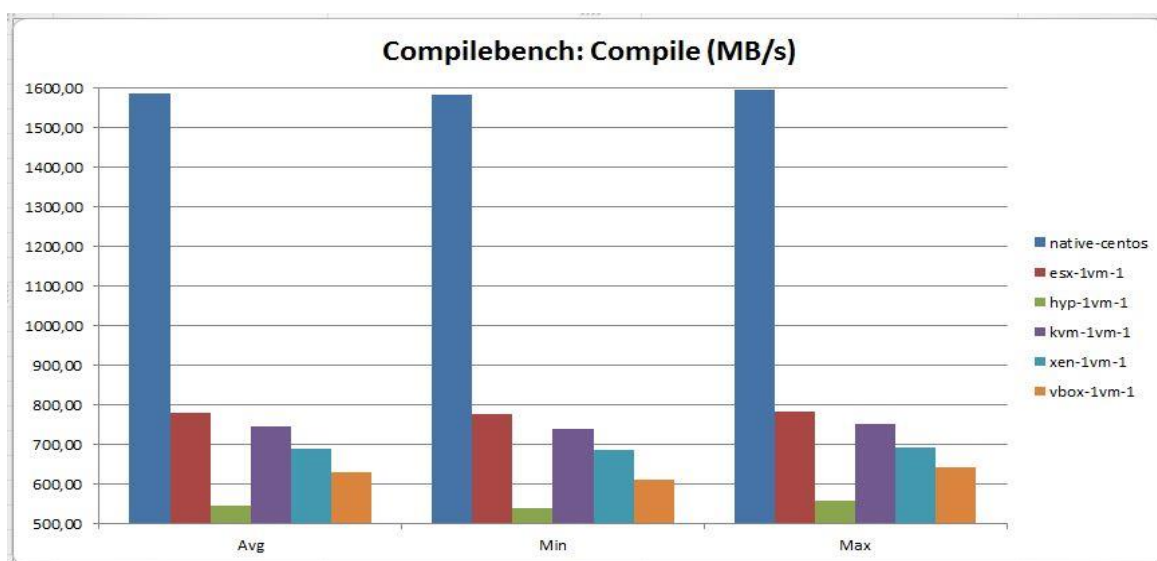
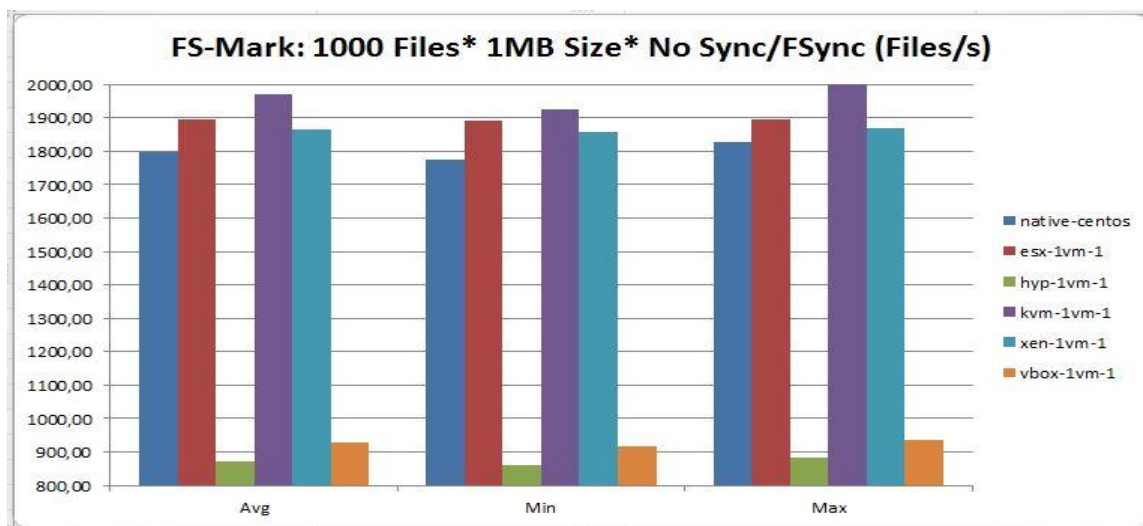
Oracle VM VirtualBox 5.0 Overview. An Oracle White Paper, March 2016.

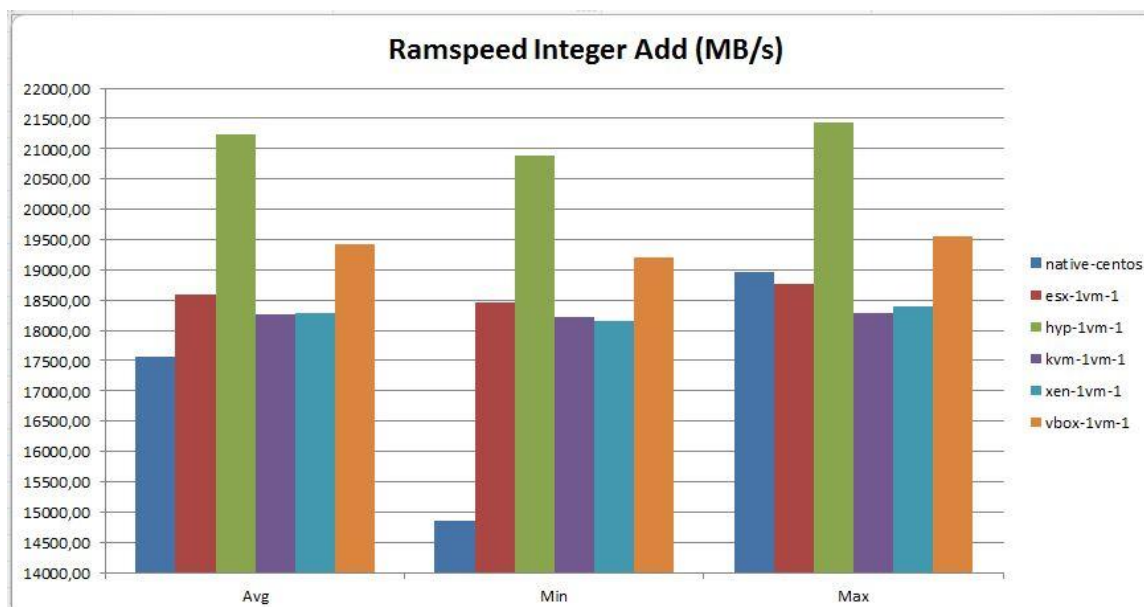
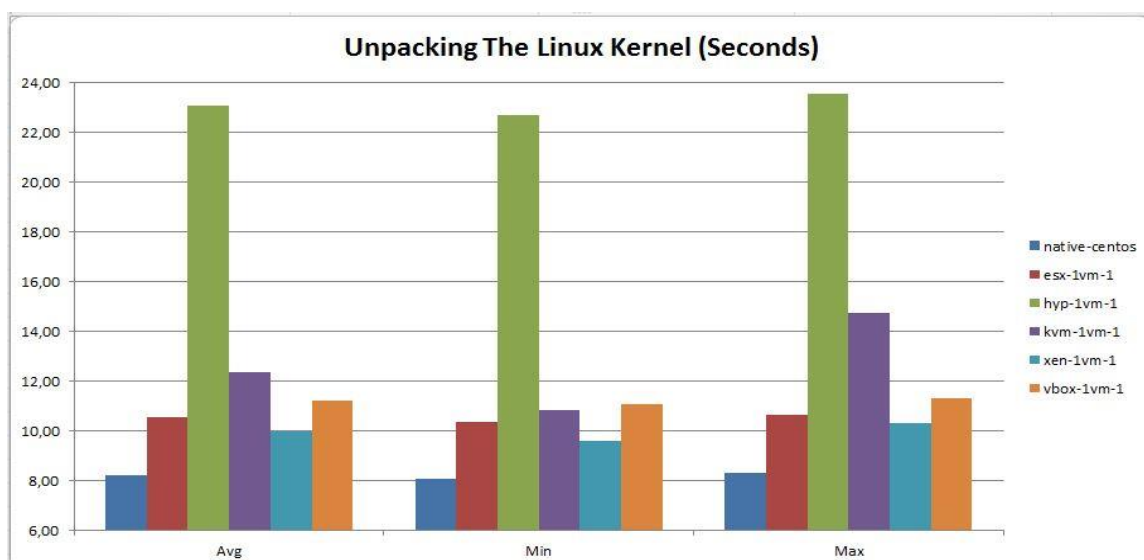
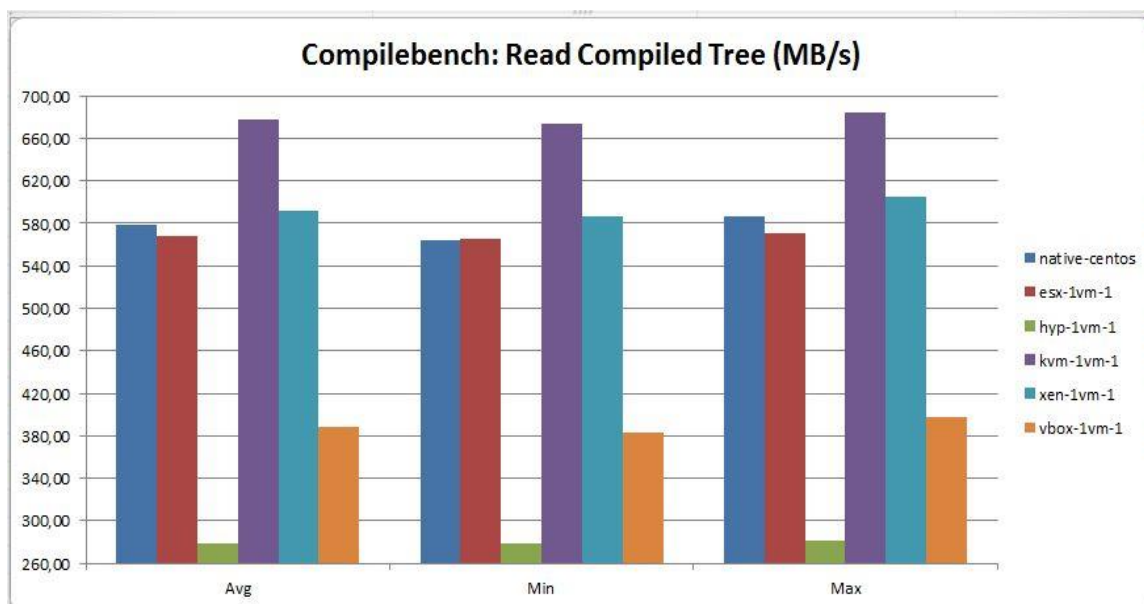
(<http://www.oracle.com/us/technologies/virtualization/oracle-vm-virtualbox-overview-2981353.pdf>), viitattu 20.5.2017.

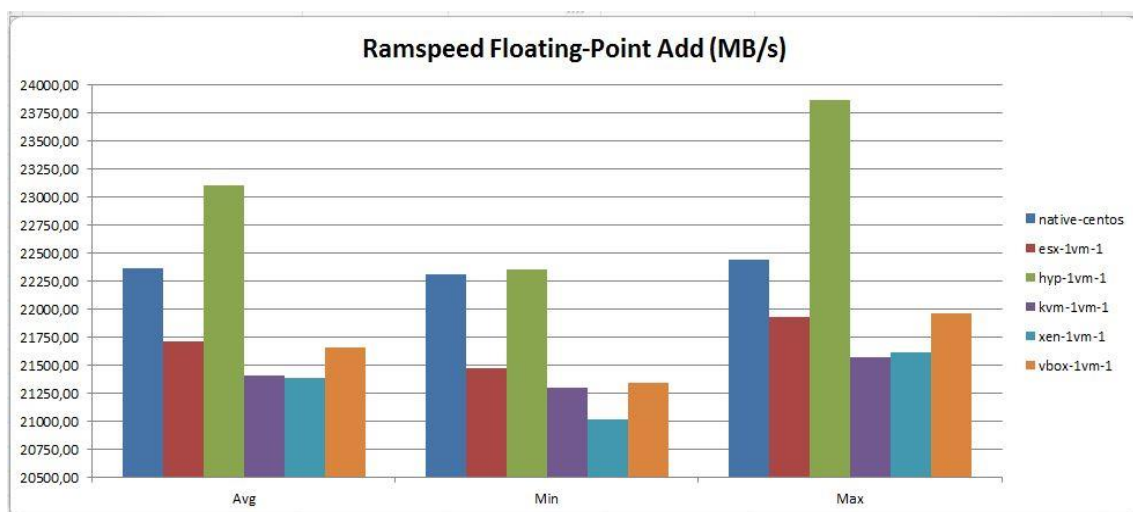
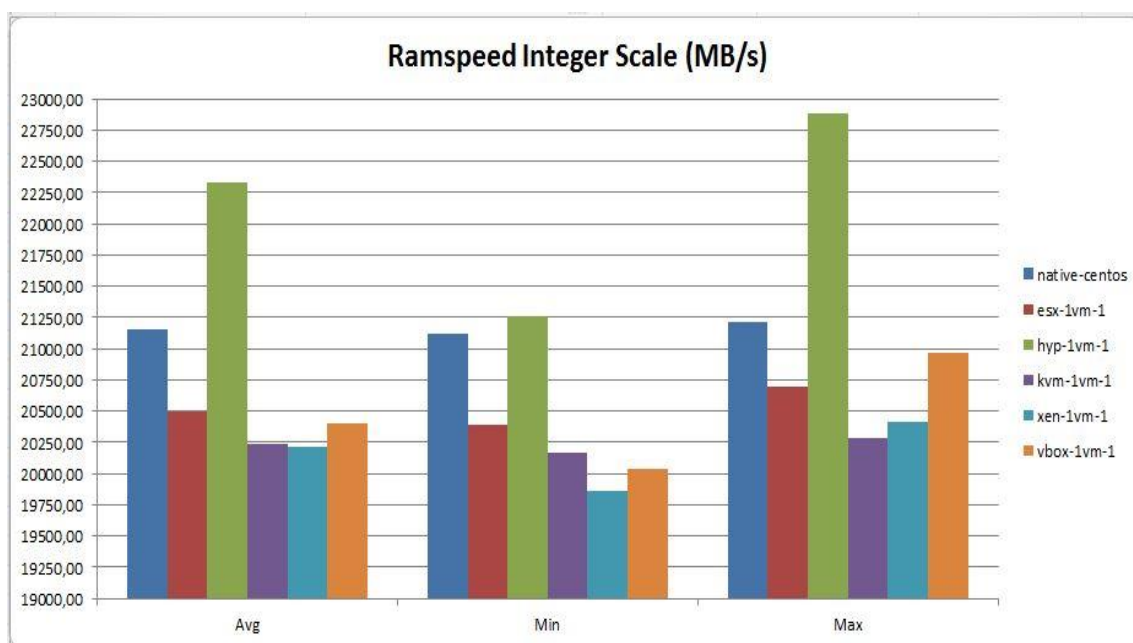
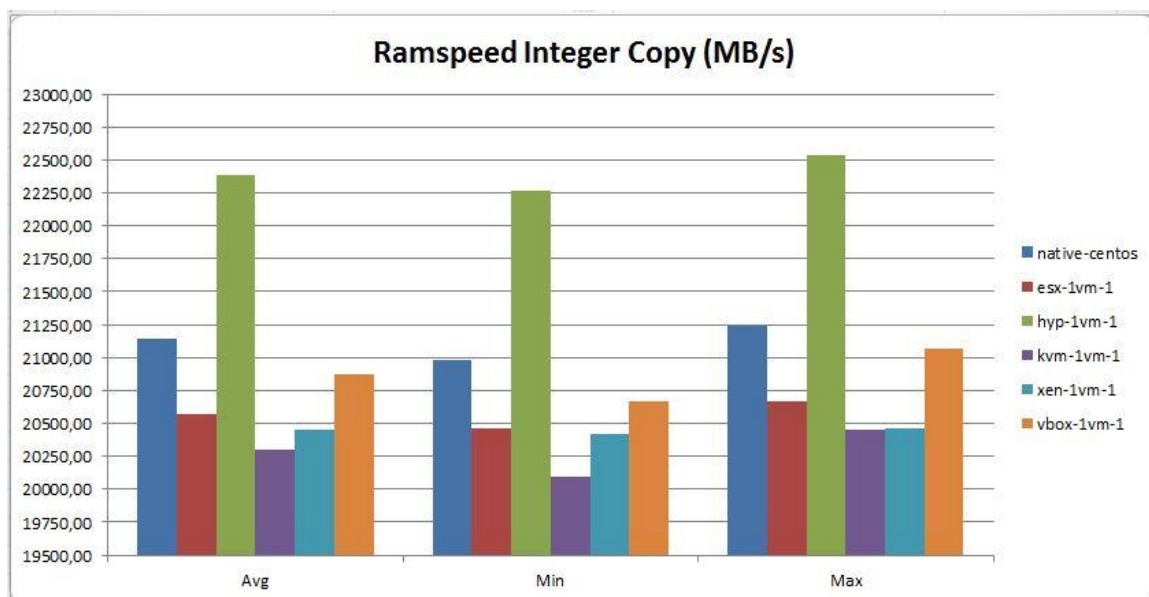
LIITE 1: SAADUT TULOKSET KAAVIOINA

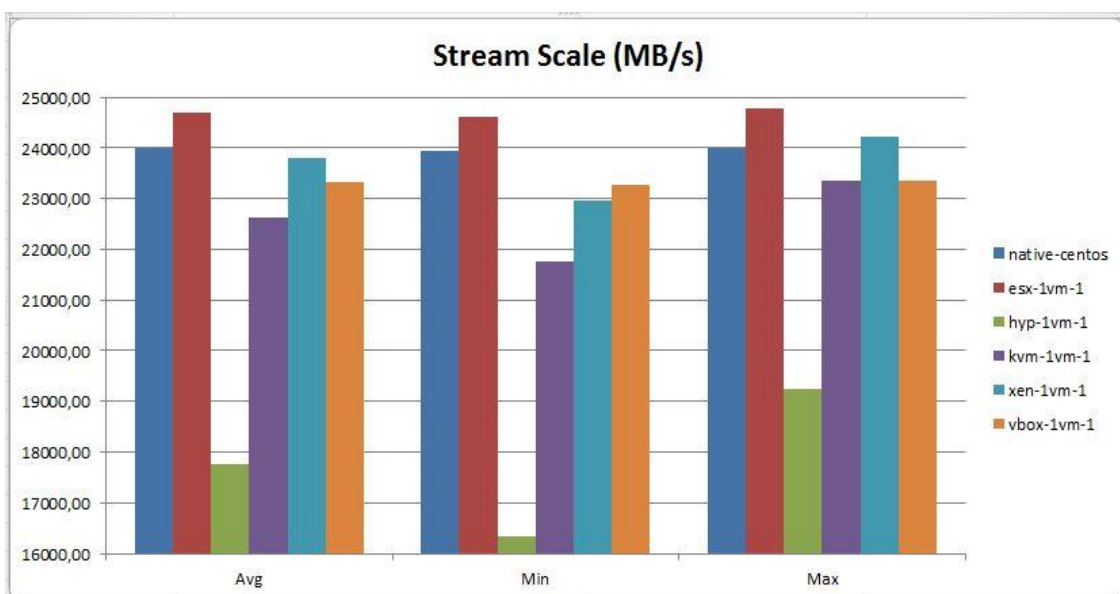
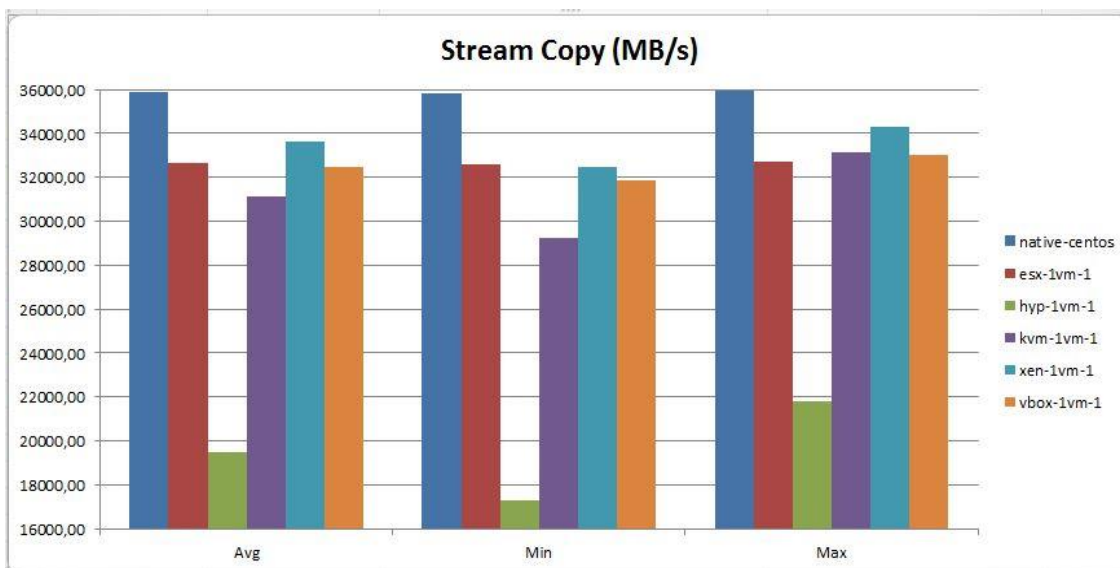
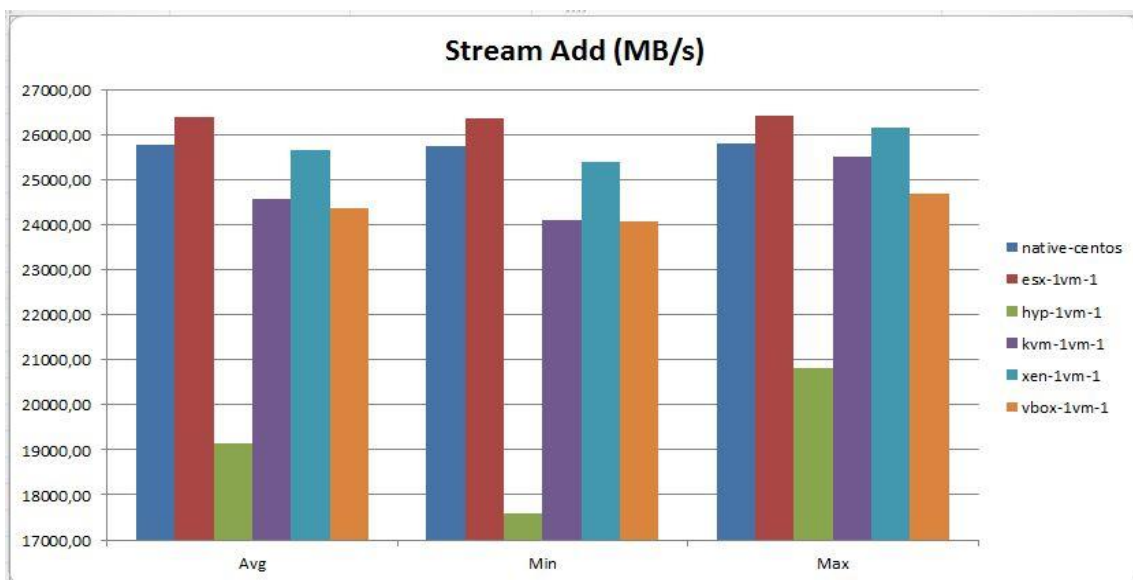
Liite 1.1. Yhden virtuaalitetokoneen tulokset kaavioina

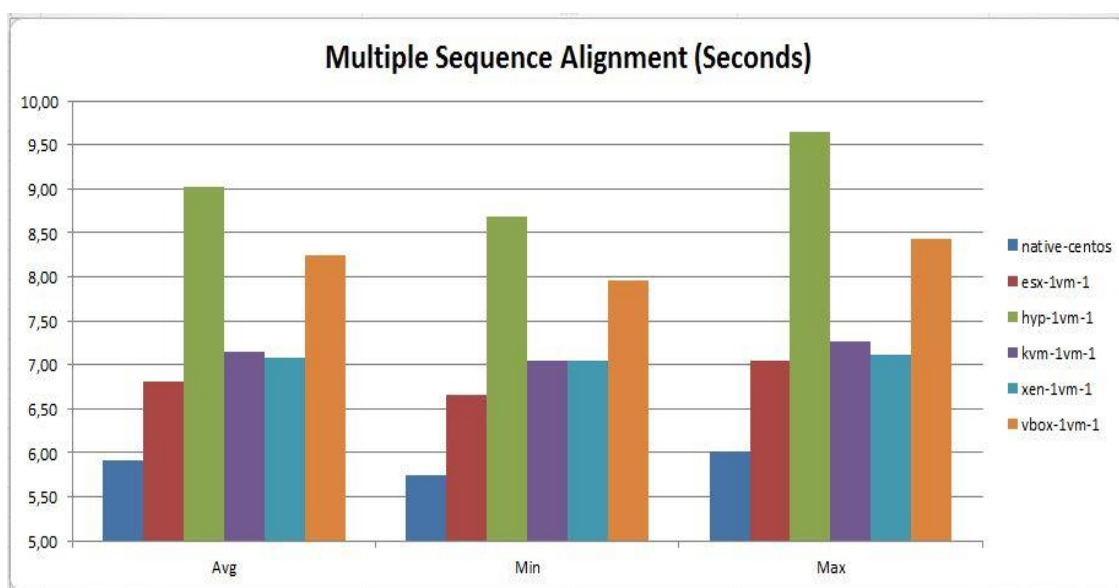
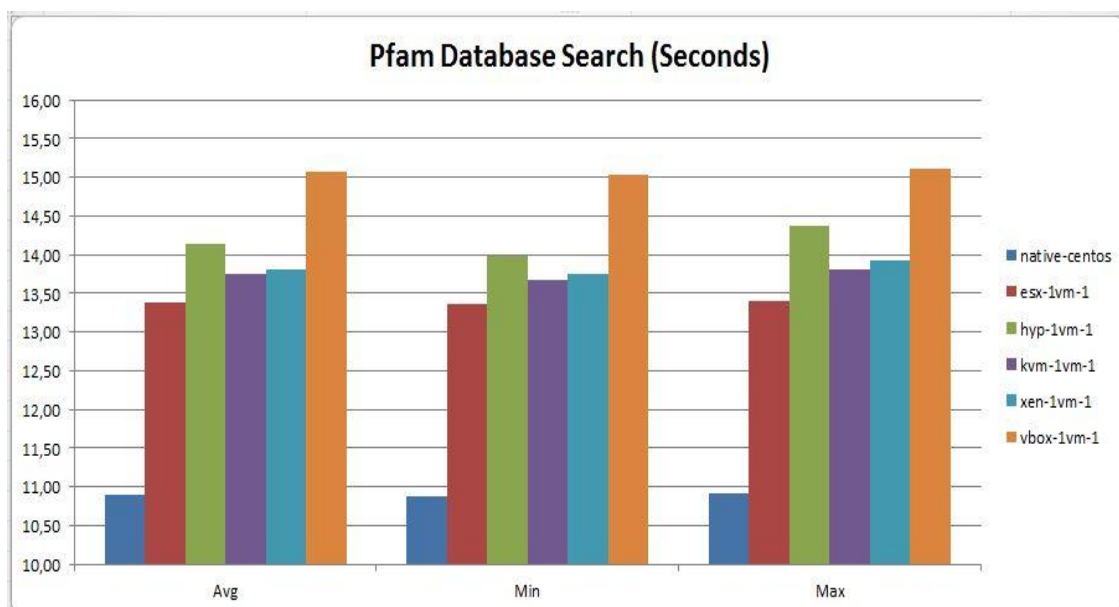
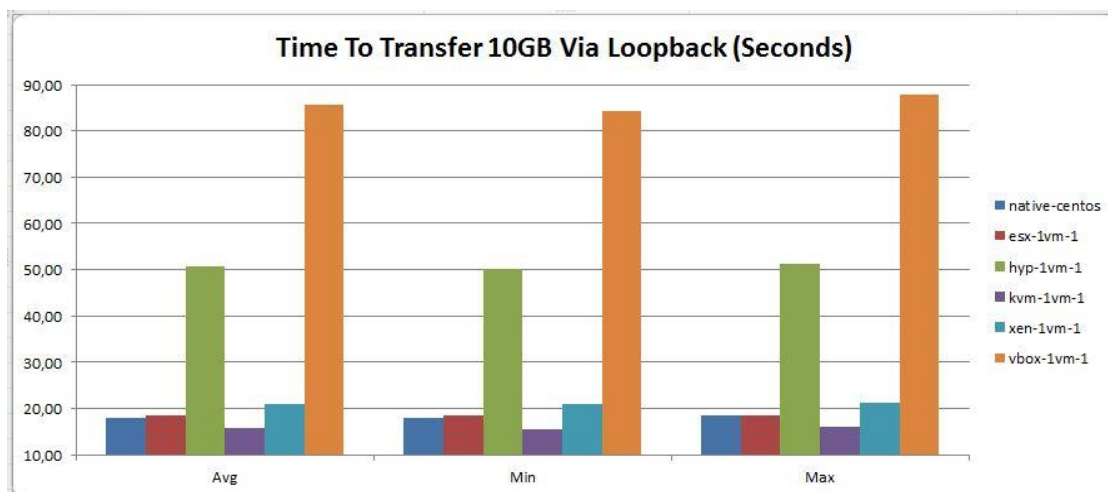


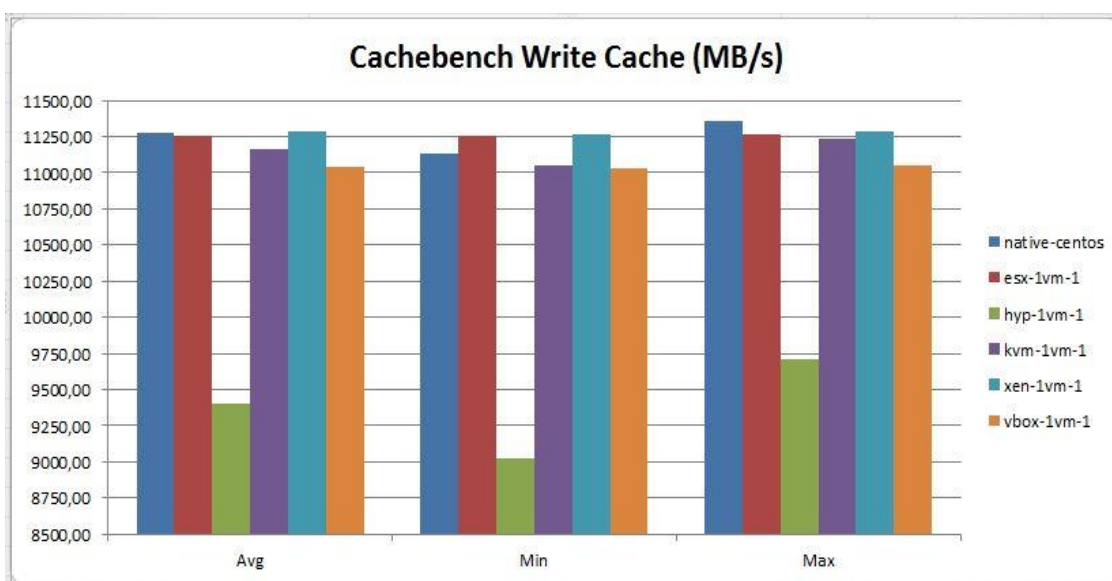
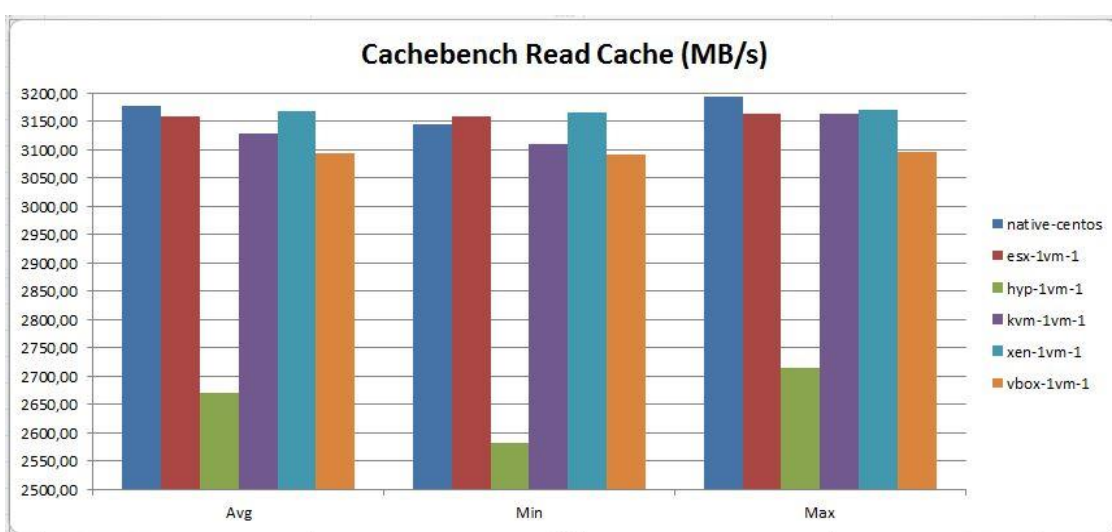
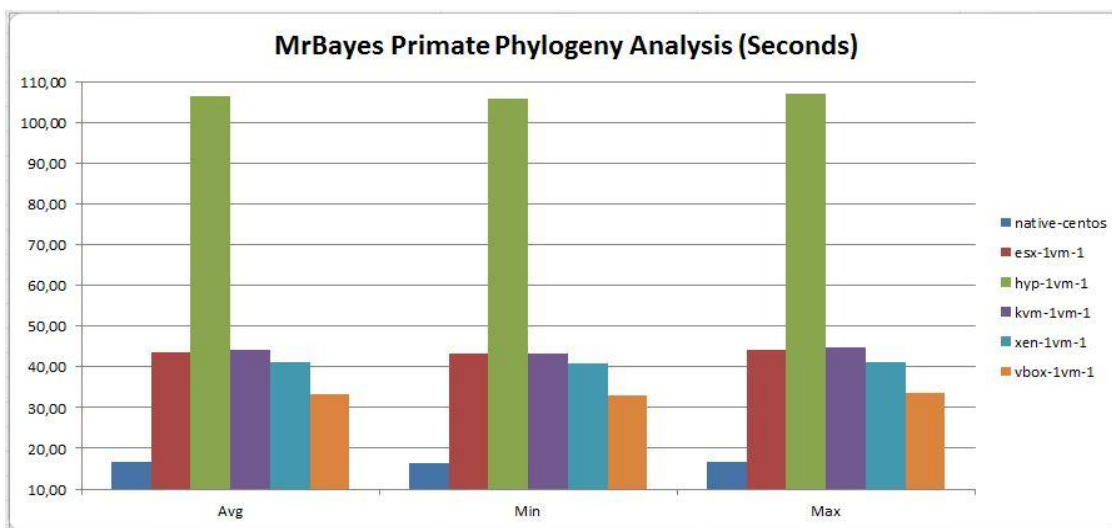


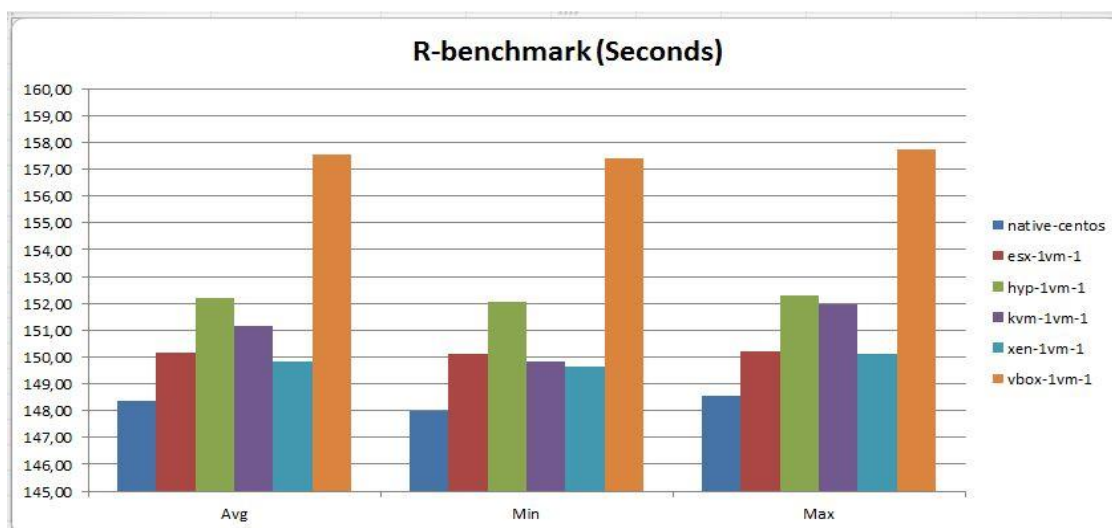
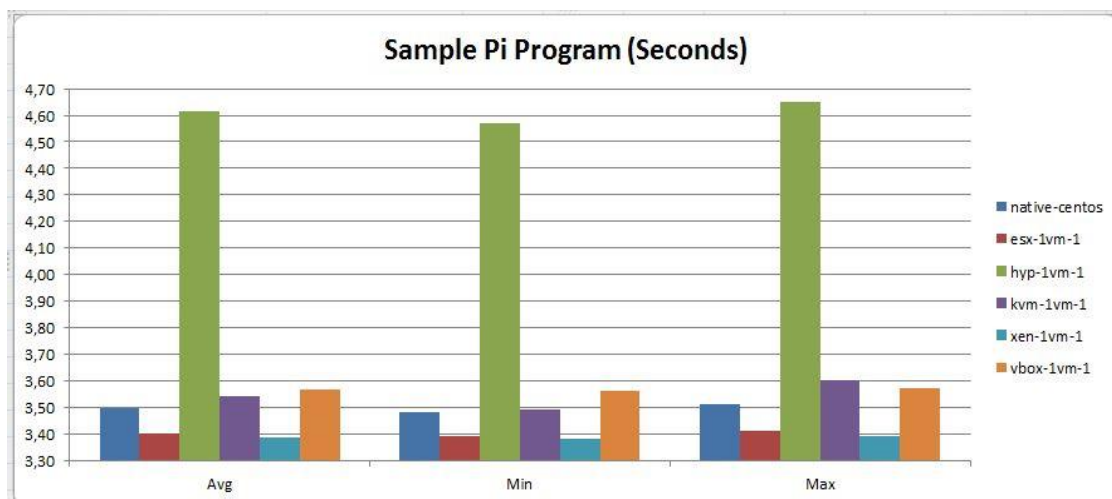




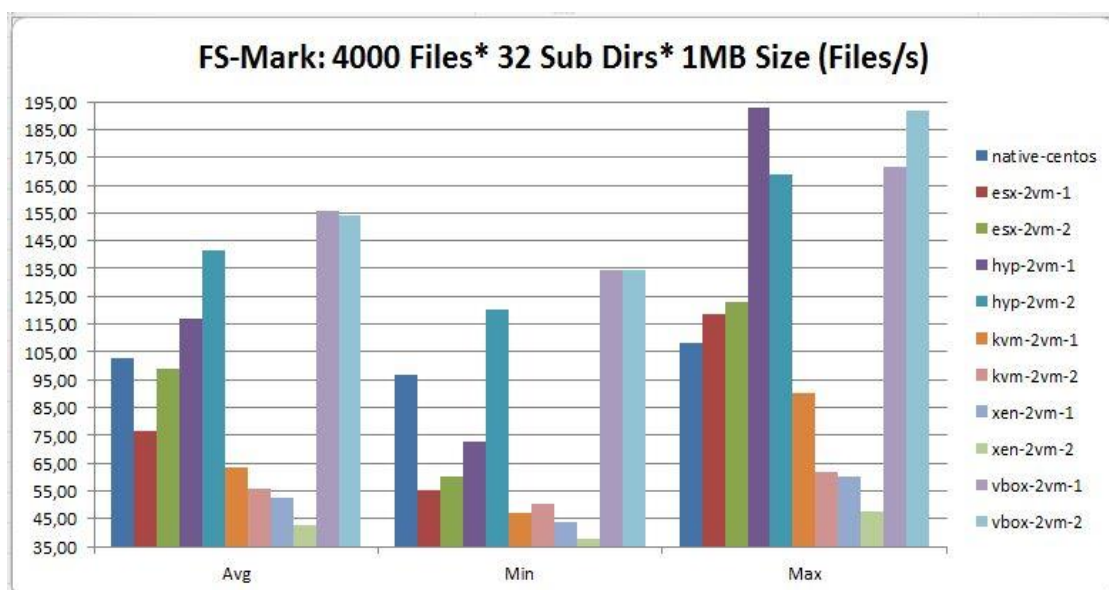
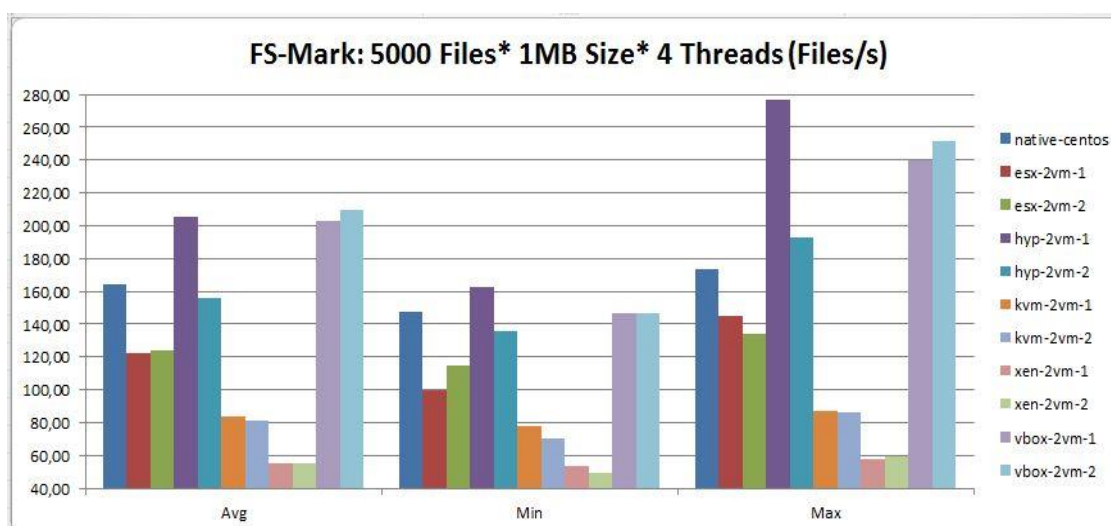
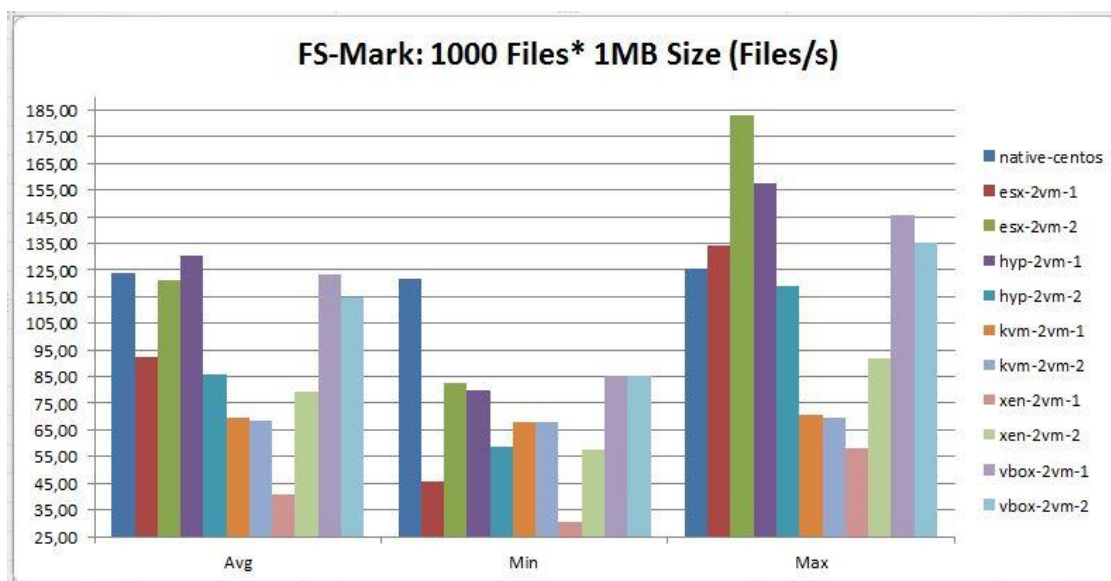


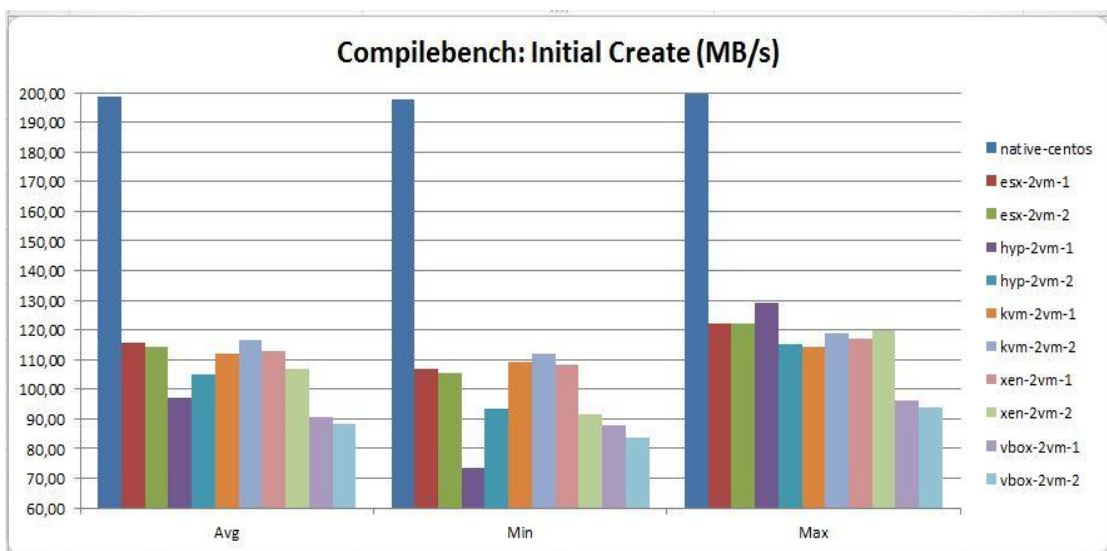
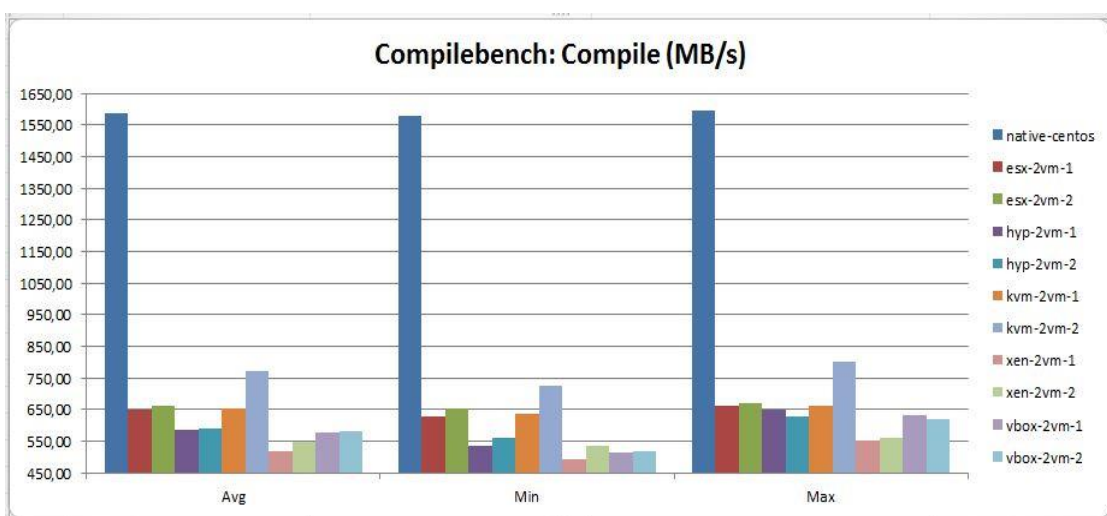
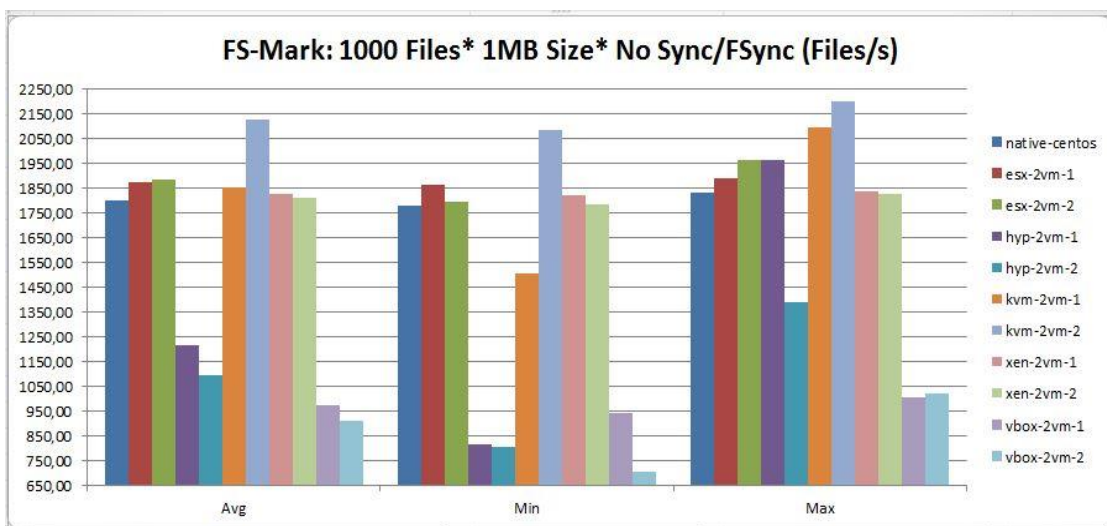


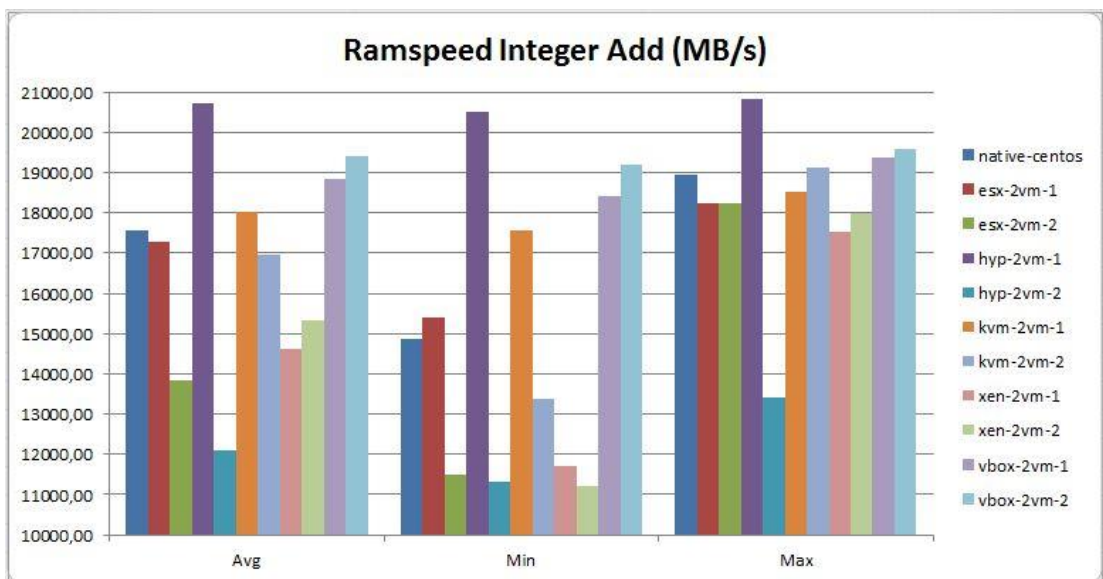
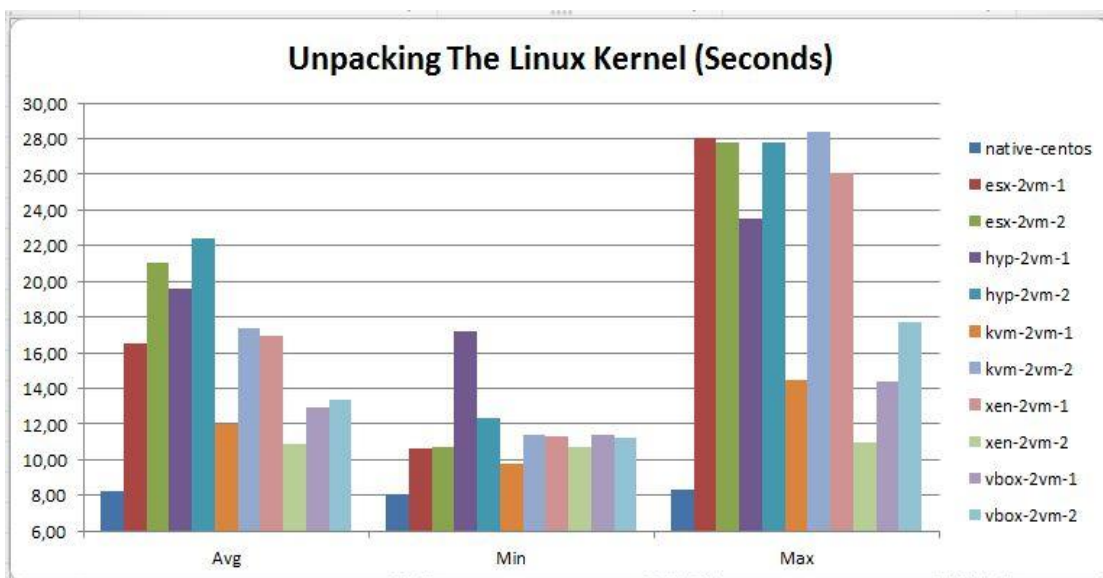
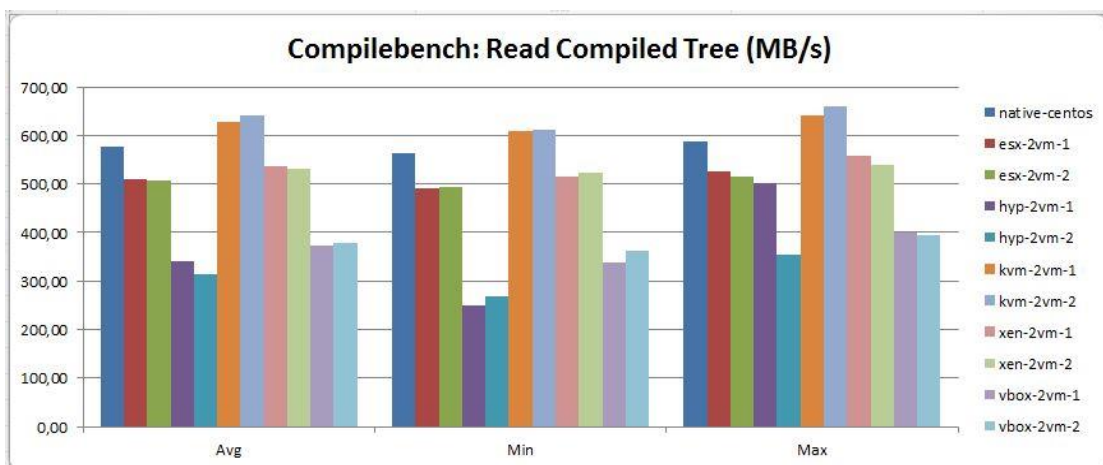


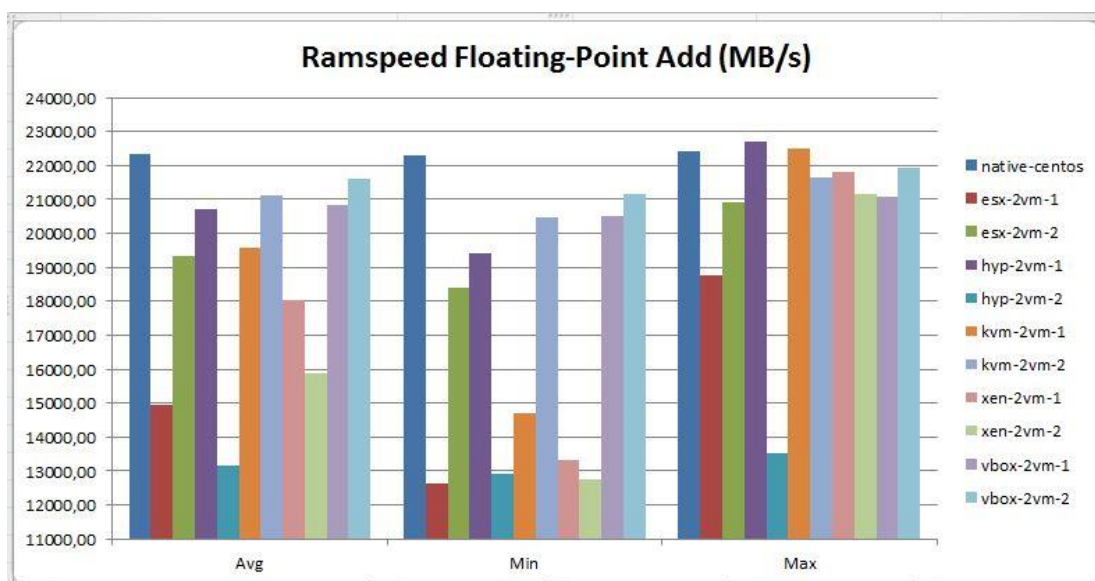
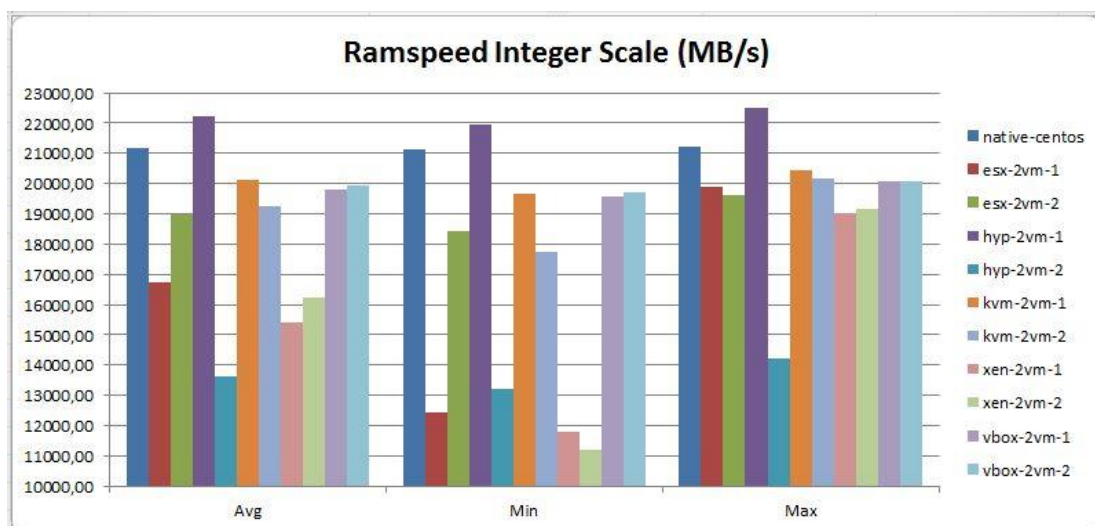
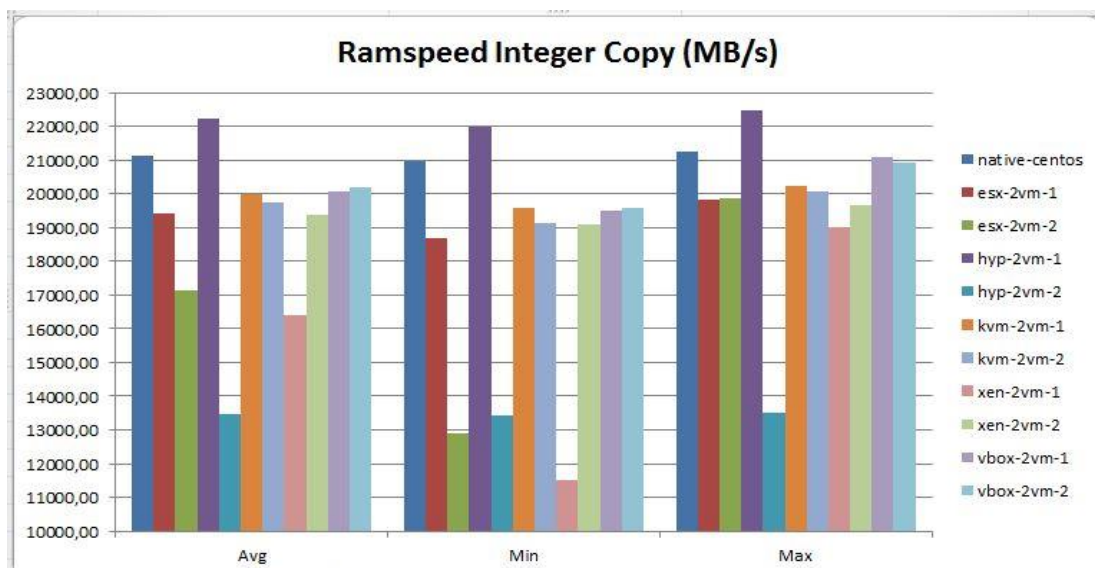


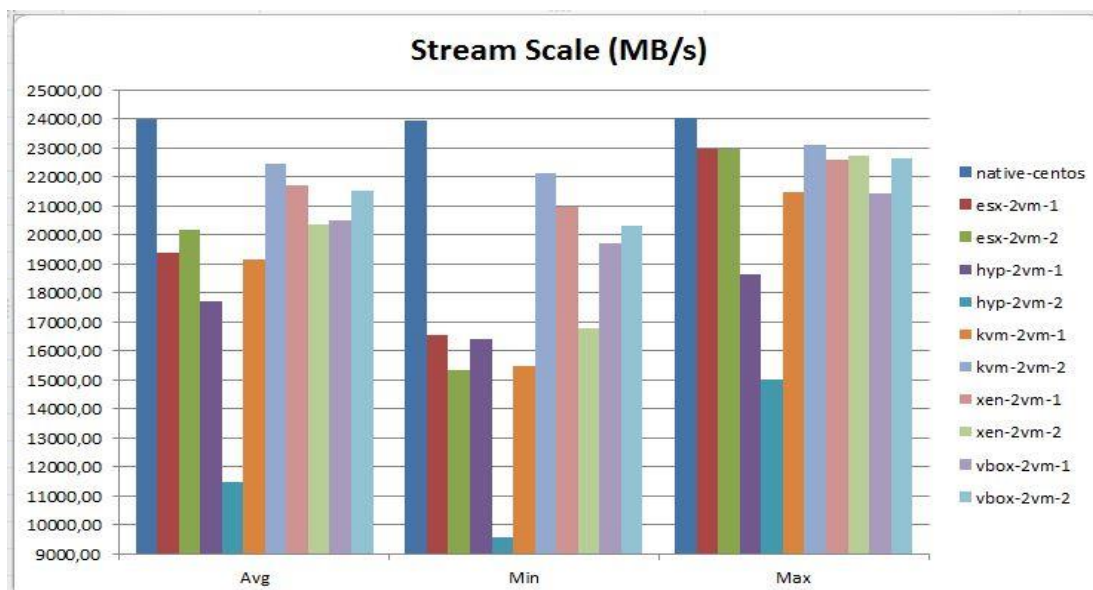
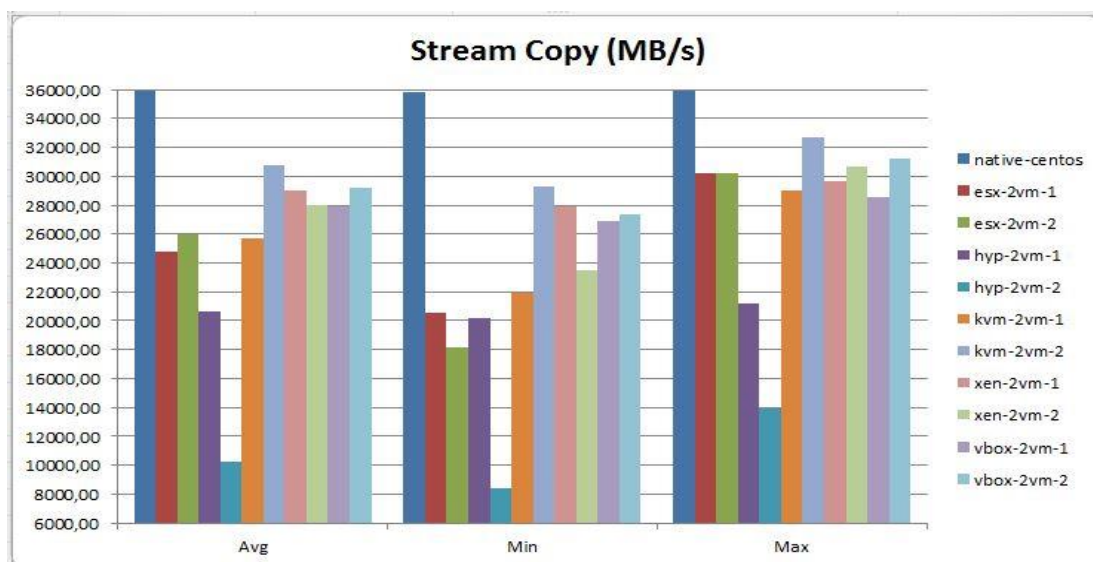
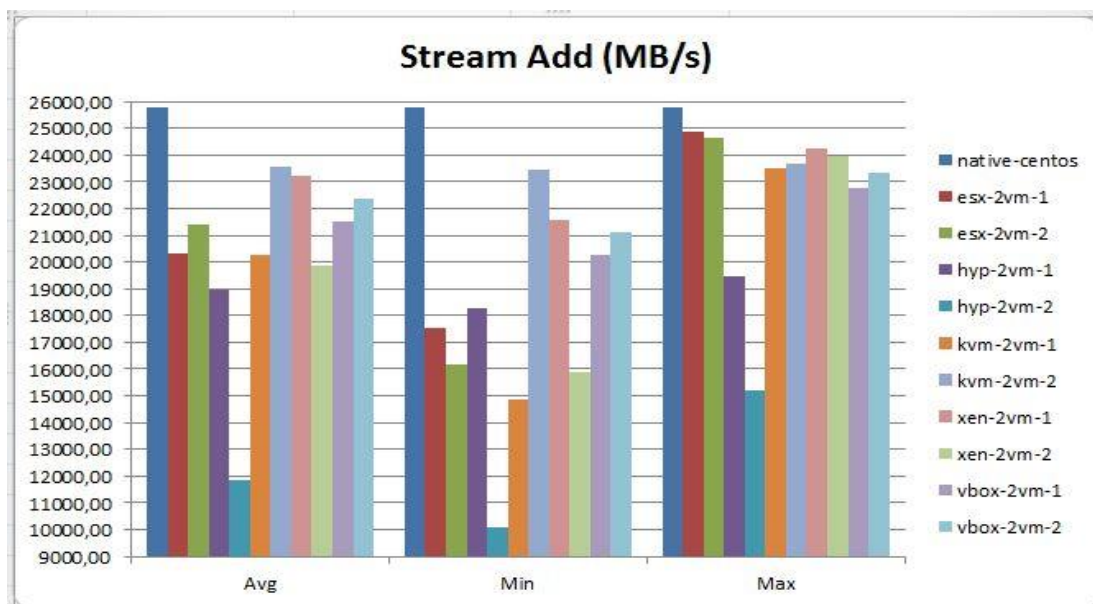
Liite 1.2. Kahden virtuaalitietokoneen tulokset kaavioina.

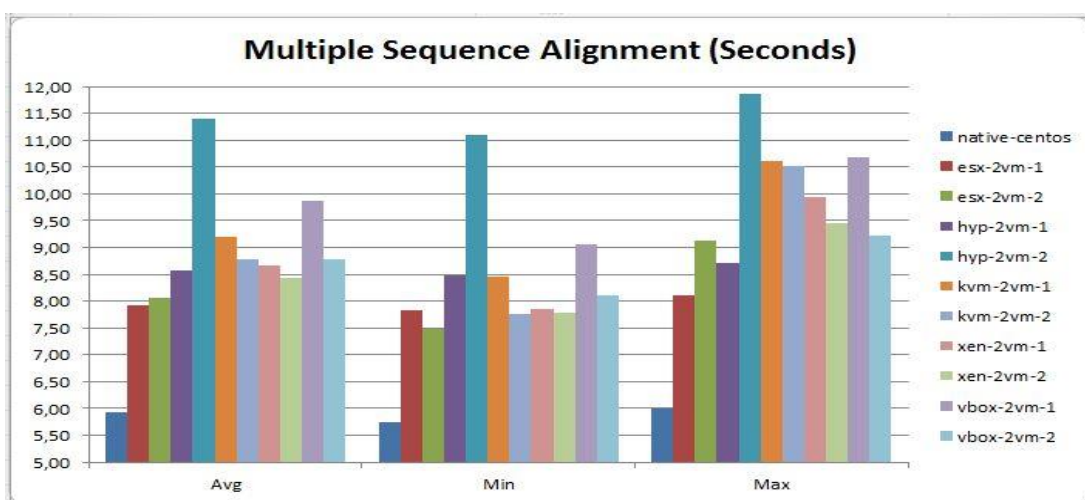
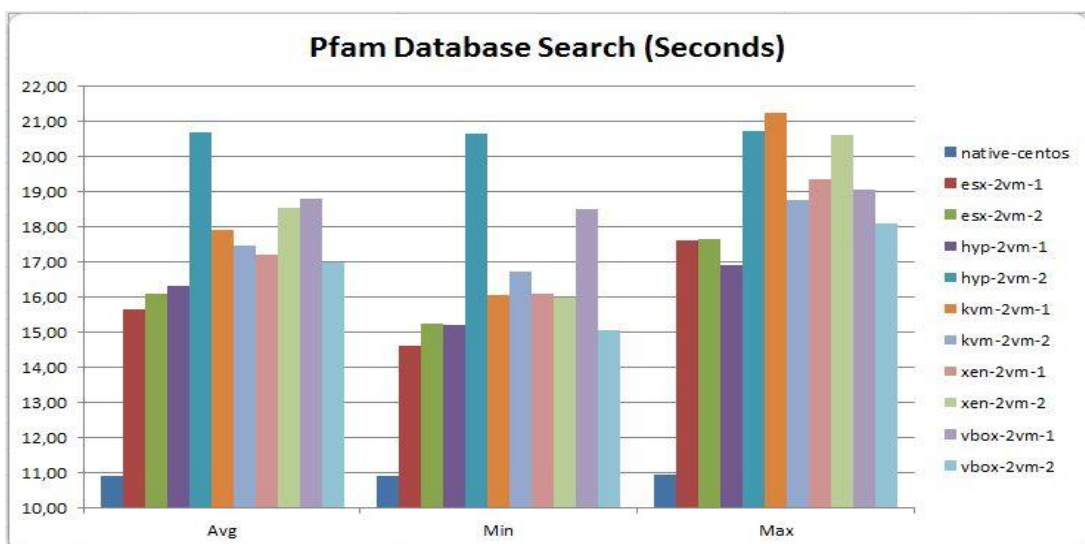
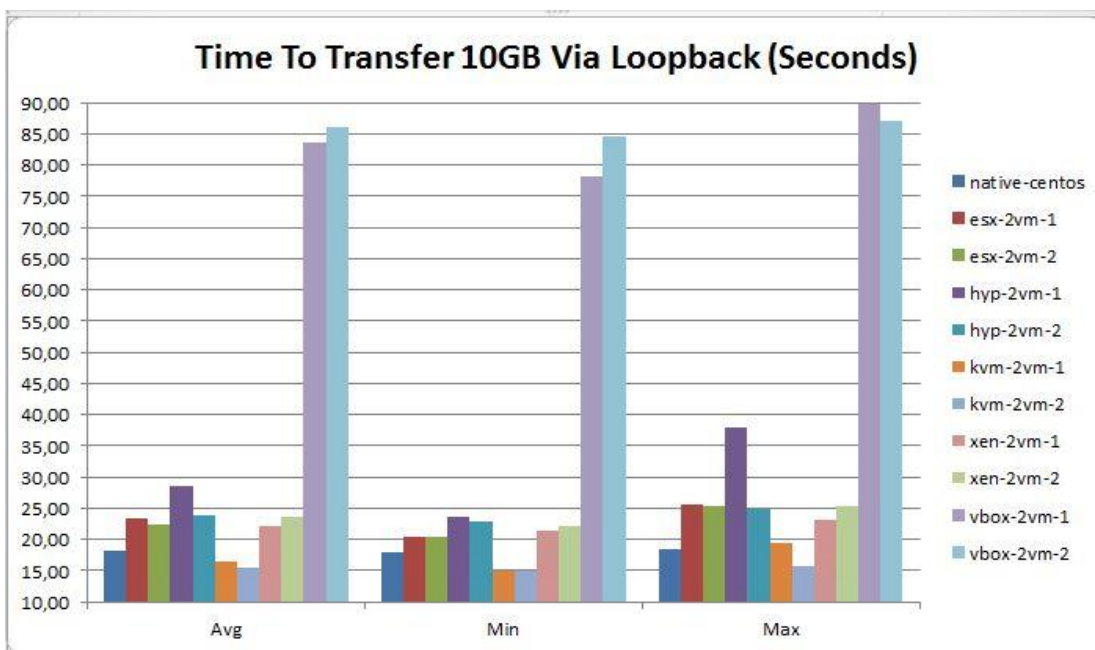


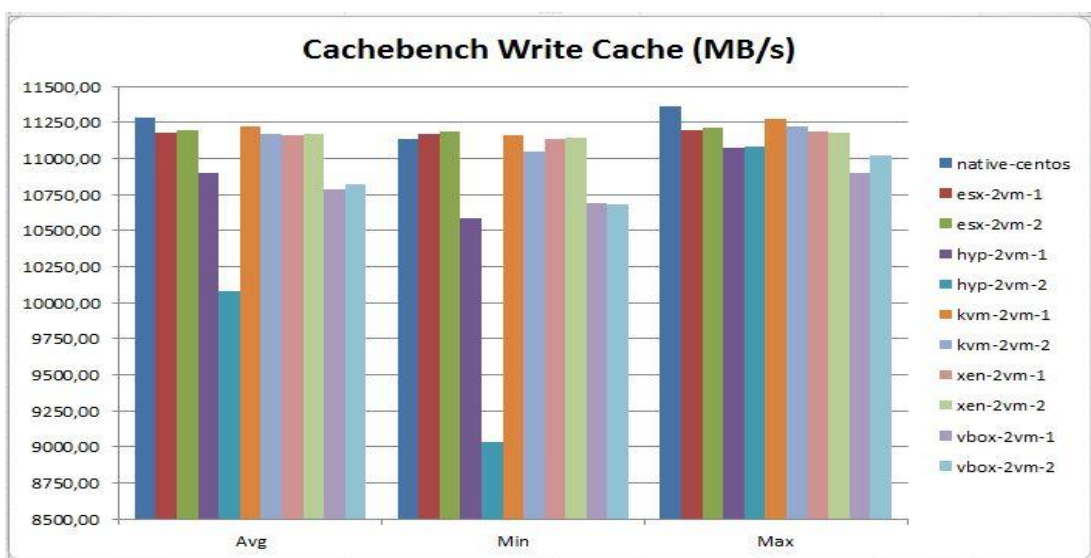
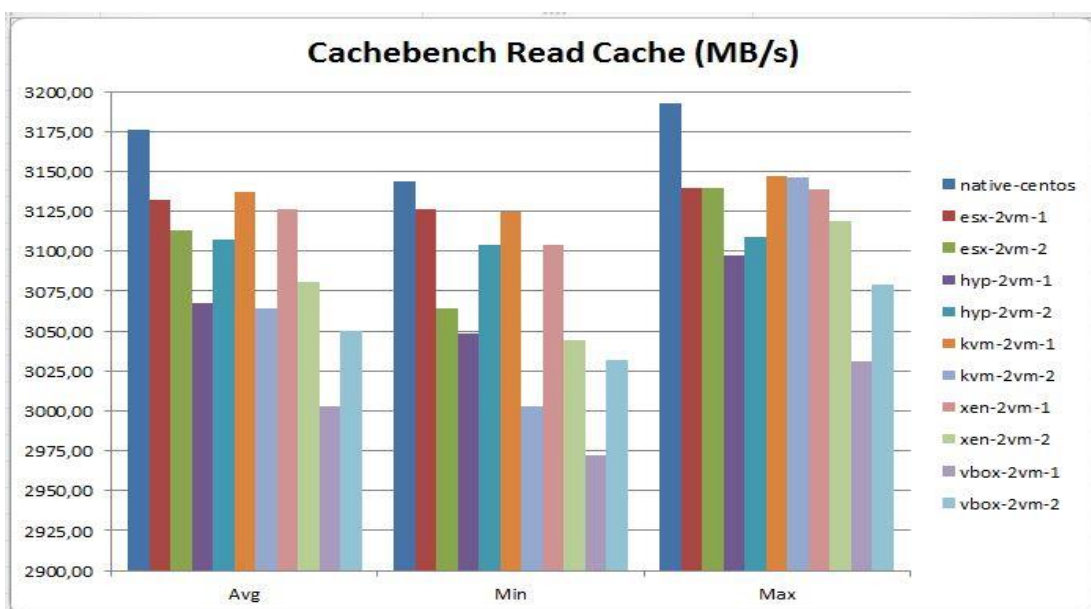
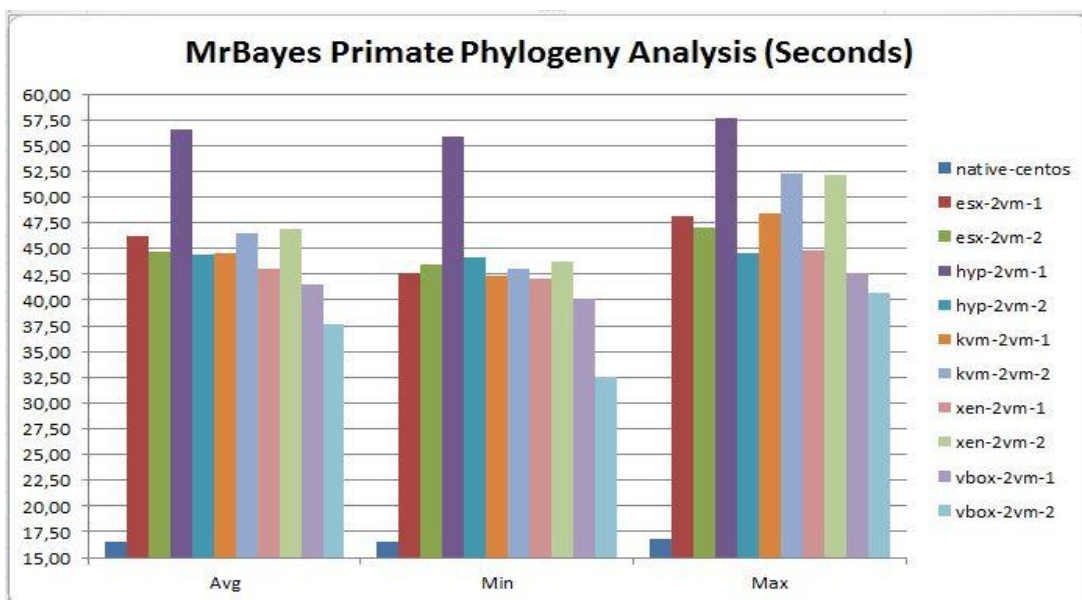


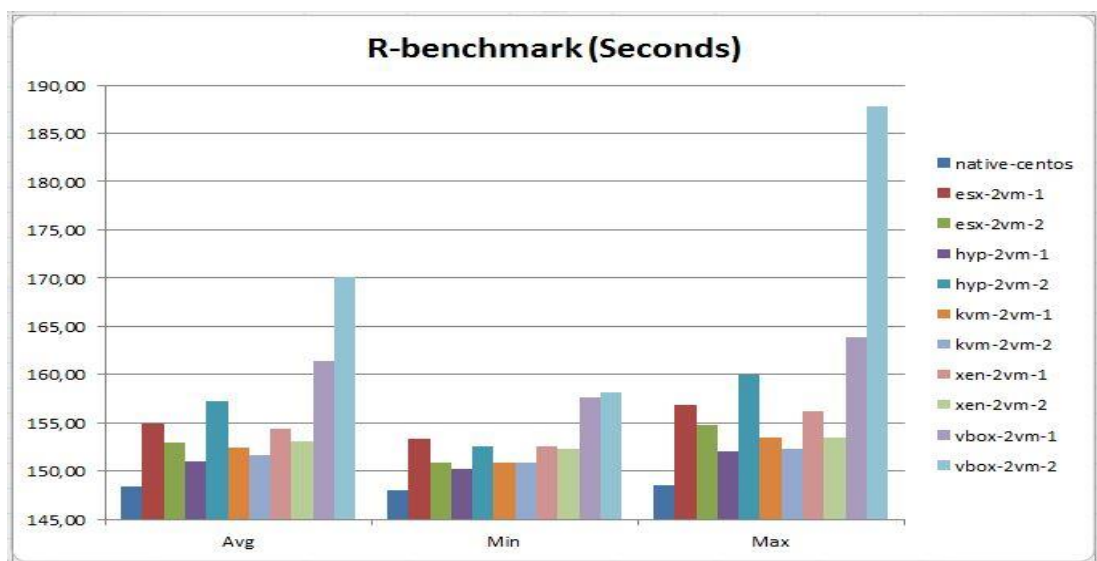
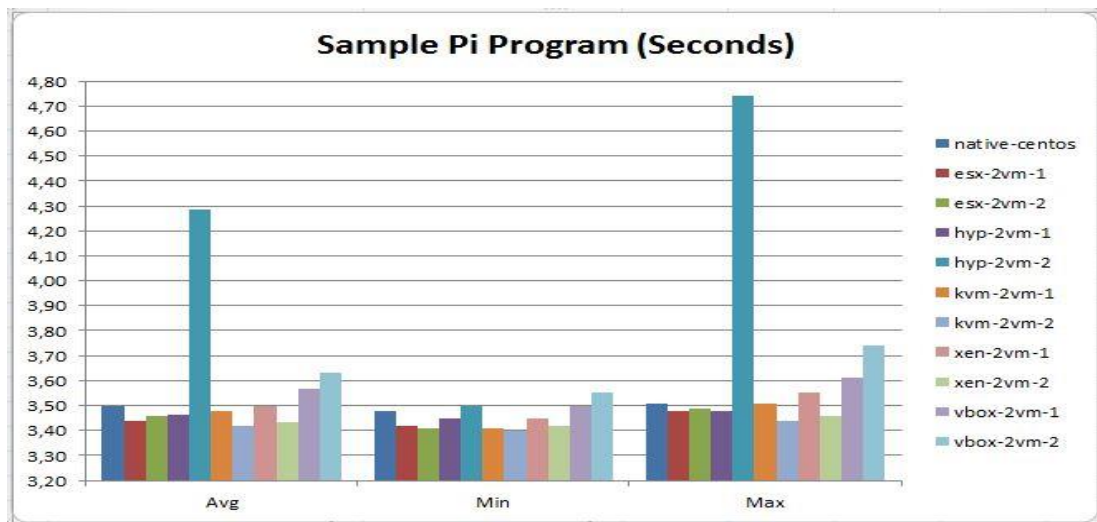












Liite 1.3. Neljän virtuaalitietokoneen tulokset kaavioina.

