

Dynamo-Sovellusprojekti

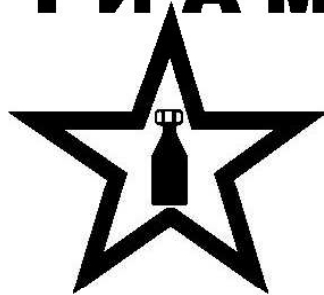
Sovellusraportti

Tero Hätinén

Joni Purojärvi

Antti Pyykkönen

D Y N A M O



Versio 1.1

Julkinen

31.1.2008

Jyväskylän yliopisto

Tietotekniikan laitos

Jyväskylä

Hyväksyjä	Päivämäärä	Allekirjoitus	Nimenselvennys
Projektipäällikkö	__.__.2008		
Tilaaja	__.__.2008		
Ohjaaja	__.__.2008		

Tietoa dokumentista

Tekijät:

- | | | |
|------------------------|--------------------|-------------|
| • Tero Häätinen (TH) | tejuhati@cc.jyu.fi | 050-3528817 |
| • Joni Purojärvi (JP) | jopuroja@cc.jyu.fi | 040-5271885 |
| • Antti Pyykkönen (AP) | anpyykko@cc.jyu.fi | 050-5376727 |

Dokumentin nimi: Dynamo-projekti, Sovellusraportti

Sivumäärä: 32

Tiedosto: sovellusraporttill.tex

Tiivistelmä: Projekti kehitti Dynamics-järjestelmää vastaamaan paremmin tietotekniikan laitoksen ja LaiLa-projektin tarpeita. Dynamo-projekti toteutti järjestelmään DHCP-tuen ja IPv4-verkon kuormituksen siedon testaustyökalun prototyypin. Dokumentissa kuvataan projektissa jatkokehitettävien ominaisuuksien toteutusratkaisuja, toteutusta, testausta ja jatkokehitysideoita.

Avainsanat: DHCP, Dynamics, jatkokehitysideat, kotiagentti, mobiililaite, Mobile IP, testaus, toteutusratkaisut, vierasagentti, virtuaalinen vierasagentti.

Muutoshistoria

Versio	Päivämäärä	Muutokset	Tekijät
0.1	27.11.2007	Ensimmäinen luonnos. Laadittu luvut "Johdanto", "Termit", "Toteutustekniikat ja työkalut", "Dynamicsin alkuperäinen toiminta ja rakenne", "Projektissa saavutetut tavoitteet ja jatkokehitys", "Kotiagentin DHCP-tuki", "Virtuaalinen vierasagentti", "Yhteenveto" ja "Lähteet".	TH, AP, JP
0.2	27.11.2007	Lisätty lukuun "Virtuaalinen vierasagentti" sisältö. Lisätty virtuaalisen vierasagentin osalta saavutetut tavoitteet ja jatkokehitysideat. Korjattu kirjoitus- ja muotoiluvirheitä.	JP
0.3	11.12.2007	Korjattu kirjoitusvirheitä kaikista luvuista ja muokattu lukujen järjestystä. Poistettu vaatimusmäärittelyssä käsiteltyjä asioita.	TH
0.4	12.12.2007	Korjattu kirjoitusvirheitä kaikista luvuista ja lisätty luku "Virtuaalisen vierasagentin toteutusratkaisut".	TH
0.5	19.12.2007	Korjattu kirjoitusvirheitä kaikista luvuista ja muokattu lukujen järjestystä edelleen. Lisätty jatkokehitysideoita ja kuvattu testauksen suoritus. Johdantoa ja yhteenvetoa on muokattu. Toteutusratkaisujen kehityksen kuvaus on lisätty lukuihin 5 ja 6.	TH
0.6	15.1.2008	Korjattu kirjoitusvirheitä kaikista luvuista. Lisätty luku 5.3 sekä kuvat Dynamicsin rakenteesta ja tilakaavioista. Sisällysluettelo on muokattu aakkosjärjestykseen.	TH
0.7	20.1.2008	Korjattu kirjoitusvirheitä kaikista luvuista. Muokattu kuvia luvusta 5.1.	TH
0.8	27.1.2008	Korjattu kirjoitusvirheitä kaikista luvuista. Muokattu kuvia luvusta 5.1 sekä lisätty lauseita lukuihin 3.1, 4.2, 8, 8.4. Muokattu lukua 6.3. Tavutettu sanoja.	TH
1.0	29.1.2008	Korjattu kirjoitusvirheitä ja tavutusta.	JP
1.1	31.1.2008	Korjattu kirjoitusvirheitä ja lähteitä.	JP

Tietoa projektista

Dynamics on Teknillisen korkeakoulun kehittämä järjestelmä Mobile IPv4-verkkoon. Se mahdollistaa liikkuvien päätelaitteiden liikkumisen verkosta toiseen katkaistamatta yhteyttä, sekä huolehtii pakettien välittämisestä verkosta toiseen. Dynamo-projekti jatkokehitti Dynamics-sovellusta vastaamaan paremmin tietotekniikan laitoksen ja LaiLa-projektin tarpeita.

Tekijät:

- Tero Hätinä (TH) `tejuhati@cc.jyu.fi` 050-3528817
- Joni Purojärvi (JP) `jopuroja@cc.jyu.fi` 040-5271885
- Antti Pyykkönen (AP) `anpyykko@cc.jyu.fi` 050-5376727

Tilaaja:

- Riku Ahonen `riahonen@jyu.fi` 040-5174014
- Olli Alanen `opalanen@jyu.fi` 014-2604974

Ohjaajat:

- Juha Huikari `juha.huikari@jyu.fi` 044-5329883
- Jukka-Pekka Santanen `santanen@mit.jyu.fi` 014-2602756

Yhteystiedot:

- Sähköpostilistat `dynamo@korppi.jyu.fi`,
 `dynamo_opetus@korppi.jyu.fi`,
 `dynamo-oma.group@korppi.jyu.fi`
- Sähköpostiarkistot <https://korppi.jyu.fi/list-archive/dynamo/ind.html>,
 https://korppi.jyu.fi/list-archive/dynamo_opetus/ind.html
- WWW-sivut <http://sovellusprojektit.it.jyu.fi/dynamo>

Sisältö

1	Johdanto	1
2	Termit	2
3	Toteutustekniikat, kehitystyökalut ja -ympäristö.	4
3.1	Toteutustekniikat ja kehitystyökalut	4
3.2	Kehitys- ja testausympäristö	4
4	Järjestelmän jatkokehityksen toteutuminen	6
4.1	Dynamicsin alkuperäinen toiminta ja rakenne	6
4.2	DHCP-tuen toteutuminen	7
4.3	Virtuaalisen vierasagentin toteutuminen	8
4.4	Mobiililaitteen toteutuminen	8
5	Kotiagentin DHCP-tuen toteutusratkaisut	9
5.1	Tekninen toteutus	9
5.2	Kehitysvaiheet	13
5.3	Säikeistetyn ja säikeistämättömän Dynamics-version ero	14
5.4	Uusi tietorakenne	14
5.5	Tietorakennetta käsittelevät aliohjelmat	15
5.6	Kotiagentin muutetut aliohjelmat ja tiedostot	16
5.7	Kotiagenttiin lisätyt tiedostot	18
6	Virtuaalisen vierasagentin toteutusratkaisut	19
6.1	Tekninen toteutus	19
6.2	Kehitysvaiheet	19
6.3	Kääntäminen	20
6.4	Uudet tietorakenteet	20
6.5	Asetustiedosto	21
6.6	Lisätyt aliohjelmat	22
6.7	Muutetut aliohjelmat	23
7	Suoritettut testaukset ja tulokset	24
7.1	Testaus toteutusvaiheessa	24
7.2	Testaus projektin testausympäristössä	24
7.3	DHCP-tuen testaus tilaajan verkossa	25

8	Jatkokehitysideoita	27
8.1	DHCP-tuen jatkokehitys	27
8.2	Muut kotiagentin jatkokehitysideat	28
8.3	Virtuaalisen vierasagentin jatkokehitys	28
8.4	Dynamicsin rakenne ja kommentointi	28
9	Yhteenveto	30
	Lähteet	31

1 Johdanto

Dynamics on Teknillisen korkeakoulun kehittämä järjestelmä Mobile IPv4 -verkkoon. Se mahdollistaa liikkuvien päätelaitteiden liikkumisen verkosta toiseen katkaisematta yhteyttä, sekä huolehtii pakettien välittämisestä verkosta toiseen. Dynamo-projekti oli Jyväskylän yliopiston tietotekniikan laitoksen syksyn 2007 sovellusprojekti. Se määritteli ja suunnitteli sekä osin toteutti ja testasi Dynamics-järjestelmään tarvittavat muutokset, joilla Dynamics saatiin paremmin vastaamaan LaiLa-projektin tarpeita.

Dokumentissa esitetään järjestelmän jatkokehityksen tavoitteiden toteutuminen, toteutusratkaisut ja projektin jälkeiset jatkokehitysideat. Jatkokehitysvaatimukset ja niiden toteutumiset esitellään vaatimusmäärittelyssä [13]. Projektin käytänteitä, aikataulua, riskejä ja muita projektin läpivientiin liittyviä asioita käsitellään projektiraportissa [4]. Testaussuunnitelmassa [12] sekä testausraporteissa [9], [10], ja [11] esitellään testitapaukset ja testaustulokset. DHCP-tuen asennus esitellään asennusohjeessa [14]. Muutosdokumentissa [15] on esitetty lyhyesti järjestelmään toteutetut muutokset. Kotiagentin ja virtuaalisen vierasagentin Doxygen-dokumenteissa [7] ja [8] on esitetty aliohjelmat ja funktiot.

Luvussa 2 on kuvattu tärkeimmät projektiin liittyvät termit. Sovelluskehityksessä käytettyjä ohjelmistoja ja tekniikoita esitellään luvussa 3. Järjestelmän jatkokehitystavoitteiden toteutuminen esitellään luvussa 4. Toteutusratkaisuja kuvataan tarkemmin DHCP-tuen osalta luvussa 5 ja virtuaalisen vierasagentin osalta luvussa 6. Luvussa 7 käsitellään tarkemmin DHCP-tuen ja virtuaalisen vierasagentin testausta ja sen tuloksia. Luvussa 8 esitellään jatkokehitysideoita.

2 Termit

Luvussa kuvataan dokumentin ja projektin olennainen termistö. Aihealueen termit ovat seuraavat:

Dynamics	on Teknillisen korkeakoulun kehittämä järjestelmä Mobile IPv4 -verkkoon.
Dynamo-verkko	on Dynamo-projektissa käytettävä verkko, jonka vaatimuksia vastaavaksi Dynamics-järjestelmää muokataan.
GPL	on käytetyin vapaan lähdekoodin ohjelmien lisenssi.
LaiLa	on tietotekniikan laitoksen tutkimusprojekti, jossa tarkastellaan langattomien laajakaistapalveluiden hallintaa multi-access -verkossa.

Teknisiä termejä ovat seuraavat:

Agenttipyyntöviesti	(engl. <i>Agent Solicitation Message</i>) on mobiililaitteen generoima viesti, kun laite tulee verkkoon ja haluaa rekisteröityä agentille.
Agentti	on joko koti- tai vierasagentti.
Autentikointi	on tapahtuma, jolla kaksi osapuolta varmistavat toistensa identiteetin.
DHCP	(engl. <i>Dynamic Host Configuration Protocol</i>) on verkko-protokolla, joka jakaa dynaamisesti IP-osoitteita verkkoon liittyville päätelaitteille.
Elinaika	on tunnelin elinaika agenttien välillä.
Lease-aika	on aika sekunteina, jonka ajan DHCP:n antama IP on käytettävissä.
Kotiagentti	on kotiverkossa oleva ohjelmisto, joka tunneloi kaikki mobiililaitteelle menevät paketit ja ylläpitää tietoa mobiililaitteen sijainnista.

Kotiosoite	on mobiililaitteen kotiverkon osoite, joka pysyy aina samana, vaikka laite siirtyisi kotiverkosta vierasverkkoon.
Kotiverkko	on verkko, johon mobiilin päätelaitteen kotiosoite kuuluu.
Jaettu salaisuus	on symmetrisessä salauksessa käytetty avain, jonka avulla selväkielinen teksti salataan.
MAC-osoite	on laitteen fyysinen osoite.
Mobiililaite	on laite, joka voi liikkua verkosta toiseen.
Mobile IPv4	on IPv4-protokollan laajennus, joka mahdollistaa liikkuvien päätelaitteiden toiminnan IPv4-verkossa.
NAI	(engl. <i>Network Address Identifier</i>) on Mobile IPv4 -verkossa käytettävä mobiililaitteen tunniste.
Rekisteröintiviesti	on mobiililaitteen kotiagentille generoitu rekisteröintipyyntö.
SPI	on mobiililaitteen salausavain.
Tiedostokuvaaja	(engl. <i>File Descriptor</i>) on viite muistialueeseen, jota voi käyttää kuten tiedostoa.
Tilapäisosoite	(engl. <i>Care-off -Address</i>) on mobiililaitteen osoite vierasverkossa (vierasagentin ja mobiililaitteen välillä).
Vastaanottava solmu	(engl. <i>Corresponded Node</i>) on mobiililaitteen kanssa keskusteleva palvelu tai kone.
Vierasagentti	ohjaa liikennettä kotiagentin ja mobiilin päätelaitteen välillä, sekä tarjoaa verkon palvelut päätelaitteelle.
Vierasverkko	on verkko, jossa mobiili päätelaite vierailee.
Viestiputki	(engl. <i>Pipe</i>) on erikoistiedosto, jolla on kaksi tiedostokuvaajaa.
Virtuaalinen vierasagentti	on kotiagentin kuormituksen testaukseen tarkoitettu työkalu.

3 Toteutustekniikat, kehitystyökalut ja -ympäristö.

Luvussa esitellään Dynamics-järjestelmän jatkokehityksessä käytetyt toteutustekniikat ja kehitystyökalut.

3.1 Toteutustekniikat ja kehitystyökalut

Järjestelmän jatkokehitys tapahtui C-kielellä testausympäristössä KDevelop 3.4.1 -sovelluskehitysympäristöllä. Järjestelmän kehitys DHCP-tuen osalta oli inkrementaalista. Virtuaalisesta vierasagentista kehitettiin prototyyppi.

Jatkokehitettävään ohjelmakoodiin tehtiin pieniä muutoksia, joiden toimivuus varmennettiin heti yksikkötestauksella. Wireshark-ohjelman versiolla 0.99.4 seurattiin verkon liikennettä ja GDP-debuggerilla tarkailtiin muuttujien tiloja. Lokitulostuksilla selvitettiin ohjelman toimintaa.

Lisätyt ja muutetut ohjelmakoodit kommentoitiin Doxygenin vaatimalla tavalla. Muokatuista ja uusista ohjelmakoodeista laadittiin englanninkielinen muutosdokumentti [15].

UDHCP-asiakasohjelmaa käytettiin Dynamicsin kotiagentin DHCP-tuen toteutuksessa. DHCP-palvelimena käytettiin Linuxin DHCPD-palvelinta.

DHCP-käsittelyä yritettiin eriyttää säikeistyksen avulla muusta Dynamicsin kotiagentin toiminnasta. Rajapinta soketteihin ja pakettien välitykseen oli Dynamics-järjestelmään jo toteutettu Dynamics-projektissa.

Soketit ovat yksilöllisiä verkkokommunikaation alku- ja loppupisteitä. Sokettien kautta kirjoitetaan viestejä verkkoon ja luetaan verkosta tulevia viestejä. Dynamicsissä käytetään soketteja UDP-protokollan mukaisten viestien luomiseen.

3.2 Kehitys- ja testausympäristö

Järjestelmän ohjelmointi tapahtui projektin testausympäristössä, koska Dynamics-järjestelmästä puuttuivat projektitiedostot, joita tarvitaan kehittämiseen kehitystyökalulla (IDE). Testausympäristössä voitiin varmistaa myös paketin lähteminen verkkoon oikeilla arvoilla ja saapuminen oikeaan osoitteeseen.

Testausympäristö koostui neljästä koneesta. Kaikkiin neljään koneeseen asennettiin Linux-käyttöjärjestelmä. Lisäksi yhteen koneeseen (mobiili päätelaite) asennettiin Windows XP sekä kaupallinen Cisco-mobiililaiteohjelmisto. Testausympäristön koneet muodostivat itsenäisen verkon erilleen yliopiston verkosta. Testahuoneeseen asennettiin lisäksi yksi kone, joka oli yhteydessä yliopiston verkkoon, mutta erillään testiverkosta. Testausympäristö on esitelty tarkemmin testaussuunnitelmassa [12].

4 Järjestelmän jatkokehityksen toteutuminen

Projekti jatkokehitti Dynamics Mobile IPv4 -ohjelmistoa, joka on kirjoitettu C-kielellä. Luvussa esitellään sovellukseen kehitetyt toiminnot pääpiirteittäin.

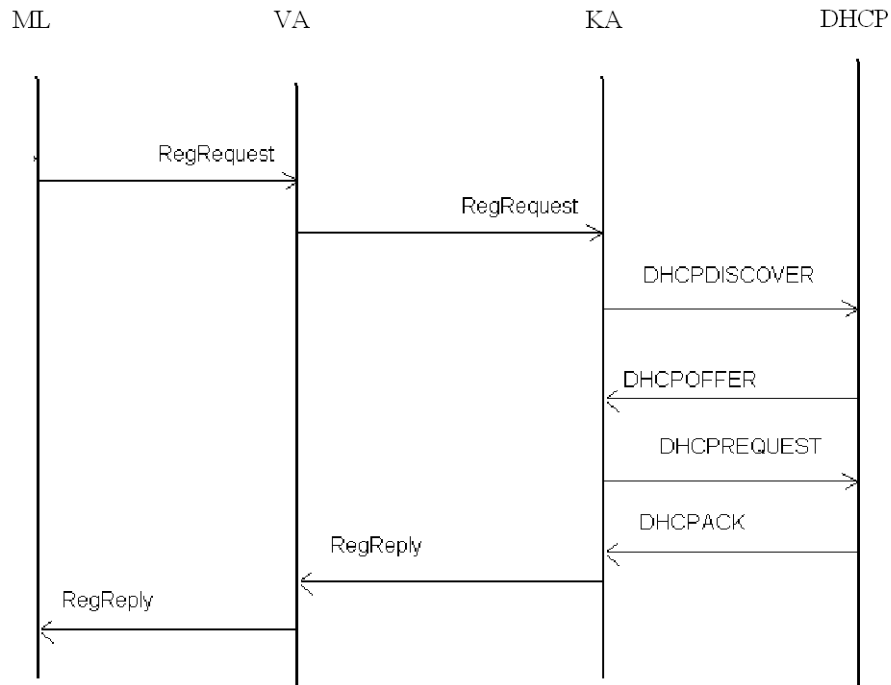
Projektin tärkeimpänä tavoitteena oli lisätä DHCP-tuki Dynamicsin kotiagenttiin. DHCP-tuen toteutuksessa hyödynnettiin C-kielellä kirjoitettua UDCHP-asiakasohjelmaa. DHCP-tuki toteutettiin säikeistystä lukuunottamatta. Virtuaalisen vierasagentin tavoitteet toteutettiin osittain. Se lähettää useita liittymis- ja poistumisviestejä verkkoon asetustiedostossa määritettyjen parametrien mukaan.

4.1 Dynamicsin alkuperäinen toiminta ja rakenne

Dynamics (versio 0.8.1) on Teknillisen korkeakoulun kehittämä järjestelmä Mobile IPv4 -verkkoon. Se mahdollistaa liikkuvien päätelaitteiden liikkumisen verkosta toiseen katkaisematta yhteyttä, sekä huolehtii pakettien välittämisestä verkosta toiseen.

Alkuperäistä toimintaa on esitelty tarkemmin vaatimusmäärittelyssä [13]. Kuvassa 4.1 esitellään Dynamicsin rakenne DHCP-tuen kera.

Dynamics pitää sisällään kaikki tarvittavat ohjelmat Mobile IPv4 -verkon pystytykseen. Mukana tulevat sovellukset ovat kotiagentti, vierasagentti ja mobiililaite (kuvassa 4.1 KA, VA ja ML). Dynamics ei tue dynaamisia koti- tai tilapäisosoitteita, joten kaikki IP-tiedot tulee syöttää etukäteen sovellusten asetustiedostoihin.



Kuva 4.1: Dynamicsin rakenne DHCP-tuen kera.

4.2 DHCP-tuen toteutuminen

DHCP-tuen tavoitteiden toteutuminen on esitelty vaatimusten osalta vaatimusmäärittelyssä [13] ja toteutusratkaisujen osalta luvussa 5.

DHCP-tuen tavoitteet saatiin toteutettua siltä osin, että kotiagentti pyytää IP-osoitteet dynamisesti DHCP-palvelimelta, tallentaa ne tietorakenteeseen ja lähettää IP-osoitteen sitä pyytäneelle mobiililaitteelle. Säikeistykseen puutteesta johtuen DHCP-keskustelun ajan kotiagentti ei voi vastata muiden mobiililaitteiden rekisteröitymispyyntöihin. DHCPRELEASE-paketin lähettäminen ei onnistunut koodissa olleen virheen takia.

DHCP-tuen toteuttamisessa oli ongelmana DHCP-keskustelun toteuttaminen omassa säikeessä. Monimutkaisen kotiagentin rakenteesta johtuen muuttujien siirtäminen aliohjelmista säikeeseen ei onnistunut, ja säikeistys sovittiin tilaajan kanssa jatkokehitykseen.

DHCP-tuen toteutuksessa huomattiin, että Dynamicsin vierasagentti ei kykene välittämään IP-osoitteesta 0.0.0.0 tullutta pakettia kotiagentille. Mobiililaite ei siten

voi kysyä IP-osoitetta dynaamisesti, kun se on liittynyt vierasagentin kautta verkkoon. Ongelma kierrettiin käyttämällä kiinteää osoitetta Ciscon asiakasohjelmistossa, vaikka IP-osoite haetaan dynaamisesti DHCP-palvelimelta.

4.3 Virtuaalisen vierasagentin toteutuminen

Virtuaalisen vierasagentin toteutuminen on esitelty tarkemmin vaatimusten osalta vaatimusmäärittelyssä [13] ja toteutusratkaisujen osalta luvussa 6.

Tavoitteiden mukaisesti virtuaalinen vierasagentti lähettää rekisteröintipyyntöjä ja poistumispyyntöjä kotiagentille. Virtuaalisten mobiililaitteiden asetukset luetaan erillisestä tiedostosta, jonka avulla virtuaalista vierasagenttia hallitaan.

Virtuaalinen vierasagentti toteutettiin Dynamicsin mobiililaitesovelluksen ympärille, koska sovellus sisälsi jo valmiiksi lukuisia aliohjelmiä pakettien lähettämiseen ja vastaanottamiseen. Virtuaaliseen vierasagenttiin luotiin tietorakenne ja sen käsittelyyn tarvittavat aliohjelmat. Virtuaalinen vierasagentti lukee asetukset käyttäjän antamien parametrien mukaan asetustiedostosta ja luo niiden pohjalta tietorakenteen. Tietorakenteen luomista käyttäjän antamien komentoriviparametrien avulla ei ehditty projektissa toteuttaa.

4.4 Mobiililaitteen toteutuminen

Dynamics-järjestelmän mobiililaitte ei tue dynaamista kotiosoitetta. Jatkokehitettävän järjestelmän testauksessa käytettiin Ciscon Mobile IPv4 -asiakasohjelmistoa, joka tukee dynaamista kotiosoitetta.

5 Kotiagentin DHCP-tuen toteutusratkaisut

Luvussa kuvataan DHCP-tuen toteutus, toteutuksen kehitysaskeleet sekä tietorakenne ja aliohjelmat. Dynamicsiin liitettiin DHCP-tuki dynaamiselle kotiosoitteella, mutta DHCP-keskustelua ei saatu säikeistettyä.

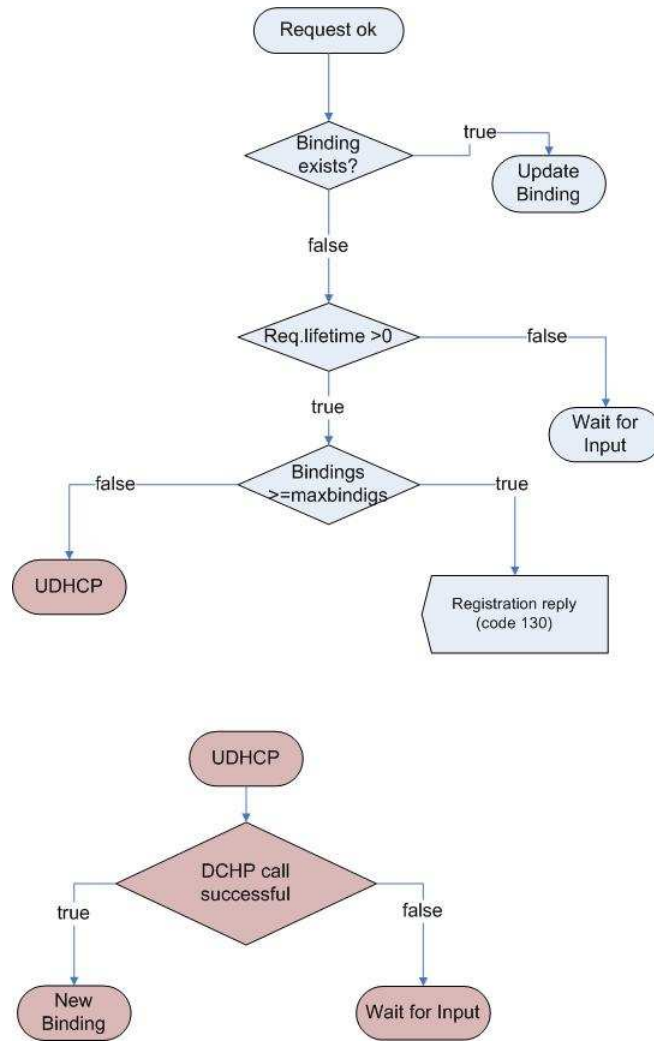
5.1 Tekninen toteutus

Dynamicsin kotiagenttiin lisättiin **tuki dynaamiselle kotiosoitteelle**. Kotiagentti hyväksyy IP-osoitteesta 0.0.0.0 tulevat rekisteröintipyynnöt, tekee tämän jälkeen DHCP-kutsun ja palauttaa rekisteröintivastauksessa DHCP:lta saadun IP-osoitteen kotiosoitteena mobiililaitteelle.

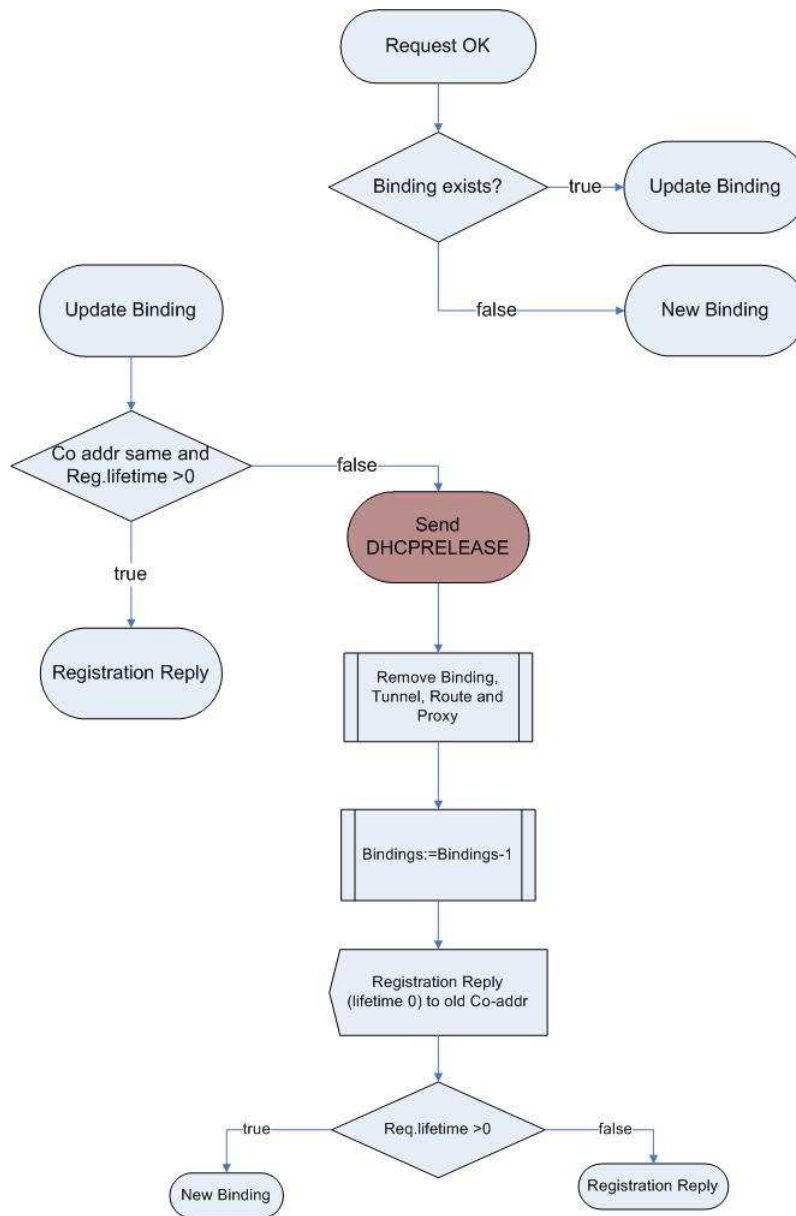
UDHCP:n `udhcp.c`-tiedoston lähdekoodi siirrettiin kokonaan kotiagentin `ha.c`-tiedostoon. Tämän jälkeen UDHCP:sta siirrettyä lähdekoodia muokattiin siten, että UDHCP käytti kotiagenttiin rakennettua tietorakennetta. UDHCP palauttaa DHCP-palvelimelta saadun IP-osoitteen sen sijaan, että se asettaisi osoitteen pyytävän verkkolaitteen osoitteeksi. Kotiagentti asetettiin lähettämään DHCP-kutsuja laitteen rekisteröityessä, poistuessa ja tunnelin elinaikaa uusittaessa.

DHCP-tuen toimintaa on esitelty lähdekoodin muutosdokumentissa [15] ja referenssimanuaalissa [7].

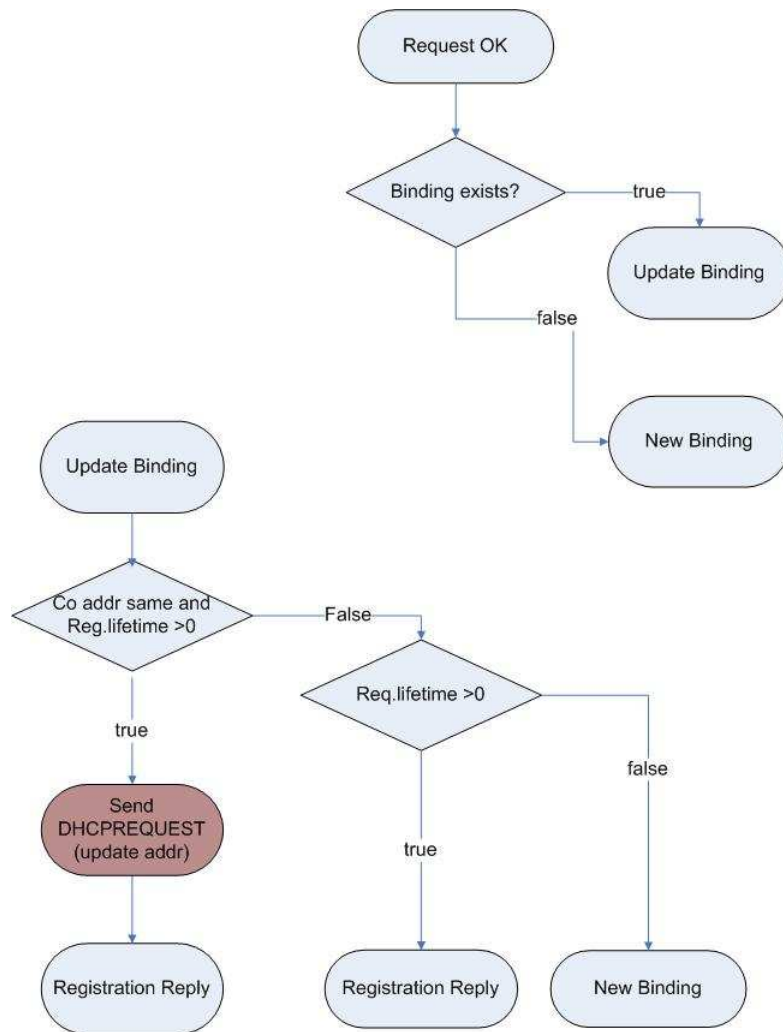
Dynamicsin kotiagentin toimintaa havainnollistetaan mobiililaitteen liittymisen osalta kuvan 5.1 tilakaavioissa, verkosta poistumisen osalta kuvassa 5.2 ja IP-osoitteen uusimisen osalta kuvassa 5.3. Kuvissa tilat `New Binding` ja `Registration Reply` ovat alkuperäisen Dynamicsin tilakaavion tiloja [1].



Kuva 5.1: Mobiililaite liittyy verkkoon.



Kuva 5.2: Mobiililaite poistuu verkosta.



Kuva 5.3: Mobiililaite uusii IP-osoitteensa.

5.2 Kehitysvaiheet

DHCP-tuen toteutus aloitettiin **tutustumalla** Dynamicsin kotiagentin lähdekoodiin ja dokumentointiin sekä selvittämällä kotiagentin aliohjelmien toimintaa ja viestinvälitysketjuja. Samalla tuli tutuiksi C-kieli ja IPv4-protokolla.

Tutustumisen jälkeen **mietittiin DHCP-tuen toteutustapoja**. Vaihtoehtoina olivat erillinen IP-protokollan avulla Dynamicsin kanssa keskusteleva DHCP-asiakas tai DHCP-asiakasohjelman sisällyttäminen osaksi Dynamicsin kotiagenttia. Tekninen ohjaaja löysi GPL-lisenssin alaisen UDHCPC-asiakasohjelman, joka soveltui parhaiten projektin tarpeisiin. Vaihtoehtona oli myös IPv6-tuella varustettu ISC-asiakasohjelma. Asiakasohjelmaksi valittiin UDHCPC ja toteutustavaksi DHCP-asiakasohjelman liittäminen osaksi Dynamicsia, koska UDHCPC oli toteutettu selkeästi ja ilman IPv6-tukea. DHCP-keskustelu ja -viestit tulivat tutuiksi jäsenille tässä vaiheessa.

Dynamicsin ja UDHCPC:n yhdistäminen toteutettiin viikossa. Ensimmäisen DHCP-viestin lähettämisen jälkeen toteutettiin tietorakenne, joka pitää yllä tietoa rekisteröityneistä mobiililaitteista ja niiden tiedoista. Tässä vaiheessa huomattiin, että DHCP-keskustelu DHCP-palvelimen kanssa on toteutettava erillisessä säikeessä, jotta Dynamicsin pääohjelma ei jäisi odottamaan DHCP-keskustelun loppumista.

Säikeistykseen toteuttaminen aiheutti ongelmia. Aluksi säikeistettiin koko Dynamicsin viestinkäsittely ja UDHCPC-keskustelu, jolloin säikeitä tuli sadoittain liikaa. Toteutusajatus ei siten ollut oikea, joten kokeiltiin jakaa viestinkäsittely kahtia. IP:n hankkiminen ja vastausviestin muodostus päätettiin siten toteuttaa omassa säikeessä. Tämä toinen toteutusratkaisu ei toiminut paremmin kuin ensimmäinen. Turhia säikeitä muodostui edelleen sadoittain. Kolmas säikeistysidea oli säikeistämisen siirtäminen koodin aikaisempaan kohtaan. Tämä idea osoittautui myös toimimattomaksi, sillä ylimääräisiä säikeitä syntyi edelleen. Todettiin säikeen sisällä olevan edelleen liikaa toiminnallisuutta.

Säikeistykseen toteuttamisen oikeaksi kohdaksi todettiin em. kokeilujen pohjalta DHCP-kutsu, jolloin vain UDHCPC:n käsittelemät viestit säikeistetään. Em. ajatuksen ongelmaksi tuli Dynamicsin parametrien välitys säikeelle ja säikeeltä takaisin Dynamicsin koodiin. Ratkaisu osoittautui liian työlääksi, joten projektin osalta DHCP-tuen säikeistys sovittiin tilaajan kanssa jatkokehitykseen.

DHCP-tuen viimeisenä vaiheessa **testattiin** säikeistämätöntä DHCP-keskustelua, etsittiin virheitä, tarkastettiin pakettien sisältöjä ja muokattiin koodin ulkoasua julkaistavaan kuntoon.

5.3 Säikeistetyn ja säikeistämättömän Dynamics-version ero

Säikeistämättömässä Dynamicsin versiossa kaikki toiminta suoritetaan ohjelman pääsäikeessä. Säikeistetyssä versiossa DHCP-kutsu on säikeistetty siten että viimeisessä kokeilussa `handle_req_message` on jaettu kahtia. Säikeistys alkaa DHCP-kutsusta ja loppuu `handle_req_message`-kutsuun.

5.4 Uusi tietorakenne

Kotiagenttiin `ha.c` lisättiin uusi tietorakenne, jonka avulla säilytetään tietoa kotiverkkoon rekisteröityneistä mobiililaitteista. Tietorakenteen avulla siirretään tietoa eri aliohjelmille. Dynamicsin alkuperäinen `binding`-tietorakenne sisältää hieman päällekkäistä tietoa toteutetun tietorakenteen kanssa, mutta Dynamicsin monimutkaisuuden vuoksi näitä kahta tietorakennetta ei yhdistetty.

Tietorakenne on lista C:n tietueista. Tietorakenne on seuraava:

```
struct dhcp_mobile {
    unsigned char *nai;
    int state;
    unsigned long requested_ip;
    unsigned long server_addr;
    unsigned long timeout;
    int packet_num;
    int fd;
    int signal_pipe[2];
    int nai_length;
    int lease;
    int spi;
};
```

Tietuelista luodaan rivillä

```
struct dhcp_mobile dhcp_mobile_array[HA_DEFAULT_MAX_BINDINGS]
```

Tietueen muuttujat kuvaavat seuraavia ominaisuuksia:

nai	on mobiililaitteen NAI.
state	on mobiililaitteen DHCP-tilakoneen standardin [17] mukainen tila.
requested_ip	on mobiililaitteen saama IP-osoite DHCP-palvelimelta.
server_addr	on mobiililaitteelle IP-osoitteen antaneen DHCP-palvelimen IP-osoite.
timeout	on DHCP-asiakkaaseen asetettu vakioaika, jonka se odottaa vastausta DHCP-palvelimelta.
packet_num	kertoo, kuinka monta DHCPDISCOVER-pakettia kukin DHCP-asiakas on lähettänyt.
fd	on DHCP-asiakkaan tarvitsema tiedostokuvaaja (engl. <i>file descriptor</i>), jota käytetään viestin lähettämiseen sokettiohjelmoinnissa.
signal_pipe	on DHCP-asiakkaan käyttämä viestiputki, johon viestit lähetetään.
nai_length	on mobiililaitteen NAI:n pituus.
lease	on mobiililaitteen IP-osoitteen "laina-aika" sekunneissa.
spi	on mobiililaitteen salausavain.
HA_DEFAULT_MAX_BINDINGS	on kotiagentin oletusarvo määrittäen, kuinka monta mobiililaitetta kotiagenttiin voi liittyä.

5.5 Tietorakennetta käsittelevät aliohjelmat

Tietorakenteen käsittelyyn liittyen lisättiin tiedostoon `ha.c` seuraavat aliohjelmat:

add_dhcp_mobile	lisää mobiililaitteen tietorakenteeseen. Sitä kutsutaan parametrein IP-osoite, NAI ja NAI:n pituus. Se palauttaa 1 lisäyksen onnistuessa ja 0 lisäyksen epäonnistuessa.
------------------------	---

- delete_dhcp_mobile** poistaa parametrilla määritetyn mobiililaitteen tietorakenteesta. Se palauttaa 1 poiston onnistuessa ja 0 poiston epäonnistuessa.
- find_dhcp_mobile** etsii mobiililaitteen tietorakenteesta. Sitä kutsutaan parametrein NAI ja NAI:n pituus. Se palauttaa laitteen paikan tietorakenteessa tai -1, jos laitetta ei löytynyt.

DHCP-keskustelu suoritetaan seuraavalla aliohjelmalla, joka on UDHCPC-asiakasohjelman vanha pääohjelma:

- udhcp** kutsutaan mobiililaitteen paikan tietorakenteessa osoittavalla parametrilla. Se palauttaa DHCP-palvelimelta saadun IP-osoitteen ja asettaa sen uuteen tietorakenteeseen.

5.6 Kotiagentin muutetut aliohjelmat ja tiedostot

Kotiagentin muokatut tiedostot, aliohjelmat ja niihin tehdyt muutokset on esitelty luvussa lyhyesti. Tarkempi kuvaus koko kotiagentin lähdekoodista löytyy Doxygenillä luodusta referenssimanuaalista [7] ja tehdyistä muutoksista lähdekoodin muutosdokumentista [15].

Tiedostoon `ha.h` on lisätty UDHCPC:tä varten:

- UDHCPC:n käyttämät DHCP-standardin mukaiset tila- ja viestivakiot,
- UDHCPC:n käyttämä `client_config`-tietorakenne ja
- vakiot `RENEWIP`, `REQUESTIP` sekä `RELEASEIP`, jotka kuvaavat ohjelman tiloja.

Tiedostoon `ha.c` on lisätty UDHCPC:tä varten:

- globaali tietorakenne `dhcp_mobile_array`,
- UDHCPC:n käyttämä tietorakenne `client_config`,
- UDHCPC:n käyttämät vakiot ja

- lähes kokonaan UDHCPC:n tiedoston `udhcp.c` lähdekoodi tiedoston alkuun.

Tiedoston `ha.c` aliohjelmaan `handle_reg_msg` on lisätty

- mobiililaitteen poistumispyynnön käsittelykohtaan DHCPRELEASE-viestin lähetys `udhcp`-aliohjelmalla ja ko. mobiililaitteen poisto tietorakenteesta oikeasta kohtaa sekä
- mobiililaitteen rekisteröitymispyynnön käsittelykohtaan mobiililaitteen lisääminen tietorakenteeseen ja IP-osoitteen hakeminen `udhcp`-aliohjelmalla.

Tiedoston `ha.c` aliohjelmasta `validate_request` on poistettu

- IP-osoitteen tarkastaminen, jotta kotiagentti hyväksyy rekisteröintipyynnöt IP-osoitteesta `0.0.0.0`.

Tiedoston `ha.c` aliohjelmasta `udhcp` on muutettu

- UDHCPC-asiakasohjelman pääohjelma `udhcp`-aliohjelmaksi osaksi kotiagenttia ja
- käsittelemään tietoja tietorakenteesta alkuperäisen yksittäisen toteutuksen globaalien attribuuttien sijaan.

Tiedoston `ha.c` pääohjelmassa `main` on siirretty

- `udhcp_if`-tieto kotiagentin asetustietorakenteesta UDHCPC:n asetustietorakenteeseen.

Tiedostoon `ha_config.h` on lisätty

- `config`-tietorakenteeseen `dhcp_if`-muuttuja, johon asetetaan DHCP-keskustelun suorittavan verkkokortin nimi.

Tiedoston `ha_config.c` aliohjelmaan `process_load_ha` on lisätty

- UDHCPC-parametrin lukeminen `dynhad.conf`-tiedostosta `dhcp_if`-muuttujaan.

Tiedostoon `Makefile` on lisätty

- viitteet UDHCN:n tarvitsemiin tiedostoihin `options.h`, `client_packet.h`, `packet.h`, `script.h`, `socket.h` ja `pidfile.h` sekä em. tiedostojen `c-` ja `o-`tiedostoihin.

5.7 Kotiagenttiin lisätyt tiedostot

Kotiagentin lähdekoodihakemistoon `.../src/ha/` lisättiin UDHCN:n lähdekooditiedostot, header-tiedostot ja object-tiedostot seuraavista tiedostoista `options`, `client_packet`, `packet`, `script`, `socket` ja `pidfile`.

UDCHP-asiakasohjelman lähdekooditiedosto `udhpc.c` liitettiin lähes sellaisenaan osaksi kotiagentin `ha.c`-tiedostoa.

6 Virtuaalisen vierasagentin toteutusratkaisut

Virtuaalinen vierasagentti saatiin lähettämään useita liittymis- ja poistumisviestejä verkkoon asetustiedostossa määritettyjen tapahtumien mukaisesti. Luvussa esitellään virtuaalisen vierasagentin toteutusratkaisut.

6.1 Tekninen toteutus

Virtuaalinen vierasagentti toteutettiin muokkaamalla Dynamicsin mobiililaitetta. Virtuaaliseen vierasagenttiin toteutettiin rekisteröintipyynnön ja poistumispyynnön lähettäminen sekä tietorakenne ja tiedostoluku.

Mobiililaitteen pääohjelma oli aluksi eri tiedostossa, josta normaalisti lähetetään yhden mobiililaitteen viestit (`mn_reg.c`). Mobiilin laitteen pääohjelma siirrettiin tiedostoon `mn_reg.c`. Samaan tiedostoon rakennettiin tietorakenne "virtuaalisille mobiililaitteille" ja tietorakenteen käsittely.

Virtuaaliset mobiililaitteet luetaan tietorakenteeseen tiedostosta `mobiles.txt`, joka sisältää muuttuvat tiedot kutakin mobiililaitetta kohden. Kaikille virtuaalisille mobiililaitteille samat tiedot luetaan Dynamicsin alkuperäisestä asetustiedostosta `dynamnd.conf`

Tarkempi kuvaus koko virtuaalisen vierasagentin lähdekoodista löytyy Doxygenillä luodusta referenssimanuaalista [8] ja lähdekoodin muutosdokumentista [15].

6.2 Kehitysvaiheet

Ensimmäisessä vaiheessa virtuaalinen vierasagentti saatiin **lähettämään yhden liittymispyynnön verkkoon**. Tämän toteutusidean pohjalta rakennettiin toiminta, jossa useita liittymis- ja poistumispyyntöjä lähetetään verkkoon tietorakenteen avulla.

Toisessa vaiheessa luotiin **tiedoston käsittely**, jossa virtuaalisten mobiililaitteiden tiedot luetaan asetustiedostosta ja tallennetaan tietorakenteeseen. Toteutuksen jälkeen virtuaalinen vierasagentti lähetti käyttäjän syöttämällä arvoilla liittymis- ja poistumispyyntöjä verkkoon.

6.3 Kääntäminen

Käännettäessä virtuaalinen vierasagentti, on muistettava muuttaa muutama muuttuja tiedostoista. Ohjelma ei käänny ilman muutoksia. Tarkemmat ohjeet muutoksille ja kääntämisen on kuvattu asennusohjeessa [16].

6.4 Uudet tietorakenteet

Tiedostoon `mn_reg.c` lisättiin tiedoston lukuun oma ja virtuaalisille mobiililaitteille oma tietorakenne. Tietorakenteet ovat listoja C:n tietueista. **Virtuaalisten mobiililaitteiden tietorakenne** on seuraava:

```
struct mobile{
    struct mn_data mn;
    struct mn_config config;
    int state;
};
```

Tietorakenne muodostetaan seuraavalla rivillä

```
struct mobile mobile_array[500];
```

Lisäksi virtuaalisten mobiililaitteiden lukemiseen liittyy vielä `mobiles`-niminen globaali muuttuja kertoen, kuinka monta mobiililaitetta luotiin.

Tiedoston luvussa käytettävä tietorakenne on seuraava:

```
struct device {
    char ip_home[17];
    char ip_homeagent[17];
    char ip_careoff[17];
    char nai[32];
    char timestart[3];
    char state[2];
};
```

Laitteiden tiedoista luodaan tietorakenne rivillä

```
struct device devices[500];
```

Tiedoston lukuun liittyy vielä `rownumber`-niminen globaali muuttuja kertoen, kuinka monta riviä tiedostosta luettiin.

Tietuiden muuttujat kuvaavat seuraavia ominaisuuksia:

mn	sisältää Dynamicsin mobiililaitteen tarvitsemat perusasetustiedot.
config	sisältää Dynamicsin mobiililaitteen tarvitsemat perusasetustiedot. Samat tiedot haetaan kahdesta paikasta (ks. <code>mn</code>).
state	on virtuaalisen mobiililaitteen tilan muutos, jonka arvo 0 vastaa eroa verkosta, 1 verkkoon liittymistä ja 2 kolmannen osapuolen ohjelman suorittamista.
ip_home	on virtuaalisen mobiililaitteen kotiosoite.
ip_homeagent	on virtuaalisen mobiililaitteen kotiagentin osoite.
ip_careoff	on virtuaalisen mobiililaitteen tilapäisosoite.
nai	on virtuaalisen mobiililaitteen NAI.
timestart	on aika sekunteina, jolloin virtuaalinen mobiililaitte tekee tilan <code>state</code> mukaisen toiminnon.

6.5 Asetustiedosto

Virtuaalisten mobiililaitteiden tiedot ja niihin liittyvät parametrit luetaan sekä rekisteröinti- että poistumistapauksissa tiedostosta `mobiles.txt`. Sen tietokioiden välimerkkinä toimii välilyönti, ja riveillä esitettävät tapahtumat erotetaan toisistaan rivivaihdolla. Kunkin rivin parametrit ovat aika sekunneissa, toiminto (0=eroa verkosta, 1=liity verkkoon ja 2=kolmannen osapuolen ohjelma), kotiosoite, kotiagentin osoite, tilapäisosoite ja NAI.

Asetustiedostosta näyttää seuraavalta:

```
1 1 192.168.0.45 192.168.0.1 172.168.0.1 zormal@jyu.fi
2 1 192.168.0.25 192.168.0.1 172.168.0.1 zorma2@jyu.fi
3 2 thirdpartyprogram --parameter
5 0 192.168.0.45 192.168.0.1 172.168.0.1 zormal@jyu.fi
10 0 192.168.0.35 192.168.0.1 172.168.0.1 zorma2@jyu.fi
```

Yo. mukainen ohjelman ajo suorittaisi aluksi kahden laitteen liittymisviestin lähettämisen. Tämän jälkeen suoritetaan kolmannen osapuolen ohjelma. Ajan hetkellä 5 lähetetään ensimmäisen laitteen poistumisviesti ja siitä 5 (10-5) sekunnin päästä lähetetään toisen laitteen poistumisviesti.

6.6 Lisätyt aliohjelmat

Tiedostoon `mn_reg.c` lisättiin **tiedoston lukua** varten seuraavat aliohjelmat:

readfile	lukee parametrilla määritetyn asetustiedoston sisällön rivi kerrallaan taulukkoon. Se palauttaa 1 tiedoston luvun onnistuessa ja -1 epäonnistuessa.
parse_rows	lukee määritetyiltä riveiltä arvot muuttujiin. Se palauttaa 1 tiedoston rivien luvun onnistuessa ja -1 epäonnistuessa.

Tiedostoon `mn_reg.c` lisättiin **tietorakenteen käsittelyä ja viestien lähettämistä** varten seuraavat aliohjelmat:

mn_init	lukee Dynamicsin mobiililaitteen perusasetustiedot. Se palauttaa 1 tehtävän onnistuessa ja -1 epäonnistuessa.
create_mobile	luo asetustiedoston mukaiset virtuaaliset mobiililaitteet <code>mobile_array</code> -tietorakenteeseen.
send_messages	lähettää määritellyn aikavälin ja virtuaalisen mobiililaitteen tilan mukaisesti joko rekisteröitymis- tai poistumisviestin.

Tiedostoon `mn_reg.c` on lisätty seuraava **uusi pääohjelma**:

main	alustaa laitteet ja lähettää asetustiedoston mukaiset viestit.
-------------	--

Tiedostoon `net/wireless.h` on lisätty tiedoston alkuun määrittely

```
ifnamsiz      define ifnamsiz 16, jotta ohjelma kääntyisi.
```

6.7 Muutetut aliohjelmat

Tiedostossa `mn.c` on muutettu aliohjelma

```
main          vaihtamalla sen nimeksi main_old.
```

Tiedostossa `dyn_wireless.c` on vaihdettu muuttujan

```
ifr_name      nimeksi u.name johtuen tietueen iwreq muuttumisesta.
```

7 Suoritetut testaukset ja tulokset

Sovellusta testattiin koko kehityshistorian ajan hyödyntäen tekstipohjaista GDB-debuggeria ja WireShark-pakettikaapparia. Testaussuunnitelman mukaan toteutussa testauksissa tuli esiin ongelmia, joiden syyt selvitettiin ja osa ongelmista ratkaistiin.

Luvussa kuvataan DHCP-tuen ja virtuaalisen vierasagentin suoritettua testausta sekä tuloksia ja havaittuja virheitä Dynamo-verkossa ja tilaajan verkossa.

7.1 Testaus toteutusvaiheessa

Dynamics-protokollaa testattiin Dynamo-verkossa kunkin lähdekoodin muutoksen jälkeen. Yksikkö- ja integrointitestausta suoritettiin koko sovelluskehityksen ajan, joten yllätyksiä järjestelmätestauksen testauskerroilla ei tullut.

Dynamicsin kotiagentin, vierasagentin ja mobiililaitteen toimivuutta ei voitu testata irrallisina, koska Dynamics-projektista puuttuivat projektitiedostot, joita tarvitaan ohjelmakoodin kääntämiseen kehitystyökalussa. Projektitiedostojen puuttumisen

vuoksi testausverkon pystyttäminen ja jatkokehittäminen suoritettiin testausverkossa. Testausverkon rakenne ja testitapaukset on esitelty tarkemmin testaussuunnitelmassa [12].

Sovellusta testattiin koko kehityshistorian ajan hyödyntäen tekstipohjaista GDB-debuggeria ja WireShark-pakettikaapparia. Testausympäristön ongelmana oli muutujien arvojen lukeminen. Se oli työlästä ja hidasta komentoriviargumenttien pohjalta toimivalla GDB-debuggerilla. Apuna käytettiin välitulostuksia, joiden avulla selvitettiin sovelluksen lähdekoodin toimintaa ja rakennetta.

7.2 Testaus projektin testausympäristössä

DHCP-tuen testaukset projektin testausympäristössä suoritettiin yksikkö-, integrointi- ja järjestelmätestauksella testaussuunnitelmassa [12] määritetyillä testitapauksilla. Testaus suoritettiin liittämällä kahta mobiililaitetta verkkoon ja poistamalla niitä verkosta.

Kumpikin kone sai IP-osoitteen kotiagentilta, jos liittymispyynnöt tehtiin eri aikoihin. Kun laitteet liitettiin verkkoon nopeasti peräkkäin, vain toinen laite sai osoitteen. Em. ongelmaksi paljastui vierasagentin toiminta. Se ei pysynyt liittymispyyntöjen vauhdissa mukana. DHCP-tuen testitapauksien tulokset on esitelty tarkemmin testausraportissa [9].

Virheitä esiintyi myös testausverkon reititystaulujen asetuksissa. Testausympäristön vierasagentti ei lähettänyt paketteja oikein eteenpäin kotiverkosta vierasverkkoon tai vierasverkosta kotiverkkoon. Vierasagentti pystyi kuitenkin lähettämään dataa mobiililaitteille ja mobiililaitteet vierasagentille. Tekninen ohjaaja oli ratkomaan em. ongelmaa, mutta ratkaisua ei löytynyt. Ongelma saattoi olla myös vierasagentin asetuksissa. Vastaavaa ongelmaa ei havaittu testattaessa DHCP-tukea tilaajan verkossa.

Kolmas ongelma ilmeni mobiililaitteen poistuessa verkosta. DHCP-tuki generoi oikeanlaisen DHCPRELEASE-viestin ja lähetti sen verkkoon, mutta viesti ei löytänyt perille. Vastaava ongelma havaittiin myös testattaessa DHCP-tukea tilaajan verkossa.

Virtuaalisen vierasagentin testauksessa ei ilmennyt ongelmia. Virtuaalisen vierasagentin testitapauksen tulokset on luettavissa testausraportista [11].

7.3 DHCP-tuen testaus tilaajan verkossa

Testauksessa tilaajan verkossa haluttiin erityisesti selvittää DHCP-tuen toiminnot, jotka aiheuttivat virheitä projektin testausympäristössä testattaessa (ks. luku 7.2). Virtuaalista vierasagenttia ei testattu, koska sen toiminnot osoittautuivat luotettavasti toimiviksi jo Dynamo-verkossa testattaessa.

Ensimmäisellä testauksella tilaajan verkossa Dynamicsin asentaminen verkon laitteisiin aiheutti ongelmia. Ongelmina olivat Dynamicsin kääntäminen ja `makefile` n tekeminen. Ensimmäisen testauksella ongelma paljastui väärä SPI-arvo mobiililaitteessa ja väärät asetukset verkon reititystauluissa. Ongelma saatiin ratkaistua, mutta testitapauksia ei ehditty suorittaa tuolloin. Kyseisten ongelmien välttämiseksi jatkossa kirjoitettiin Dynamicsin asennusohjeet [14].

Toisella testauksella työtä tehti verkon oikeiden asetusten löytäminen. Ongelmana oli Ciscon mobiililaitteen asiakasohjelman toiminta. Asiakasohjelma asennettiin uudelleen puhtaalle koneelle ja testitapaukset saatiin suoritettua.

DHCPRELEASE-paketin lähettäminen ei onnistunut tilaajankaan verkossa, ja ongelmaksi havaittiin vika lähdekoodissa. Paketin lähettäjäksi asetetaan mobiililaitteen osoite, vaikka sen pitäisi olla kotiagentin osoite. Kyseisen virheen korjaaminen sovittiin tilaajan kanssa projektin jälkeiseen jatkokehitykseen. Muutama testitapaus jäi suorittamatta, koska virheellisillä tiedoilla varustettuja paketteja ei saatu luotua tilaajan verkossa. Testaustulokset on luettavissa toisen testauskerran dokumentista [?].

8 Jatkokehitysideoita

Luvussa esitellään DCHP-tuen ja virtuaalisen vierasgentin jatkokehitysideoita. Jatkokehitysideat periytyvät pääosin vaatimusmäärittelyn [13] niistä vaatimuksista, joita Dynamo-projektissa **ei toteutettu**. Tärkeimmäksi jatkokehityskohteeksi katsottiin säikeistuksen luominen DHCP-keskusteluun. Myös DHCPRELEASE-viestin lähettäminen oikeilla tiedoilla DHCP-palvelimelle on välttämätöntä toteuttaa, jotta DHCP-palvelimen tiedot pysyvät ajan tasalla.

8.1 DHCP-tuen jatkokehitys

DHCP-kutsun ajaminen omassa säikeessä nopeuttaisi sovelluksen muuta toimintaa, sillä kotiagentin toiminta ei keskeytyisi DHCP-kutsun ajaksi. Ilman säikeistystä virtuaalista vierasagenttiakaan ei voida käyttää kuormituksen testaustyökaluna, jos kotiagenttina on Dynamicsin kotiagentti.

DHCP-keskustelun säikeistys ei toteutunut, koska Dynamicsissa suurin osa järjestelmän sovelluksen muuttujista on omissa aliohjelmissa, eikä globaaleina muuttujina. Tällöin Dynamicsiin tulleiden pakettien sisältöön pääsee käsiksi vain aliohjelmasta, jossa pakettien sisällön käsittely on alunperin toteutettu. Dynamicsin pääohjelmasta ei siten pääse käsiksi kaikkiin tarvittaviin muuttujiin, joiden tietoja säikeistuksen luominen vaatisi. Aliohjelmien muuttujien kopiointi pääohjelmaan ei onnistunut, koska aliohjelmien tietorakenteiden sisällä oli osoittimia toisiin tietorakenteisiin, joissa taas oli osoittimia edelleen toisiin tietorakenteisiin. Tietorakenteita Dynamicsin kotiagentissa on kymmeniä.

Em. ongelman olisi voinut ratkaista lisäämällä jokaiseen aliohjelmaan funktion, joka kopioi aliohjelman lokaalit muuttujat globaaleihin muuttujiin. Tämän idean pohjalta koko Dynamicsin perusrakennetta olisi pitänyt muuttaa. Muutos olisi johtanut siihen, että jokainen globaalien muuttujien käsittely oli pitänyt suojata lukoilta. Säikeistuksen toteutus tämän idean pohjalta nähtiin liian työlääksi toteuttaa Dynamo-projektissa.

Toinen DHCP-tukee liittyvä jatkokehitys on **DHCPRELEASE -viestin lähettäminen oikeilla tiedoilla DHCP-palvelimelle**. DHCPRELEASE -paketin lähettäjäksi asetetaan Dynamo-projektin toteutuksessa mobiililaitteen osoite, vaikka sen pitäisi olla kotiagentin osoite.

8.2 Muut kotiagentin jatkokehitysideat

Kotiagentin jatkokehitysideoita on kaksi. Dynamicsin kotiagentin tulisi antaa mobiililaitteelle **kotiosoite dynaamisesti tai staattisesti** riippuen mobiililaitteen kotiosoitteentästä. Kotiagentin asetustiedostosta voisi valita, käytetäänkö staattista vai dynaamista kotiosoitetta.

Mobiililaitteen NAI tulisi sitoa kotiagentissa määriteltävään SPI-arvoon. Ratkaisu lisäisi Dynamicsin tietoturva.

8.3 Virtuaalisen vierasagentin jatkokehitys

Virtuaalisen vierasagentin tärkein jatkokehitystavoite on, että virtuaalinen vierasagentti **lukisi asetukset komentoriviargumenteista** ja muodostaisi niiden pohjalta asetustiedoston. Vaatimuksen toteutuminen nopeuttaisi virtuaalisen vierasagentin käyttöä. Lisäksi virtuaaliseen vierasagenttiin pitäisi toteuttaa toiminto, jossa asetustiedoston avulla voitaisiin **käynnistää kolmannen osapuolen ohjelmia**, joilla kuormitettaisiin verkkoa.

Virtuaaliseen vierasagentin pitäisi myös **kerätä tilastotietoja** lähetetyistä ja vastaanotetuista viesteistä. Tilastotietojen keruun toteutuessa virtuaalinen vierasagentti soveltuisi IPv4-verkon erinomaisesti tietoliikenneasiantuntijoiden testaustyökaluksi. Em. ominaisuuden toteuttaminen vaatii pakettien vastaanoton muokkaamista Dynamicsin mobiilin päätelaitteen lähdekoodissa.

8.4 Dynamicsin rakenne ja kommentointi

Verrattaessa Dynamicsin ohjelmakoodia UDHCPC-koodiin, **Dynamicsin rakenne on sekava**. Rakenteen uudelleen järjestäminen voisi selkeyttää säikeistyksen luomista Dynamicsin kotiagenttiin. Dynamicsin rakenne olisi muokattava siten, että Dynamicsin protokollan pakettien lähetys- ja vastaanotto olisi erillään DHCP-käsittelystä ja DHCP-vastausviestin lähettämisestä.

Dynamicsin kotiagentin viestinvälitys kannattaisi eriyttää muusta Dynamicsin toiminnasta. Eriyttämisen avulla kotiagentin lähdekoodi saataisiin omiin kokonaisuuksiin ja siten paremmin muunneltaviksi.

Dynamicsin ohjelmakoodia ole kommentoitu kattavasti, eikä ohjelman rakennetta ole dokumentoitu riittävästi. **Koodin kommentoinnin puutteet** ja ohjelman sekava rakenne aiheuttavat varmasti ongelmia jatkokehityksessä.

Dynamicsin jatkokehitystä voidaan suositella henkilöille, joilla on osaamista C-kielstä ja säikeiden toteuttamisesta C-ympäristössä. C-kielen käyttö aiheuttaa työtä sellaisissa ongelmissa, jotka kehittyneimmissä ohjelmointikielissä vaativat vain hetken työn. Mm. osoittimet ja tietueiden kopioinnit ovat C-kielessä työläitä käsitellä.

9 Yhteenveto

Dynamo-projektin suunnittelemien ja toteuttamien Dynamics-järjestelmän muutosten jälkeen se vastaa paremmin Jyväskylän yliopiston tietotekniikan laitoksen ja LaiLa-projektin tarpeita. Sovellus toteutettiin Linux-alustalla ja C-kielellä olemassaolevan Dynamics-järjestelmän ja UDHCP-ohjelman lähdekoodia hyödyntäen.

Projektin tärkein tavoite DHCP-tuen lisääminen kotiagenttiin saatiin toteutettua säikeistystä lukuunottamatta. Virtuaalinen vierasagentti oli tavoitteena toteuttaa ainakin vaatimusten määrittelyn ja suunnittelun osalta sekä ohjelmoida ajan salliessa. Virtuaalinen vierasagentti saatiin lähettämään verkkoon useita liittymis- ja poistumisviestejä asetustiedostossa määriteltyjen tapahtumien mukaisesti.

Järjestelmän jatkokehitetyt osat testattiin kokonaisuudessaan projektin testausympäristössä ja DHCP-tuki tilaajan verkossa. Testauksessa huomattiin DHCPRELEASE-viestin lähetysvirhe, joka lisättiin jatkokehitysideoihin. Tärkeimmäksi Dynamicsin jatkokehityskohteeksi katsottiin säikeistyksen luominen DHCP-keskusteluun, mikä avulla kotiagentin toimintaa voidaan nopeuttaa.

Lähteet

- [1] Andersson B., "Dynamics, Technical Definition", Helsinki University of Technology, April 1999.
- [2] Droms R., "Dynamic Host Configuration Protocol", RFC 1531 , Bucknell University, October 1993.
- [3] Droms R., "Dynamic Host Configuration Protocol", RFC 2131, Bucknell University, November 1997.
- [4] Häätinen T., Pyykkönen A. ja Purojärvi J., "Dynamo-projekti, Projektiraportti", Jyväskylän yliopisto, tietotekniikan laitos, joulukuu 2007.
- [5] Häätinen T., Pyykkönen A. ja Purojärvi J., "Dynamo-projekti, Projektisuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, lokakuu 2007.
- [7] Häätinen T., Pyykkönen A. ja Purojärvi J., "Dynamo Reference Manual for Home Agent", generated with Doxygen , University of Jyväskylä, Department of Mathematical Information Technology, December 2007.
- [8] Häätinen T., Pyykkönen A. ja Purojärvi J., "Dynamo Reference Manual for Virtual Foreign Agent", generated with Doxygen , University of Jyväskylä, Department of Mathematical Information Technology, December 2007.
- [9] Häätinen T., Pyykkönen A. ja Purojärvi J., "Dynamo-projekti, Testausraportti, kotiagentin testaus Dynamo-verkossa", Jyväskylän yliopisto, tietotekniikan laitos, joulukuu 2007.
- [10] Häätinen T., Pyykkönen A. ja Purojärvi J., "Dynamo-projekti, Testausraportti, kotiagentin testaus tilaajan verkossa", Jyväskylän yliopisto, tietotekniikan laitos, joulukuu 2007.
- [11] Häätinen T., Pyykkönen A. ja Purojärvi J., "Dynamo-projekti, Testausraportti, virtuaalisen vierasagentin testaus Dynamo-verkossa", Jyväskylän yliopisto, tietotekniikan laitos, joulukuu 2007.
- [12] Häätinen T., Pyykkönen A. ja Purojärvi J., "Dynamo-projekti, Testaussuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, joulukuu 2007.
- [13] Häätinen T., Pyykkönen A. ja Purojärvi J., "Dynamo-projekti, Vaatimusmäärittely", Jyväskylän yliopisto, tietotekniikan laitos, marraskuu 2007.

- [14] Hätinén T., Pyykkönen A. ja Purojärvi J., "How to Install Dynamics 0.8.1.dynamo.1", University of Jyväskylä, Department of Mathematical Information Technology, December 2007.
- [15] Hätinén T., Pyykkönen A. ja Purojärvi J., "Changes in Dynamics ver 0.8.1.dynamo.1", University of Jyväskylä, Department of Mathematical Information Technology, December 2007.
- [16] Teknillinen korkeakoulu, Tietoliikennearkkitehtuurit-kurssin materiaali, 2003.
- [17] Perkins C., "IP Mobility Support", RFC 2002, IBM, October 1996.
- [18] Perkins C., "IP Mobility Support for IPv4", RFC 3344, Nokia Research Center, August 2002.