

Pekka Pääkkönen

Kodin verkotettujen laitteiden palveluiden hyödyntäminen

Kodin verkotettujen laitteiden palveluiden hyödyntäminen

Pekka Pääkkönen
VTT Elektronikka

ISBN 951-38-6073-6 (URL: <http://www.inf.vtt.fi/pdf/>)
ISSN 1455-0865 (URL: <http://www.inf.vtt.fi/pdf/>)

Copyright © VTT 2002

JULKAISIJA – UTGIVARE – PUBLISHER

VTT, Vuorimiehentie 5, PL 2000, 02044 VTT
puh. vaihde (09) 4561, faksi (09) 456 4374

VTT, Bergsmansvägen 5, PB 2000, 02044 VTT
tel. växel (09) 4561, fax (09) 456 4374

VTT Technical Research Centre of Finland, Vuorimiehentie 5, P.O.Box 2000, FIN-02044 VTT, Finland
phone internat. + 358 9 4561, fax + 358 9 456 4374

VTT Elektronikka, Kaitoväylä 1, PL 1100, 90571 OULU
puh. vaihde (08) 551 2111, faksi (08) 551 2320

VTT Elektronik, Kaitoväylä 1, PB 1100, 90571 ULEÅBORG
tel. växel (08) 551 2111, fax (08) 551 2320

VTT Electronics, Kaitoväylä 1, P.O.Box 1100, FIN-90571 OULU, Finland
phone internat. + 358 8 551 2111, fax + 358 8 551 2320

Toimitus Leena Ukaskoski

Espoo 2002

Tiivistelmä

Diplomityö käsittelee kodin verkotettujen laitteiden palveluiden hyödyntämistä kotiympäristön ulkopuolelta. Palveluiksi käsitetään mikä tahansa kodin laitteen tarjoama tieto, jota ulkopuolinen käyttäjä voi hyödyntää. Tämän tiedon käsittely aiheuttaa usein toimintoja kontrolloitavassa laitteessa. Suurimmat ongelmat aiheeseen liittyen ovat lukuisat kodin laitteiden verkotukseen käytettävät teknologiat ja laitteiden kontrolloinnin turvallisuus. Työn tavoitteena oli toteuttaa kotona olevien yksinkertaisten laitteiden palveluiden hyödyntäminen kodin ulkopuolelta käyttäen liikkuvaa päätelaitetta.

Työssä tutkittiin SIP- ja OSGi-teknologioiden soveltuvuutta tähän aihepiiriin. Tutkimusta rajattiin siten, että palveluiden hyödyntämisen turvallisuuteen, tunnistukseen ja luotettavuuteen ei otettu kantaa ja SIP:n osuutta painotettiin suhteessa OSGi:hin. Näihin teknologioihin liittyen tutkittiin laitteiden nimeämistä, osoitteistusta, liikkuvuutta, käyttöliittymiä, teknologiariippumattomuutta, kommunikaatiovaatimuksia, skaalautuvuutta ja läsnäoloa. Tätä varten toteutettiin järjestelmä, jonka avulla oli mahdollista käyttää kodin lampun ja hälytysjärjestelmän tarjoamia palveluita liikkuvalla päätelaitteella. Ohjelmisto toteutettiin Java-kielellä sen alustariippumattomuuden ja verkkotuen vuoksi. Järjestelmän toimintaa demonstroidaan viidellä käyttötapauksella, jotka havainnollistavat kyseisten laitteiden käyttöliittymän lataamista ja sen jälkeen tapahtuvaa kommunikointia. Käyttötapausten kulkua kuvataan SIP-viestikaavioilla ja ohjelmiston rakenne selitetään UML-menetelmää apuna käyttäen.

Lopuksi suoritettun evaluoinnin perusteella todettiin SIP:n olevan helposti laajennettava ja tukevan laitteiden liikkuvuutta ja nimeämistä. Se soveltui myös hyvin kommunikointivaatimusten toteuttamiseen. OSGi:n parhaaksi puoleksi huomattiin sen toimivuus kodin ja ulkopuolisen Internetin välisenä dynaamisena palvelualustana.

Pääkkönen, Pekka. Kodin verkotettujen laitteiden palveluiden hyödyntäminen [The utilization of services provided by networked home appliances]. Espoo 2002. VTT Tiedotteita – Research Notes 2157. 69 p.

Keywords SIP, OSGi, service gateway, presence

Abstract

This diploma thesis deals with the utilization of services provided by networked home appliances from outside of the home environment. Services are considered to be any information provided by a home appliance which the outside user can utilize. The processing of this information often produces functions in the controlled home devices. The biggest problems related to this subject are the wide variety of technologies used to network home appliances and the safety related to the control of these devices. The goal of this thesis is to implement the utilization of services from outside of the home environment offered by simple home networked appliances using a nomadic terminal.

The applicability of SIP- and OSGi-technologies are studied in relation to this subject matter. The research is limited in that it does not deal with security, identification and the reliability of the utilized services and the stress on SIP's part in this study is emphasized in its relation to OSGi. Related to these technologies research is conducted on the naming, addressing, mobility, user interfaces, technological independence, requirements for communication, scalability and presence of various devices. To enable this, a system was implemented with which services provided by a home's lamp and alarm system were able to be utilized with a nomadic terminal. Software was implemented with the Java programming language because of its support for platform independence and networking. The operation of the system was demonstrated with five use cases which illustrated the downloading of the user interfaces of the devices in question and the communication which takes place after it. The progress of the use cases is illustrated with the aid of SIP message charts and the structure of the software is depicted with UML.

Based on an evaluation performed at the end, SIP was recognized to be easily expandable and to support the mobility and naming of the devices. It was also found to be suitable for the implementation of the communication requirements. OSGi's best feature was noticed to be its functionality as a dynamic service gateway between the home and the outside Internet.

Alkusanat

Tämän diplomityön tarkoituksena oli tutkia SIP- ja OSGi-teknologioiden soveltuvuutta kodin verkotettujen laitteiden palveluiden hyödyntämiseen. Työ on tehty VTT Elektronikan MIDAS-projektissa (Middleware for Embedded Mobile Distributed Applications), joka oli osa PLA-strategiaohjelmaa (Product Line Architectures).

Tämän työn tekemistä on ohjannut Juhani Latvakoski, jota haluan kiittää ammattitaidosta, luottamuksesta ja kärsivällisyydestä. Kiitos kuuluu myös työn valvojalle Juha Röningille ja toiselle tarkastajalle Jukka Riekille sekä kaikille muille tutkijoille, jotka ovat auttaneet minua työn valmistumisessa.

Oulussa 4.3.2002

Pekka Pääkkönen

Sisällysluettelo

Tiivistelmä.....	3
Abstract.....	4
Alkusanat.....	5
Lyhenteiden ja merkkien selitykset.....	8
1. Johdanto.....	11
1.1 Tavoitteet ja rajaus.....	11
2. Kotiverkko ja verkotetut laitteet.....	13
2.1 Kodin teknologiat.....	14
2.2 Palveluyhdyskäytävä.....	15
2.3 Verkotettu laite.....	16
2.4 Verkotetun laitteen tarjoamat palvelut.....	17
2.5 Alan viimeisin kehitys.....	19
3. SIP- ja OSGi-teknologiat.....	21
3.1 SIP.....	21
3.1.1 SIP-käyttäjäagentti.....	22
3.1.2 SIP-palvelimet.....	23
3.1.3 SIP-esimerkki.....	24
3.1.4 SIP-viestit.....	25
3.2 SIP-laajennukset verkotetuille laitteille.....	29
3.2.1 DMP.....	29
3.2.2 DO.....	31
3.2.3 Läsnäolotieto.....	31
3.3 OSGi.....	34
3.3.1 Kehys.....	34
3.3.2 Nippu.....	36
3.3.3 Ydinpalvelut.....	37
4. Järjestelmä ja käyttötapaukset.....	38
4.1 Järjestelmä.....	38
4.2 Ohjelmiston arkkitehtuuri.....	39
4.3 Käyttötapaukset.....	40
4.3.1 Laitteen fyysinen osoittaminen.....	41
4.3.2 Rekisteröityminen ja käyttöliittymän lataaminen.....	42
4.3.3 Kontrollointi.....	45

4.3.4	Tiedustelu	47
4.3.5	Kirjautuminen ja ilmoituksen saaminen.....	48
5.	Toteutus	52
5.1	SIP-viestit	53
5.2	Kommunikointi	54
5.3	Sovellusrajapinta	55
5.4	DMP	57
5.5	Tila- ja kirjautumispalvelu	57
5.6	SIP-sovellukset	58
5.7	Käyttöliittymät.....	58
5.8	Tiedostojen rekisteröinti.....	59
6.	Evaluointi.....	60
6.1	Nimeäminen ja osoitteistus	60
6.2	Liikkuvuus.....	60
6.3	Käyttöliittymä.....	61
6.4	Teknologiariippumattomuus	61
6.5	Kommunikointivaatimukset	62
6.5.1	Läsnäolotieto	62
6.6	Skaalautuvuus.....	62
6.7	Järjestelmän suhde alan viimeisimpään kehitykseen	63
6.8	SIP:n soveltuvuus	63
6.8.1	SIP ja HTTP	64
6.9	OSGi:n soveltuvuus.....	64
7.	Yhteenveto	65
	Lähdeluettelo	66

Lyhenteiden ja merkkien selitykset

3GPP	Third Generation Partnership Program, kolmannen sukupolven matkapuhelinverkkojen standardointijärjestö
API	Application Programming Interface, sovellusrajapinta
CEA	Consumer Electronics Association, kulutuselektronikkaan keskittynyt järjestö
DNS	Domain Name Server, nimipalvelin
DMP	Device Messaging Protocol, XML-pohjainen laitteiden kanssa kommunikointiin käytettävä protokolla
DSL	Digital Subscriber Line, Internet-yhteys
GSM	Global System for Mobile Communications
HAVI	Home Audio/Video Interoperability, kotiverkotusteknologia
HTTP	Hyper Text Transfer Protocol, sovelluserroksen protokolla
IEEE	Institute of Electrical and Electronics Engineers, standardointijärjestö
IETF	Internet Engineering Task Force, Internetin standardointijärjestö
IP	Internet Protocol, Internet-protokolla
ISUP	ISDN User Part, SS7-merkinantokanavan ISDN-käyttjäosa
IrDA	Infrared Data Association, tiedonsiirtoon infrapuna-aaltojen avulla keskittynyt organisaatio
JAR	Java Archive, Javan tiedostotyyppi
JINI	Jini Is Not Initials, laitteiden rekisteröitymiseen ja löytämiseen käytettävä Java-pohjainen teknologia
JPG	Joint Photographic Experts Group, kuvatiedostojen pakkaamiseen keskittynyt järjestö

LAN	Local Area Network, paikallisverkko
LON	Local Operating Network, kodin hajautettu automaatioverkko
NAT	Network Address Translator, Internet-osoitteiden muunnos
OSGi	Open Services Gateway Initiative, väliohjelmistoratkaisu
PC	Personal Computer
PDA	Personal Digital Assistant, henkilökohtainen langaton laite
RF	Radio Frequency, radiotaajuus
RFB	Remote Frame Buffer, käyttöliittymien etäkäyttöön keskittyvä protokolla
RGW	Residential Gateway, paikallinen yhdyskäytävä
SDK	Software Development Kit, ohjelmistojen kehitysympäristö
SDP	Session Description Protocol, verkkoistuntojen kuvaussyntaksi
SIP	Session Initiation Protocol, Internetin signaalointiprotokolla
SIP-KA	SIP-käyttäjäagentti
SMS	Short Message Service, tekstiviestipalvelu
SMTP	Simple Mail Transfer Protocol, sähköpostien välittämiseen käytettävä protokolla
SOAP	Simple Object Access Protocol, XML-pohjainen etäkutsuihin perustuva protokolla
TCP/IP	Transmission Control Protocol/Internet Protocol, internet, Internetin protokollapino
UDP	User Datagram Protocol, epäluotettava tiedonsiirtoprotokolla
UIML	User Interface Markup Language, käyttöliittymien kuvauskieli

UML	Unified Modeling Language, oliopohjainen mallinnustapa
UpnP	Universal Plug and Play, kotiverkotukseen käytettävä teknologia
URL	Uniform Resource Locator, Internet-resurssin osoite
USSD	Unstructured Supplementary Service Data, GSM-standardin istuntopohjainen tiedonsiirto-ominaisuus
VL	Verkotettu laite
VNC	Virtual Network Computing, käyttöliittymien bittikarttojen siirtoon käytettävä ohjelmisto
WML	Wireless Markup Language, Wireless Application Protocol -protokollan kuvauskieli
XML	Extensible Markup Language, kieli dokumenttien kuvaamiseen, jotka sisältävät rakenteisia tietotyypppejä

1. Johdanto

Kodin laitteita verkottamalla saadaan tuotettua uusia palveluita, joita käyttäjät voivat hyödyntää. Esimerkiksi videokamera tarjoaa käyttäjälleen videoleikkeitä, lämpömittari ja muunlaiset anturit mittausperäistä tietoa. Tämän lisäksi on olemassa palveluja, joiden avulla käyttäjä voi ohjata kodin laitteita, esimerkiksi asettaa saunan päälle. Näiden ja muiden palvelujen käyttäminen henkilökohtaisilla liikkuvilla laitteilla on tulevaisuudessa ajankohtaisempaa uusien laajakaistaisempien matkapuhelinverkkojen ja tehokkaampien päätelaitteiden kehittymisen ansiosta. Myös pyrkimys yhdistää kaikki laitteet Internetiin edistää asiaa.

Kodeissa on runsaasti erilaisia laitteita, jotka voidaan kytkeä kotiverkkoon. Nämä laitteet poikkeavat hyvin paljon toisistaan ja tarjoavat siten käyttäjilleen erilaisia käyttömahdollisuuksia. Kyseisten laitteiden kontrollointiin ja palvelujen hyväksikäyttöön kodin ulkopuolelta ei ole kehitetty ratkaisua, joka kattaisi kaikki kodin laitteet ja teknologiat. Tähän aihepiiriin liittyy kuitenkin useita uusia käyttömahdollisuuksia. Ihminen voisi esimerkiksi tarkistaa makuuhuoneen lampun tilan ja tarvittaessa kytkeä siitä virran pois. Ovikellon soidessa hän saisi ilmoituksen tapahtumasta ja voisi halutessaan päästää vieraan sisään käyttämällä langatonta päätelaitettaan oven aukaisemiseen. Näihin käyttötapauksiin liittyy kuitenkin lukuisia ongelmia. Esimerkiksi käyttäjät täytyy tunnistaa luotettavasti, jotta vain sallitut henkilöt voivat käyttää kodin laitteita. Palveluihin liittyvä tieto ei saa joutua ulkopuolisiin käsiin, joten kommunikointiin käytettävät viestit täytyy salata luotettavasti. Lisäksi laitteiden verkottamiseen käytetään lukuisia yhteensopimattomia teknologioita.

1.1 Tavoitteet ja rajaus

Tähän aiheeseen liittyy useita alueita, joita tutkitaan tällä hetkellä. Työn tavoitteena oli toteuttaa käytännössä kotona olevien yksinkertaisten laitteiden palveluiden hyödyntäminen kodin ulkopuolelta käyttäen langatonta päätelaitetta. Järjestelmän kehityksessä käytetään SIP- ja OSGi-teknologioita, joiden soveltuvuutta tutkitaan kyseiseen aihepiiriin painottaen SIP:n osuutta [1, 2]. Järjestelmän toimintaa demonstroidaan käyttötapauksilla. Kehitetylle järjestelmälle suoritettiin evaluointi, joka perustui järjestelmän vaatimuksiin, ratkaistaviin ongelmiin ja käyttötapauksista saatuihin kokemuksiin. Lisäksi työtä rajattiin siten, että kodin laitteiden kontrolloinnin turvallisuuteen, tunnistukseen ja luotettavuuteen ei otettu kantaa. Kontrolloivana päätelaitteena käytettiin vain yhtä langatonta terminaalialueita, jossa on avoin ohjelmointirajapinta.

Järjestelmän kehityksessä ja evaluoinnissa keskityttiin siis tiettyihin tavoitteisiin ja ongelmiin, joita esitellään seuraavassa esimerkkien avulla.

A. Nimeäminen ja osoitteistus

Kodin laitteiden nimeäminen on ratkaistava, jotta kyseisiin laitteisiin olisi mahdollista osoittaa yksikäsitteisesti. Osoitteistus liittyy tähän olennaisesti.

B. Liikkuvuus

Kodin laitteen kontrolloinnin täytyy olla mahdollista mistä tahansa, eli kontrolloivan laitteen täytyy olla liikkuva terminaali.

C. Käyttöliittymä

Kontrolloitavan laitteen käyttöliittymän tulee olla dynaamisesti ladattavissa kontrolloivaan laitteeseen.

D. Teknologiarippumattomuus

Kodin laitteiden verkottamiseen on mahdollista käyttää eri teknologioita. Kehitettävän järjestelmän tulee olla riippumaton näistä olemassa olevista teknologioista.

E. Kommunikointivaatimukset

Kodin laitteiden palveluita täytyy pystyä käyttämään kontrolloivan päätelaitteen avulla. Käyttäjän tulee olla mahdollista tiedustella, kontrolloida ja kirjautua näihin palveluihin liittyvään tietoon. Tiedustelu tarkoittaa esimerkiksi laitteen tilan kysymistä, kontrollointi tilan muuttamista ja kirjautuminen asynkronisen viestin saamista laitteen tilan muuttuessa. Käyttäjän täytyy myös pystyä hyödyntämään jotain muuta laitteen tuottamaa tietoa, kuten multimediaa.

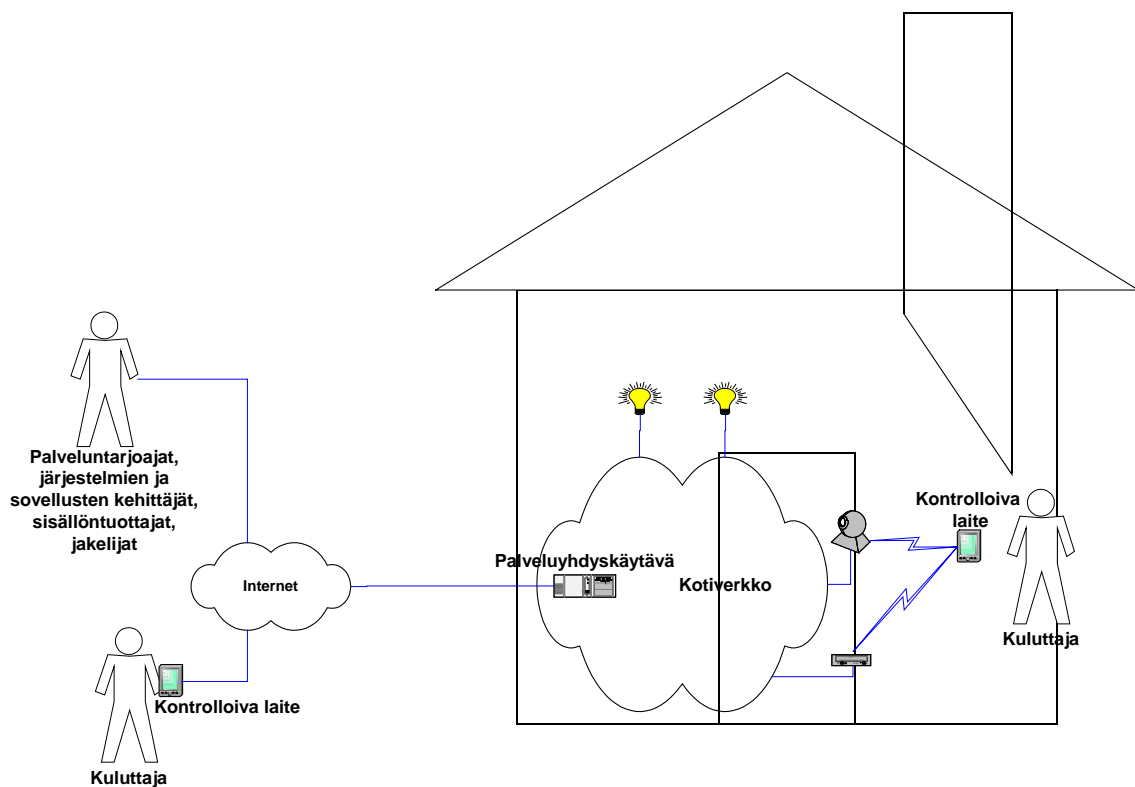
F. Skaalautuvuus

On olemassa paljon koteja, joissa on paljon laitteita, joten ratkaisun on oltava skaalautuva.

2. Kotiverkko ja verkotetut laitteet

Tässä luvussa käydään läpi nykyisiä kodin laitteiden verkotukseen käytettäviä teknologioita, aiheeseen liittyviä käsitteitä ja osapuolia sekä alan viimeisintä kehitystä. Kuvasta 1 nähdään, kuinka kodin paikallisverkkoon (residential home network) on liitetty lampuja, videokamera ja cd-soitin. Nämä laitteet tarjoavat palveluita kuluttajalle, joka käyttää kyseisiä palveluita joko kotona tai kodin ulkopuolella kontrolloivan laitteen avulla. Palveluyhdyskäytävä (residential gateway) toimii solmukohtana kodin ja ulkopuolisen ympäristön välillä.

Kodin laitteiden verkottamiseen ja palveluiden tarjoamiseen liittyy lukuisia osapuolia [3, s. 114–115]. Palveluntarjoajien täytyy ottaa huomioon kodin laitteiden yhdistämiseen käytetyt teknologiat. Myös jakelijoiden eli järjestelmäsuunnittelijoiden, myyjien ja asentajien täytyy tuntea kyseiset teknologiat. Nämä osapuolet päättävät lopulta, mitä teknologioita markkinoidaan ja myydään kuluttajille. Johtavat ja suurimmat laitevalmistajat osallistuvat kodin avoimien spesifikaatioiden kehittämiseen, jotta kotitekniikoiden integrointi helpottuu ja tulee halvemmaksi. Kotien suunnittelijoilla ja rakentajilla on suuri rooli yhdistettäessä uutta kotia ja sen kotiverkkoa. Tällöin älykkäät suunnitteluratkaisut helpottavat laitteiden käyttöä ja siten nostavat kodin arvoa. Videon, musiikin ja sovellusten jakaminen jokaiseen kodin laitteeseen koskee sisällöntuottajia ja omistajia. Järjestelmien ja sovellusten kehittäjillä on tärkeä rooli kehitettäessä käyttöjärjestelmiä, protokollia ja sovelluksia kodin laitteille ja palveluyhdyskäytävälle.



Kuva 1. Kodin laitteiden kontrollointi.

2.1 Kodin teknologiat

Seuraava kotiverkon määritelmä, jonka CEA:n HNIT-osasto (Home Networking and IT) on hyväksynyt, sopii hyvin tähän työhön: "Kotiverkko yhdistää elektronisia laitteita ja systeemejä mahdollistaakseen näiden tuotteiden, systeemien ja minkä tahansa sisälön, kuten musiikin, videon ja datan, kontrolloinnin ja etäkäsittelyn" [4]. Avainasioita kyseisessä määritelmässä ovat yhdistäminen, kontrollointi ja etäkäsittely. Tuotteet täytyy yhdistää toisiinsa, sisältöä on pystyttävä käsittelemään, ja käyttäjän on voitava kontrolloida laitteita ja niihin liittyvää sisältöä [5, s. 78].

Kodin laitteita yhdistetään useilla eri teknologioilla, kuten kuvasta 2 nähdään. Itse asiaa kotien verkottamiseen liittyy 50 standardia tai organisaatiota, joista vain osaa pidetään tärkeinä [5, s. 79]. Fyysisellä tasolla tiedonsiirtoon käytetään muun muassa sähköverkkoa, puhelinverkkoa, koaksiaalikaapelia ja RF-tekniikkaa. X.10 [6] on saantiteknologia (access technology), joka käyttää sähköverkkoa yksinkertaisten laitteiden väliseen kommunikointiin. Bluetooth [7] on lyhyen kantaman RF-radiolinkkejä hyväksi käytävä teknologia, joka toimii 2,4 GHz:n taajuusalueella. On olemassa siis langattomia ja kiinteitä ratkaisuja kodin laitteiden verkottamiseen [3, s. 116]. Langattomat teknologiat mahdollistavat laitteiden rajatun liikkuvuuden kotona, ja niiden avulla vältetään kodin langallisen infrastruktuurin asentaminen. Huonoja puolia ovat kuitenkin korkea hinta ja alhaisempi suorituskyky kiinteisiin vaihtoehtoihin verrattuna. Laitteiden verkottamiseen käytetään yleisesti IP-pohjaisia teknologioita. Tämän lisäksi tarvitaan protokollia laitteiden nimeämiseen, löytämiseen (discovery) ja niiden dynaamiseen verkkoon liittämiseen ja poistamiseen. Näitä kutsutaan sovelluserroksen teknologioiksi, joista tässä työssä käsitellään OSGi:tä [2] ja SIP:tä [1]. SIP on alun perin kehitetty verkoistuntojen luomiseen, mutta kyseistä protokollaa on mahdollista laajentaa kodin verkotettujen laitteiden kontrolloimiseksi. OSGi sen sijaan määrittelee palveluyhdyskäytävälle joukon ohjelmistorajapintoja, joiden avulla on mahdollista yhdistää kodin laitteet ulkopuolisen Internetin kanssa.

Sovelluserroksen teknologiat (OSGi, Jini, UPnP, HAVi, SIP)
Verkkoteknologiat (IP)
Saantiteknologiat (X.10, Bluetooth)
Siirtoteknologiat (sähköverkko, RF, koaksiaalikaapeli, puhelinverkko, kerrattu johdinsiipi, kuitupohjaiset ratkaisut)

Kuva 2. Kodin teknologiat.

Kotiverkko poikkeaa hieman toimistoverkosta. Siihen liitetään usein erilaisia laitteita kuin toimistoverkkoon. Esimerkiksi kotona halutaan kontrolloida viihde-elektroniikkaa, jota ei usein työpaikalla ole. Tämän lisäksi yritysten tietojen turvaamiseksi toimistoverkoissa käytetään vahvempaa suojausta kuin kodeissa, kuten esimerkiksi palomuuripalvelimia. Kotiverkot ovat myös usein pienempiä kuin yritysten lähiverkot. Lisäksi kodeista muodostetaan yhteys Internetiin yleensä modeemi- tai xDSL-yhteyksien avulla, kun taas suurten yritysten lähiverkot on usein liitetty kiinteästi Internetiin.

2.2 Palveluyhdyskäytävä

Kuten kuvasta 1 huomattiin, palveluyhdyskäytävä yhdistää ulkoisen ympäristön, kotiverkon ja sen laitteet. Se on tarkoitettu toimimaan standardina rajapintana, jonka avulla voidaan yhdistää kodin ja ulkoisen ympäristön useat verkot. Ulkopuolisen Internetin ja kodin verkotettujen laitteiden väliseen kommunikointiin liittyen sen voidaan ajatella suorittavan seuraavia toimintoja verkko- ja palvelutasolla [8, s. 3]:

- IP-osoitteen hankkiminen

Kodin laitteet tarvitsevat mahdollisesti IP-osoitteen kommunikoidakseen ulkopuolisen ympäristön kanssa.

- Tunnistaminen ja salaaminen

Ulkopuolelta tulevat viestit täytyy tunnistaa luotettavasti sekä salata, jotta niitä koskeva tieto ei leviä ulkopuolisille.

- Suodatus

Palveluyhdyskäytävä toimii palomuurina estäen ei-toivottujen tietopakettien saapumisen kotiin ja lähtemisen sen ulkopuolelle.

- Verkko-osoitteiden muuntaminen (NAT)

Laitteilla voi olla kotiverkolle kuuluvia verkko-osoitteita. Nämä eivät kuitenkaan välttämättä päde maailmanlaajuisessa Internetissä, joten osoite täytyy muuttaa, ennen kuin on mahdollista muodostaa esimerkiksi TCP-yhteys kotiverkon laitteen ja ulkopuolisen terminaalin välillä.

- Kotiautomaatiopalvelu [9]

Palveluyhdyskäytävää käytetään apuna ulkopuolisten osapuolien kehittämien kodin laitteiden kontrollointiin ja konfigurointiin liittyvien sovellusten vastaanottamiseen, asentamiseen ja hallitsemiseen.

Residential Gateway -työryhmä on esittänyt palveluyhdyskäytävälle fyysistä rakennetta [8, s. 4], joka koostuu yhteisestä väylästä, virtalähteestä, kotiverkotukseen ja palvelujen tarjoamiseen käytettävistä verkkorajapinnoista sekä prosessointituesta. Tämä johtuu siitä, että PC-alustan ei katsota riittävän palveluyhdyskäytävän tarpeisiin. Kuitenkin kodeissa käytettävät monet teknologiat vaikeuttavat tällaisen yhdyskäytävän suunnittelua.

Palveluyhdyskäytävissä sovelletaan usein väliohjelmistoja. Ne ovat "ajonaikaisen järjestelmän ohjelmistoja, jotka mahdollistavat sovellustason vuorovaikutuksen ohjelmien välillä" [10, s. 13]. Väliohjelmistot toimivat eräänlaisena "liimana" käyttöjärjestelmän ja sovellusten välillä tarjoten rajapinnan (API), joka mahdollistaa sovellusten suorittamisen useissa käyttöjärjestelmissä ja ajonaikaisissa ympäristöissä. OSGi-teknologia on esimerkki väliohjelmistoratkaisusta. Se käyttää Java-alustan ominaisuuksia määrittelyyn rajapinnat, jotka mahdollistavat kotiverkon ja ulkopuolisen ympäristön yhdistämisen palveluyhdyskäytävän avulla.

Kodin ja ulkoisen Internetin väliseen kommunikointiin käytetään tyypillisesti TCP/IP-protokollapinon perustuvaa ratkaisua. Tämän kommunikaation toteutukseen käytetään yleensä modeemi- ja xDSL-yhteyksiä.

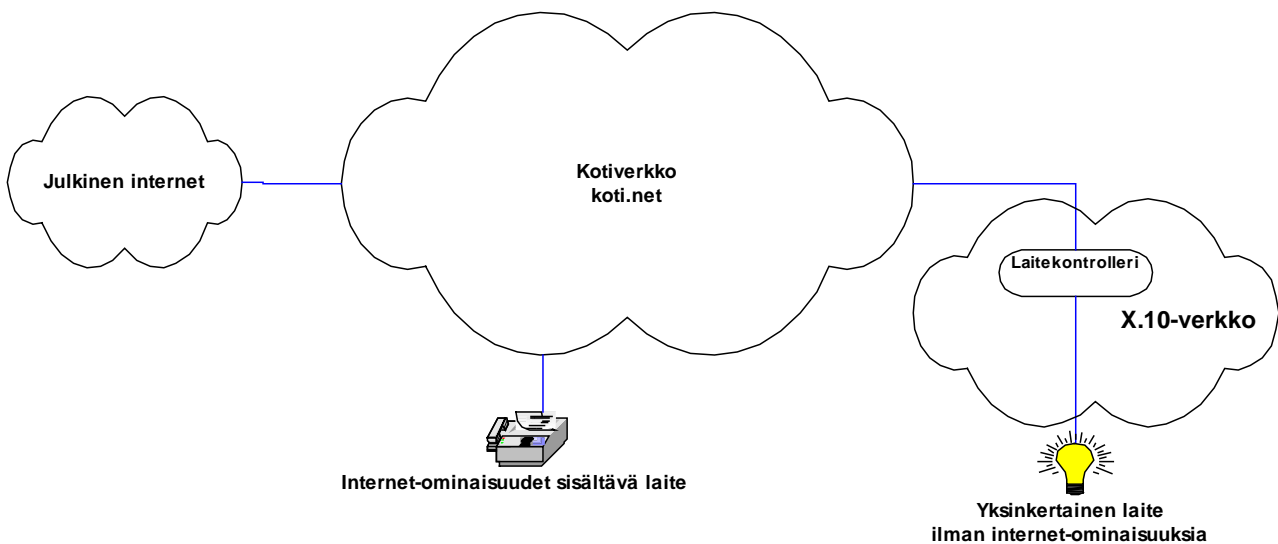
2.3 Verkotettu laite

Kotiverkon muodostamiseen käytettävät teknologiat riippuvat verkotettavien laitteiden ominaisuuksista. Yksinkertaisen laitteen, kuten lampun, ainut säädettävä muuttuja on sen virta, jolloin lamppu voidaan liittää sähköverkkoon ja X.10-teknologiaa käytetään sen kontrolloimiseen. CD-soitin on esimerkki prosessorin sisältävästä älykkästä laitteesta, jonka kontrolloimiseen ei voida käyttää pelkästään X.10-teknologiaa, mikäli halutaan säätää sen monia toimintoja, kuten esimerkiksi äänenvoimakkuutta. On myös olemassa Internet-ominaisuudet sisältäviä laitteita, jotka voidaan liittää suoraan kotiverkkoon, kuten TCP/IP-protokollapinon ominaisuuksilla varustettu herätyskello.

Erään määritelmän mukaan verkotettu laite on mikä tahansa kulutuselektroniikkaan liittyvä laite, jolla on ainakin yksi tietoverkkoon liitetty prosessori [11, s. 52–53]. Toisen lähteen mukaan [12, s. 47–48] määritelmän täyttävillä laitteilla on Internet-yhteys, ja ne voidaan jakaa kolmeen eri luokkaan riippuen siitä, mikä osapuoli määrittelee lait-

teella suoritettavat tehtävät. Ensimmäiseen luokkaan kuuluvien laitteiden tehtävät määrää valmistaja, toiseen palveluntarjoaja ja kolmanteen kuluttaja.

Edellä mainitut lähteet lähestyvät kyseessä olevaa määritelmää eri näkökulmista. Tässä työssä verkotetulla laitteella (VL) tarkoitetaan mitä tahansa laitetta, joka voidaan kytkeä kodin lähiverkkoon. Nämä laitteet jaetaan Internet-ominaisuudet sisältäviin laitteisiin, jotka voidaan kytkeä suoraan kotiverkkoon, ja yksinkertaisiin laitteisiin, joiden verkkoon liittäminen tapahtuu laitekontrollerin avulla. Kuva 3 selvittää asiaa, jossa lampun liittämisessä on käytetty apuna laitekontrolleria, jota printterin kytkemisessä ei tarvita. Kontrolloituva laite on mikä tahansa laite, jolla on mahdollista hyödyntää kodin verkotettujen laitteiden tarjoamia palveluita. Kotiverkko käsitetään yhdeksi DNS-alueeksi (koti.net).



Kuva 3. Kodin verkotetut laitteet.

2.4 Verkotetun laitteen tarjoamat palvelut

Kuten todettiin aikaisemmin, verkotetut laitteet tarjoaa kuluttajalle useita erilaisia palveluita. Nämä palvelut sisältävät erityyppistä laitteeseen tai sen ympäristöön liittyvää tietoa, joita eritellään esimerkkien avulla taulukossa 1.

Taulukko 1. Kodin verkotettuihin laitteisiin liittyviä palveluita.

Kodin verkotettuihin laitteisiin liittyviä palveluita	
Kodin laite	Palvelut
Jääkaappi	virta (kaikissa laitteissa), lämpötila
Televisio	kuva- ja ääni-informaatio
Videonauhuri	kuva- ja ääni-informaatio, ohjelmoitava nauhoitus
Stereojärjestelmä	ääni-informaatio
Uuni	lämpötila
Sauna	lämpötila
Oven lukko	lukitseminen ja avaaminen
Lämmitysjärjestelmä	tehon ja lämmityksen säätö
Videokamera	kuva- ja ääni-informaatio
Herätyskello	ajastus
Varashälytin	hälytys

Esimerkeistä huomataan, että palveluihin liittyy *tilatietoa*, jollaisena voidaan pitää esimerkiksi uunin lämpötilaa tai lampun virtaa. Laitteeseen tai sen palvelun tilatietoon liittyy myös *läsnäolotietoa*. Tällä tarkoitetaan esimerkiksi videonauhurin saatavuutta kommunikointiin ulkopuolisen käyttäjän kanssa. Voi nimittäin olla mahdollista, että laite on kytketty pois kotiverkosta tai siirretty toiseen kotiin. Näin ollen palveluihin liittyy myös *sijaintitietoa*. Taulukon 1 perusteella nähdään, että palvelut käsittelevät joko pieniä tai hyvin suuria määriä tietoa. Esimerkiksi oven lukon avaamiseen ja lukitsemiseen käytettävät viestit ovat suhteellisen pieniä kooltaan verrattuna videokameran tuottamiin suuriin videoleikkeisiin tai laajakaistaisiin mediavirtoihin. Tällaisten video- ja äänivirtojen sekä muiden suurten tietomäärien ajatellaan tässä työssä olevan luokittelematonta muuta tietoa.

Tässä työssä verkotettujen laitteiden tarjoamiksi palveluiksi ajatellaan edellä mainittujen tietotyyppien hyödyntäminen kontrolloivalla laitteella. Näiden tietojen käsittely aiheuttaa usein toimintoja kontrolloitavassa laitteessa. Tämän tiedon käsittelemiseksi pyritään toteuttamaan kohdassa 1.1 mainitut kommunikointivaatimukset.

2.5 Alan viimeisin kehitys

On olemassa lukuisia esimerkkejä kodin laitteiden palveluiden hyödyntämisestä. HAVi [13] on väliohjelmistoteknologia, jonka avulla on mahdollista verkottaa kodin laitteita käyttämällä standardin IEEE 1394 mukaista fyysisen tason väylää [14]. Lähteessä [15] kuvatun järjestelmän avulla on mahdollista kontrolloida kodin HAVi-laitteiden käyttöliittymiä kodin ulkopuolelta. Kyseinen toteutus hyödyntää VNC-tekniikkaa [16]. Tällöin verkotettujen laitteiden sovellukset ovat suorituksessa kodin VNC-palvelimella. Sovelluksen tuottaman käyttöliittymän bittikarttakuvat siirretään etänäyttöpäätteelle RFB-protokollan avulla. RFB mahdollistaa graafisten käyttöliittymien etäkontrollon. Kyseistä protokollaa käytetään myös ohjaamaan etänäyttöpäätteen käyttöliittymän tapahtumat takaisin VNC-palvelimelle. RFB on riippumaton käyttöjärjestelmästä, ikkunointijärjestelmästä, sovelluksista ja siirtoon käytettävästä tavasta, koska se perustuu näyttöpäätteen kehyspuskurin (framebuffer) etäkäyttöön [17, s. 682]. VNC-tekniikka mahdollistaa kodin HTTP-palvelimen kautta tapahtuvan HAVi-laitteiden, kuten television ja videokameran, ohjauksen käyttäen kontrolloivana laitteena NTT Docomo -operaattorin matkapuhelinta tai 3COM-laitevalmistajan PalmPilot PDA-laitetta. Järjestelmän avulla on mahdollista välittää vain bittikarttoja eikä esimerkiksi semanttista tietoa.

Artikkelissa [18] kuvatun järjestelmän avulla on mahdollista kommunikoida kodin HAVi-laitteiden sovellusten kanssa HAVin vaatiman IEEE 1394 -väylän ulkopuolelta Internet-ominaisuudet sisältävien laitteiden avulla. Tässä ratkaisussa ulkopuolisella käyttäjällä on oltava päätelaitteessaan HAVi-ominaisuudet sisältävä ohjelmistokomponentti, jonka päätelaitteeseen saamiseen on kolme keinoa. Ohjelmistokomponentti voi olla esiasennettu, se voidaan ladata dynaamisesti laitteeseen kodin yhdyskäytävältä, tai se voi olla asennettavissa valmisosana (plug-in). Itse- kontrolloivien laitteiden HAVi-sovellukset ovat ladattavissa dynaamisesti kodin yhdyskäytävältä. Kommunikointi ulkopuolisten ja kodin HAVi-sovellusten välillä on toteutettu muuntamalla HAVi-rajapinnan kutsut XML/SOAP-tyyppisiksi [19] dokumenteiksi. Tätä muunnosta varten tarvitaan kodin yhdyskäytävässä HAViML (HAVi Markup Language) -palvelu. Kommunikointi yhdyskäytävän ja kontrolloivan laitteen välillä tapahtuu käyttämällä HTTP:tä viestien siirtoon.

Lähteen [20] järjestelmän avulla yhdistetään ulkopuolinen Internet ja kodin Jini-laitteet [21]. Jini on Java-pohjainen teknologia, jonka avulla on mahdollista dynaamisesti liittää laitteita kotiverkkoon ja löytää verkkoon liitetyt laitteet. Järjestelmässä NTT Docomon -matkapuhelimella otetaan yhteys kodin EXWeb-palvelimeen HTTP:n avulla käyttäen matkapuhelimen verkkoselainta. Kyseinen palvelin muuttaa HTTP-kutsun Jini-viestiksi Jini-hakupalvelulle (look-up service), eli EXWeb-alusta yhdistää paikallisen kodin ja sen ulkopuolisen ympäristön. Jini-hakupalvelua käytetään kodin laitteiden rekisteröin-

tiin ja löytämiseen. Viestin perusteella käyttäjälle lähetetään vastausviestissä lista kotona olevista laitteista. Järjestelmän avulla on mahdollista kontrolloida kodin videonauhuria ja ohjata videokuva etäympäristössä sijaitsevan tietokoneen monitorille.

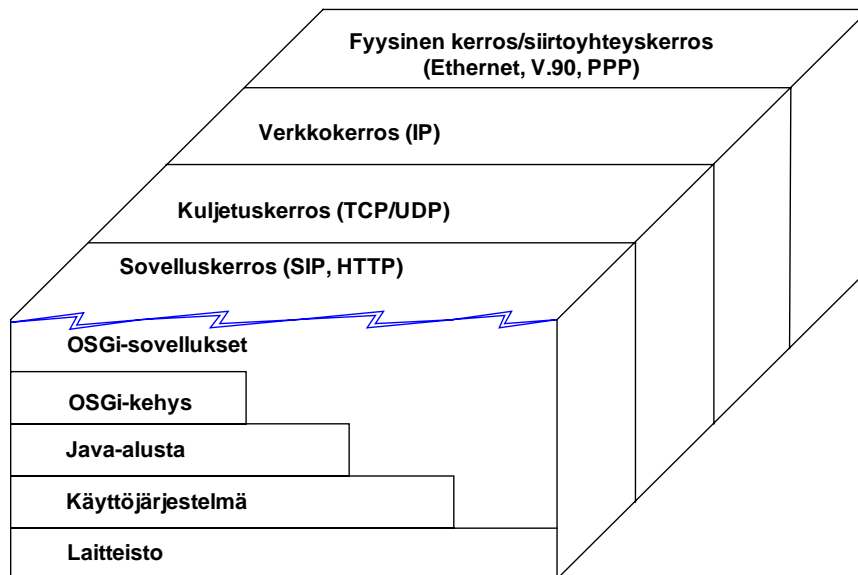
Soneran Talonmies-tuotteen [22] avulla on mahdollista kontrolloida yksinkertaisia laitteita, kuten lämmitystä ja valaistusta, SMS-viestien avulla. Tuote koostuu kotiin asennettavasta laitteistosta, johon voidaan liittää kontrolloitavia laitteita. Kyseisessä laitteistossa on GSM-liittymä, joka välittää komennot halutulle laitteelle SMS-viestin perusteella.

UIML [23] on geneerinen käyttöliittymien kuvauskieli. Se on XML-pohjainen ja mahdollistaa käyttöliittymien kuvaamisen vapaasti valituilla komponenteilla. Lähteessä [24] on kuvattu järjestelmä, jossa kodin laitteiden tarjoamia palveluita hyödynnetään geneeristen UIML-tyyppisten käyttöliittymäkuvausten avulla. Ratkaisussa ulkopuolinen käyttäjä ottaa yhteyden HTTP:n avulla kodin OSGi-palvelimelle, jossa HTML/VoiceXML-tyyppinen käyttöliittymä luodaan ajon aikaisesti kodin verkotetun laitteen UIML-kuvauksen perusteella ja lähetetään etäkäyttäjälle. VoiceXML [25] on puheohjaukseen käytettävä verkkopohjainen kuvauskieli. Kontrolloivassa laitteessa on HTML/VoiceXML-tiedostoja ymmärtävä selain, jonka avulla tulkitaan ladattu käyttöliittymä. Järjestelmän avulla on mahdollista kontrolloida kodin X.10- ja LON-laitteita sekä televisiota ja videonauhuria. Käyttäjän antamat komennot välitetään OSGi-palveluyhdyskäytävässä sijaitsevan HTTP-palvelimen avulla kohdelaitteelle.

Lähteessä [26] esitetyssä järjestelmässä kodin laitteet on liitetty HTTPS-palvelimeen turvallisuuden takaamiseksi. Kodin laitteita on mahdollista kontrolloida millä tahansa verkkoselaimen sisältävällä laitteella ja USSD-kanavaa käytävällä matkapuhelimella.

3. SIP- ja OSGi-teknologiat

Tässä diplomityössä tutkittiin SIP- ja OSGi-teknologioiden soveltuvuutta kodin verkotettujen laitteiden palveluiden hyödyntämiseen ja niiden yhteistoimintaa. Kuvassa 4 nähdään kyseisten teknologioiden välinen suhde. OSGi-kehys on väliohjelmistoa, joka toimii käyttöjärjestelmän ja sovelluskerroksen välissä. Se mahdollistaa OSGi-sovellusten suorittamisen Java-ympäristössä. Tämä on kuvattu pystyakselilla. SIP on sovelluskerroksen signaalintiprotokolla, jota käytetään muun muassa puheluiden muodostamiseen. Näin ollen SIP:n toiminnallisuus on mahdollista toteuttaa OSGi-sovelluksena. Näiden teknologioiden välinen rajapinta on merkitty katkoviivalla. SIP:n suhde muihin yleisimpiin Internet-protokolleihin selviää syvyysakselilta. Tässä luvussa kuvataan teoriaa, joka liittyy kyseisiin teknologioihin, ja kerrotaan SIP:n laajennuksista verkotettujen laitteiden palveluiden hyödyntämiseksi.



Kuva 4. SIP:n ja OSGin suhde.

3.1 SIP

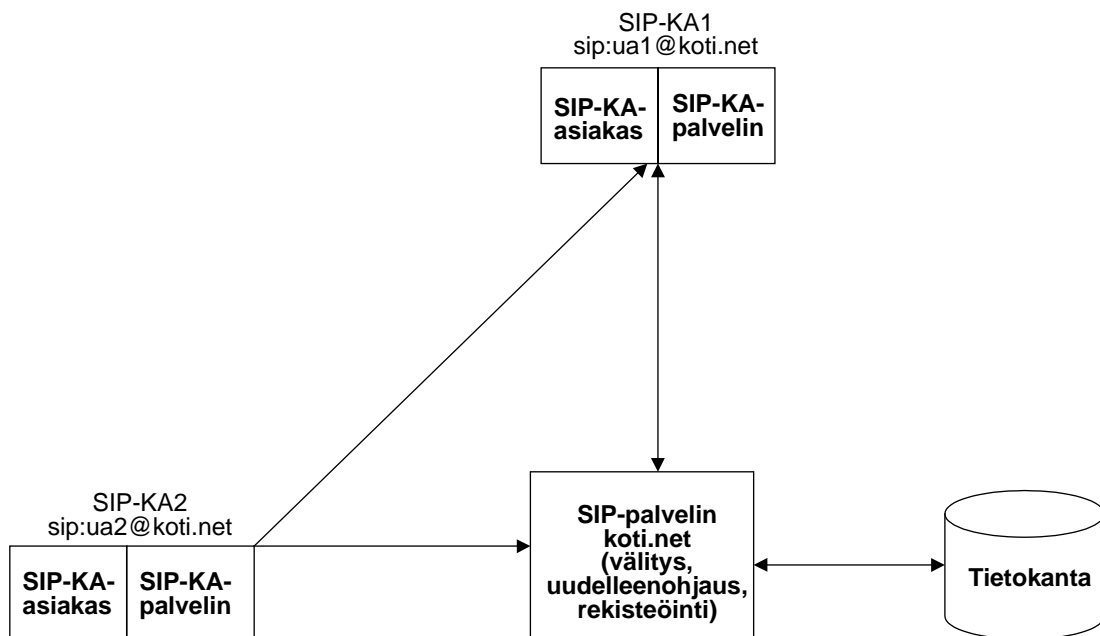
SIP-protokollaa on alun perin kehittänyt IETF:n Multiparty Multimedia Session Control (MMUSIC) -työryhmä. SIP standardoitiin vuonna 1999. 3GPP [27] tekee muunnosta SIP:sta tulevaisuuden matkapuhelinverkkoja varten. SIP yhdistää osia kahdesta laajasti käytetystä Internet-protokollasta, HTTP:sta ja SMTP:sta. SIP:n avulla voidaan muodostaa, muuttaa ja lopettaa verkkoistuntoja. Sen sovelluksia ovat tällä hetkellä Internet-puhelut ja multimediakonferenssit, mutta kyseistä protokollaa on mahdollista laajentaa myös muita käyttötarkoituksia varten, kuten välittömien sanomien lähettämiseen (instant messaging) [28].

SIP:n alkuperäinen käyttötarkoitus on siis luoda istuntoja käyttäjien välillä. Käyttäjät tunnistetaan sähköposti-tyyppisillä SIP-osoitteilla, ja mikä tahansa kokonaisuus, joka on osoitettavissa isäntäkoneen nimellä, voi osallistua SIP-istuntoon [29, s. 134]. SIP-viesteillä kommunikointiin voi liittyä neljä loogista kokonaisuutta, nimittäin käyttäjäagentti, välityspalvelin, rekisteröintipalvelin ja uudelleenvälityspalvelin.

3.1.1 SIP-käyttjäagentti

SIP-käyttjäagentit (SIP-KA) luovat käyttäjän aloitteesta mediaistuntoja muiden SIP-käyttjäagenttien kanssa. Yleensä käyttäjä on ihminen, toinen sovellus tai protokolla. Kommunikointi SIP-käyttjäagenttien välillä tapahtuu joko suoraan tai SIP-palvelimia hyväksi käyttäen. Kuvassa 5 on kaksi käyttjäagenttia (SIP-KA1 ja SIP-KA2). Niiden tunnistamiseen käytetään SIP-osoitteita (sip:ua1@koti.net ja sip:ua2@koti.net).

SIP-KA koostuu SIP-KA-asiakkaasta (SIP user agent client) ja SIP-KA-palvelimesta (SIP user agent server). SIP-KA-asiakas muodostaa ja lähettää pyyntöviestejä, ja SIP-KA-palvelin vastaa niihin vastausviesteillä. SIP-standardi suosittelee TCP- ja UDP-protokollien toteuttamista käyttäjäagentteihin viestien lähettämistä ja vastaanottamista varten, mutta SIP:n toiminta on riippumaton näistä kommunikointiprotokollista.



Kuva 5. SIP:n osapuolet.

3.1.2 SIP-palvelimet

On olemassa kolmenlaisia SIP-palvelimia: SIP-välityspalvelin (SIP proxy server), SIP-uudelleenohjauspalvelin (SIP redirect server) ja SIP-rekisteröintipalvelin (SIP registrar). Nämä kaikki ovat sovelluksia, jotka käsittelevät SIP-pyyntöviestejä ja välittävät niitä tai vastaavat niihin. On mahdollista, että yhtä palvelinta voidaan käyttää näinä kaikkina eri tyyppinä, kuten kuvasta 5 nähdään.

SIP-välityspalvelimet vastaanottavat ja välittävät SIP-pyyntöviestejä SIP-käyttäjä-agenttien puolesta. Ne käyttävät usein hyväkseen tietokantaa, joka auttaa pyyntöviestien prosessoinnissa. Näissä tietokannoissa on mitä tahansa tietoa SIP-käyttäjäagenttien sijainnista. Edellisessä kuvassa SIP-palvelimen (koti.net) tietokannassa on SIP-käyttäjä-agenttien sijainteihin liittyvää tietoa. SIP-käyttäjäagentit ilmoittavat sijaintinsa yleensä REGISTER-pyyntöviesteillä.

SIP-välityspalvelimet ovat joko tilattomia tai tilallisia. Tilaton palvelin ei tallenna muistiin prosessoimiaan SIP-viestejä eikä koskaan lähetä viestiä uudelleen. Tilallinen palvelin pitää kirjaa vastaanottamistaan ja lähettämistään viesteistä ja käyttää tätä tietoa hyväksi uusien SIP-viestien käsittelyssä.

Käyttäjällä voi olla useita laitteita, joista hänet voi tavoittaa. Tällöin tilallisella välityspalvelimella on tiedossa useita kyseiseen käyttäjään liittyvien SIP-käyttäjäagenttien sijainteja. Käyttäjälle tuleva SIP-viesti haaroitetaan tilallisesta välityspalvelimesta (haaroittavasta välityspalvelimesta) kaikkiin laitteisiin, ja kyseinen välityspalvelin huolehtii jokaisen käyttäjäagentin lähettämistä vastausviesteistä.

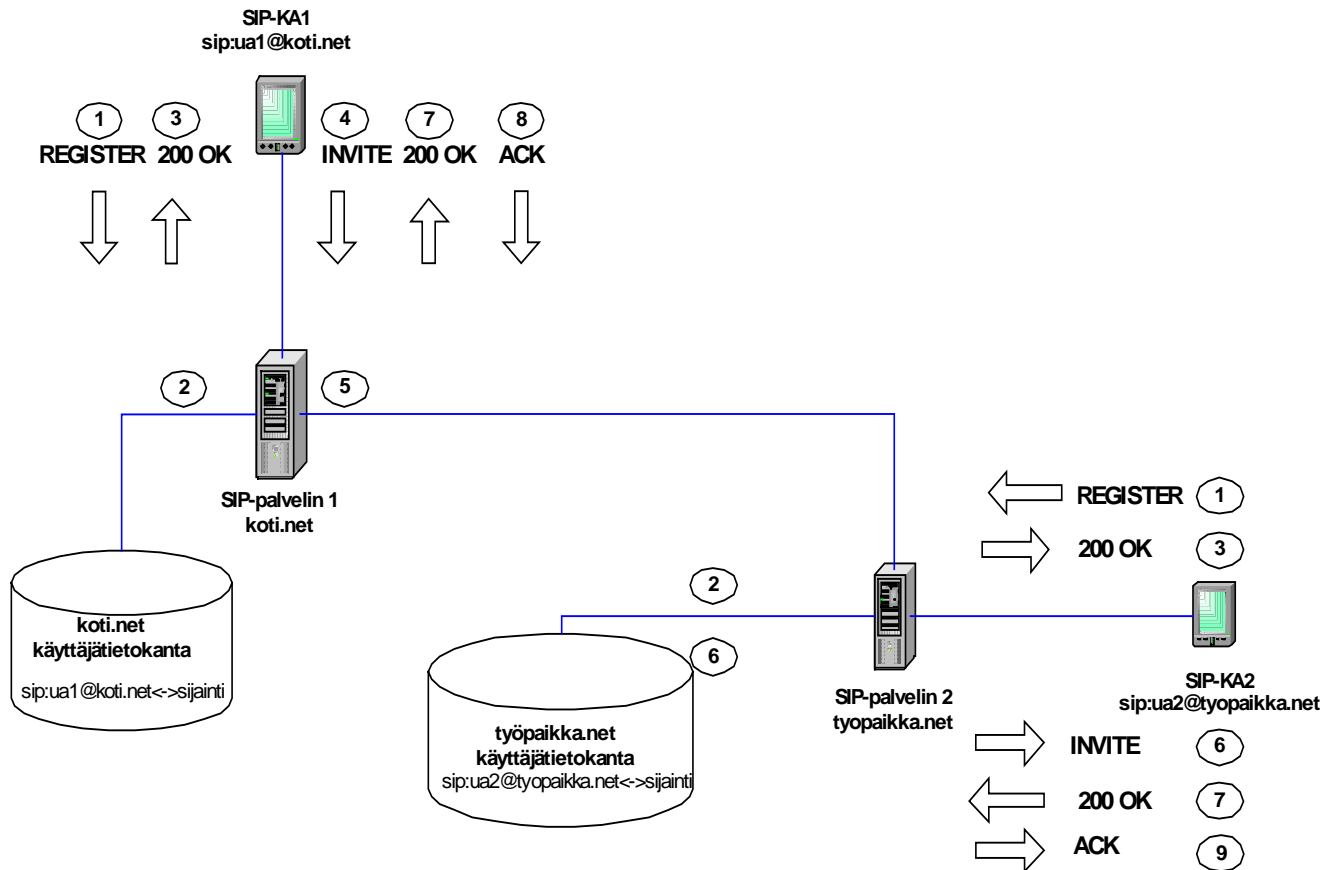
SIP-uudelleenohjauspalvelin hyödyntää tietokantoja tai sijaintipalvelua, kuten välityspalvelin, mutta se ei välitä viestejä kohti määränpäättä. Se sen sijaan palauttaa pyyntöviestin seurauksena uudelleenohjaus-tyyppisen vastausviestin, joka sisältää kohteen mahdollisen sijaintitiedon. Kohde voi olla toinen palvelin tai käyttäjäagentti. Tämän jälkeen SIP-KA voi lähettää uuden pyyntöviestin suoraan seuraavaan määränpäähän.

SIP-rekisteröintipalvelin hyväksyy vain REGISTER-pyyntöviestejä SIP-käyttäjä-agenttien sijaintitiedon lisäämiseksi, päivittämiseksi tai poistamiseksi. Kyseinen palvelin tietää oman alueensa käyttäjäagenttien sijainnin yleensä SIP-käyttäjäagenttien rekisteröintien perusteella. Esimerkiksi kuvassa 5 SIP-KA1 rekisteröi sijaintinsa kyseiselle SIP-palvelimelle, koska ne kuuluvat samaan alueeseen (koti.net). Rekisteröintipalvelin voi tarjota tätä kontaktitietoa muille saman alueen välitys- ja uudelleenohjauspalvelimille.

3.1.3 SIP-esimerkki

SIP:hen liittyvien osapuolten toiminta ja viestit selviävät parhaiten esimerkin avulla. Kuvan 6 esimerkissä SIP-käyttjäagenttien voidaan ajatella olevan puhelimia, PDA-laitteita tai tietokoneita, joissa on SIP-käyttjäagentin ominaisuudet. Verkkoistunnon luominen SIP-KA1:n ja SIP-KA2:n välillä tapahtuu seuraavasti:

- 1) SIP-KA1 ja SIP-KA2 ilmoittavat sijaintinsa heidän alueella oleville SIP-välityspalvelimille REGISTER-viestillä. Esimerkiksi SIP-KA1 lähettää viestinsä SIP-palvelimelle 1, koska ne kuuluvat samaan alueeseen (koti.net).
- 2) SIP-palvelimet tallentavat SIP-käyttjäagenttien sijainnit käyttäjätietokantoihin vastaanotetun REGISTER-viestin perusteella.
- 3) SIP-palvelimet lähettävät 200 OK -viestin vastauksena SIP-käyttjäagenteille onnistuneesta rekisteröinnistä.
- 4) SIP-KA1 haluaa muodostaa yhteyden SIP-KA2:n kanssa ja lähettää INVITE-viestin SIP-palvelimelle 1. Kyseisen viestin kohteeksi asetetaan SIP-KA2:n SIP-osoite (sip:ua2@tyopaikka.net).
- 5) SIP-palvelin1 huomaa, että viesti on tarkoitettu tyopaikka.net-alueen SIP-käyttjäagentille ja lähettää saamansa INVITE-viestin SIP-palvelimelle 2. Tässä oletetaan, että SIP-palvelin1 saa selville SIP-palvelimen 2 sijainnin DNS-haun perusteella.
- 6) SIP-palvelin2 hakee käyttäjätietokannastaan SIP-KA2:n sijainnin saamansa INVITE-viestin perusteella ja välittää kyseisen viestin edelleen SIP-KA2:lle.
- 7) SIP-KA2 lähettää 200 OK -vastausviestin SIP-palvelimelle 2 saamansa INVITE-viestin perusteella. SIP:hen kuuluvien ominaisuuksien avulla viesti kulkeutuu takaisin SIP-KA1:lle (tähän palataan SIP-otsikoiden yhteydessä).
- 8) SIP-KA1 lähettää ACK-pyyntöviestin SIP-KA2:lle, jolla se hyväksyy uuden yhteyden. Viesti lähetetään SIP-KA2:lle, kuten aikaisempi INVITE.
- 9) SIP-KA2 vastaanottaa ACK-viestin. Uusi verkkoistunto on luotu.



Kuva 6. SIP-esimerkki.

3.1.4 SIP-viestit

SIP poikkeaa muista signalointiprotokollista (ISUP, H.323), koska sen viestit ovat tekstimuotoisia ja niiden syntaksi hyvin samantapainen kuin HTTP:ssa. SIP-viestin rakenne on ilmaistu SIP-standardissa [1] käyttäen Backus Naur -notaatiota (kuva 7). SIP-viesti koostuu aloitusrivistä, yhdestä tai useammasta otsikosta sekä valinnaisesta hyötykuormasta. Pyyntö- ja vastausviestit eroavat toisistaan siten, että pyyntöviestit alkavat pyyntörivillä ja vastausviestit statusrivillä.

```

Geneerinen SIP-viesti = aloitusrivi
                        (*otsikko) CRLF
                        [hyötykuorma]
aloitusrivi = pyyntörivi | statusrivi
pyyntörivi = metodi SP pyyntö-URI SP SIP-
                        versio CRLF
statusrivi = SIP-versio SP statuskoodi SP syy
otsikko    = kentän nimi ":" [kentän arvo]CRLF

```

Kuva 7. SIP-viestin rakenne.

SIP-käyttäjäagentt lähettävät pyyntöviestejä tuottaakseen jonkin toiminnon SIP-palvelimella tai toisella SIP-KA:lla, kuten esimerkiksi rekisteröinnin REGISTER-viestiä käyttäen. Kuvassa 8 on esimerkki INVITE-pyyntöviestistä, jossa on kuusi otsikkoa (Via, From, To, Call-ID, CSeq ja Content-length). Pyyntörivin metodi kertoo viestin tyyppin. Eri tyyppisiä pyyntöviesteille ovat REGISTER, INVITE, ACK, CANCEL, BYE ja OPTIONS. Pyyntö-URI ilmaisee viestin kohdeosoitteen, joka esimerkissä on sip:ua2@domain2.fi.

```
INVITE sip:ua2@domain2.fi SIP/2.0
Via: SIP/2.0/TCP 123.123.123.123:1990
From: sip:ua1@domain1.fi
To: sip:ua2@domain2.fi
Call-ID: 3212a13p7b1fij112h1b@domain1.fi
CSeq: 1 INVITE
Content-length: 0
```

Kuva 8. INVITE-pyyntöviesti.

SIP-KA-palvelin ja SIP-palvelin luovat ja lähettävät vastausviestejä SIP-KA-asiakkaan lähettämiin pyyntöviesteihin. Nämä viestit antavat lähettäjälle tietoa pyyntöviestin onnistumisesta. Kuvassa 9 on esimerkki viestistä, joka on vastaus aikaisempaan INVITE-viestiin. Kyseisen vastausviestin statusrivin kolminumeroinen statuskoodi (200) kertoo lähettäjälle, että INVITE-pyyntö on vastaanotettu onnistuneesti. Vastauksen statusrivin syy-kenttä sisältää tekstimuotoista tietoa (OK), joka on tarkoitettu lähettävän SIP-KA:n ihmiskäyttäjälle. Vastausviestit jaetaan kuuteen eri luokkaan, joiden perusteella lähettäjä voi päätellä pyyntöviestin tuloksen. Statuskoodin ensimmäinen numero kertoo, mihin luokkaan vastaus kuuluu. Kuvan 9 vastausviesti kuuluu siten 2xx-tyyppisiin vastausviesteihin.

```
SIP/2.0 200 OK
Via: SIP/2.0/TCP 123.123.123.123:1990
From: sip:ua1@domain1.fi
To: sip:ua2@domain2.fi
Call-ID: 3212a13p7b1fij112h1b@domain1.fi
CSeq: 1 INVITE
Content-length: 0
```

Kuva 9. 200 OK -vastausviesti.

SIP-otsikko

SIP-viesteissä on useita otsikoita. Niiden keskinäisellä järjestyksellä viestissä ei ole väliä lukuun ottamatta Via-otsikoita. SIP-otsikon yleinen rakenne on kuvattu SIP-standardissa kuvan 7 mukaisesti. SIP-otsikot ovat päästä-päähän (end-to-end) tai solmusta-solmuun (hop-by-hop) -tyyppisiä. Useimmat otsikot ovat päästä-päähän-otsikoita, koska SIP:hen liittyy usein päästä-päähän-kontrolli. SIP-palvelimet voivat muuttaa ja lisätä vain solmusta-solmuun-tyyppisiä otsikoita. Kuvan 9 200 OK -vastausviestissä on kuusi päästä-päähän-tyyppistä otsikkoa, joista Via, From, To, Call-ID ja CSeq ovat pakollisia jokaisessa SIP-viestissä.

Via-otsikoista selviää pyyntöviestin reitti lähettäjältä vastaanottajalle. Jokainen pyyntöviestin varrella ollut SIP-palvelin lisää viestiin uuden Via-otsikon, jossa se ilmoittaa oman IP-osoitteen. Kun SIP-palvelimet toimivat päinvastoin vastauksen suhteen, vastausviesti kulkee saman reitin vastaanottajalta lähettäjälle kuin pyyntöviesti kulki lähettäjältä vastaanottajalle. Tämän Via-otsikon ominaisuuden avulla edellisen esimerkin 200 OK -vastaus oli mahdollista ohjata takaisin INVITE-viestin lähettäjälle. Kyseistä SIP:n piirrettä käytetään hyväksi myös estämään pyyntöviestien silmukan muodostuminen eli pyyntöviestin lähettäjälle takaisin kulkeutuminen. From-otsikko kertoo pyyntöviestin lähettäjän ja To-otsikko vastaanottajan osoitteen. Nämä otsikot ovat samanlaisia pyyntöviestiä vastaavassa vastausviestissä.

SIP-URL

SIP-protokollassa eri osapuolten (SIP-KA:n, SIP-palvelimen) osoitukseen käytetään SIP-URL-osoitteita. Tämän osoitteen syntaksissa on samoja piirteitä kuin telnet- ja mailto URL -osoitteissa. SIP-URL on muotoa sip:käyttäjä@isäntäkone. Kuvassa 10 on esimerkkejä SIP-URL-osoitteista. Käyttäjä on joko käyttäjänimi tai puhelinnumero. Isäntäkone ilmaisee alueen nimen tai fyysisen IP-osoitteen. Isäntäkoneosan perään voidaan lisätä haluttu määrä URL-parametreja (kuten esimerkin transport=tcp ja user=phone) eroteltuna puolipisteellä, jotka kertovat lisätietoa kyseisestä viestistä.

```
1) sip:henkilöA@koti.net;transport=tcp
2) sip:+1-212-555-3333@tyopaikka.net;user=phone
3) sip:henkilöB@123.123.123.123:5060
```

Kuva 10. Esimerkkejä SIP-URL-osoitteista.

SIP-URL-osoitetta käytetään viestin aloitusrivillä ja To-otsikossa ilmaisemaan vastaanottaja ja From-otsikossa lähettäjä. Contact-otsikossa kerrotaan lähettäjän kontaktitiedoista. Tässä tapauksessa SIP-URL kertoo tavan, jolla käyttäjään voidaan ottaa yhteyttä SIP:n avulla.

Transaktio ja puheluhaara

SIP-transaktioon kuuluvat lähettäjän pyyntöviesti ja ne palvelimen (SIP-palvelimen tai SIP-KA-palvelimen) vastausviestit, jotka on lähetetty vastauksena kyseiseen pyyntöön. Vastauksen To-, From-, Call-ID- ja CSeq-otsikoiden täytyy olla identtisiä verrattuna vastaavaan pyyntöviestiin, jotta vastausviestit voidaan tunnistaa kuuluvan tiettyyn transaktioon. Call-ID-otsikko on maailmanlaajuisesti ainutlaatuinen tunniste, jota käytetään osana transaktioiden ja puheluhaarojen (call-leg) tunnistamisessa. CSeq-otsikko koostuu metodikentästä ja järjestysnumerosta, jota kasvatetaan jokaisen uuden pyyntöviestin yhteydessä. Aikaisemmat esimerkit pyyntö- (INVITE) ja vastausviestistä (200 OK) kuuluivat samaan transaktioon.

Puheluhaaraan voi kuulua yksi tai useampi transaktio. Luotaessa olemassa olevaan puheluhaaraan uusi transaktio pyyntöviestin CSeq-otsikon järjestysnumeroa kasvatetaan. Esimerkiksi muutettaessa mediaistunnon parametreja lähetetään uusi INVITE-viesti, jonka To, From- ja Call-ID-otsikot ovat samat kuin olemassa olevassa puheluhaarassa, mutta CSeq-otsikon järjestysnumeroa on kasvatettu yhdellä.

Haaroittava välityspalvelin voi välittää pyyntöviestin usealle käyttäjäagentille, jotka tunnistetaan saman SIP-osoitteen avulla. Tällöin vastauksen To-otsikkoon lisätään tag-parametri, jonka avulla pyyntöviestin lähettäjä erottaa eri käyttäjäagenteilta tulleet vastaukset toisistaan. Tässä tilanteessa nähdään ero puhelun ja puheluhaaran välillä, sillä kumpikin vastaus kuuluu samaan puheluhaaraan, mutta ne voidaan luokitella eri puheleluiksi. Tag-parametria käytetään yleisenä mekanismina erottamaan yhden SIP-osoitteen avulla tunnistetut käyttäjäagentit toisistaan.

Hyötykuorma

SIP-viesti voi sisältää erityyppistä tietoa hyötykuormanaan. Content-length-otsikko kertoo hyötykuorman koon ja Content-type-otsikko kuorman tyyppin. Content-encoding-otsikko ilmaisee hyötykuorman mahdollisen kompressoititavan. SDP on yleisin SIP-viestien hyötykuorman tyyppi. Se on tekstimuotoinen syntaksi, jota käytetään mediaistuntojen kuvaamiseen. SIP:tä ei ole tarkoitettu käytettäväksi suurien tietomäärien siirtoon. Yksi syy tälle monista on, että SIP-viesteillä kommunikointiin käytetään yleisesti epäluotettavaa UDP-protokollaa [29, s. 140]. Tällöin suurien viestien lähettäminen voi johtaa pirstoutumiseen verkkokerroksella (IP-kerros).

3.2 SIP-laajennukset verkotetuille laitteille

SIP:tä täytyy laajentaa, jotta sitä voidaan käyttää verkotettujen laitteiden palveluiden hyödyntämiseen [11]. Tarvitaan uusia pyyntöviestejä (DO, SUBSCRIBE, NOTIFY), joiden hyötykuormana käytetään DMP-tyyppisiä dokumentteja. Tässä kohdassa esitetään kyseiset laajennukset.

3.2.1 DMP

Device Messaging Protocol (DMP) [30, 11, s. 55] kuvaa XML-tyyppisten dokumenttien rakenteen, jonka perusteella luotuja viestejä käytetään verkotettujen laitteiden kanssa kommunikointiin. DMP määrittelee kyseiseen kommunikointiin liittyen minimaalisen joukon toimintoja ja rakenteen näiden toimintojen välittämiseksi eikä ota kantaa erilaisten laitteiden ominaisuuksiin. DMP-viestit koostuvat pyynnöistä ja vastauksista, kuten kuvasta 11 nähdään. Nämä dokumentit koostuvat laitetunnuksesta ja laitteen kontrollointiin, tiedusteluun ja kirjautumiseen liittyvästä tiedosta. Näiden viestien täydellinen rakenne on kuvattu lähteessä [30, s. 4–8].

```
<DMPAction>
  <sequence maxOccurs='unbounded'>
    <element ref='Device'>
      <choise maxOccurs='unbounded'>
        <element ref='t:Control' />
        <element ref='t:Query' />
        <element ref='t:Subscribe' />
      </choise>
    </sequence>
  </DMPAction>

<DMPResponse>
  <sequence maxOccurs='unbounded'>
    <element ref='Device'>
      <choise maxOccurs='unbounded'>
        <element ref='t:ControlResponse' />
        <element ref='t:QueryResponse' />
        <element ref='t:Notify' />
      </choise>
    </sequence>
  </DMPResponse>
```

Kuva 11. DMP-dokumenttien perusrakenne.

DMP-viestit mahdollistavat kommunikointiin liittyen seuraavat toiminnot:

Kontrollointi:

1. Laitteen yhden tai useamman toiminnon kontrollointi peräkkäin tai rinnakkain.
2. Tiettyyn joukkoon kuuluvien laitteiden kontrollointi, esimerkiksi mahdollisuus kytkeä kaikkiin kodin lampuihin virta.
3. Usean laitteen kontrollointi yhden toiminnon tuottamiseksi, esimerkiksi ovi täytyy lukita ja hälytysjärjestelmä asettaa päälle kodin turvallisuuden takaamiseksi.

Tiedustelu:

1. Laitteen yhden tai useamman muuttujan tilatiedon tiedusteleminen, esimerkiksi on mahdollista tiedustella jääkaapin lämpötilaa ja virrankulutusta.
2. Tiettyyn joukkoon kuuluvien laitteiden tilan tiedusteleminen.

Kirjautuminen:

1. Laitteen tapahtumalle kirjautuminen ja kirjautumisen keston asettaminen. On mahdollista saada asynkroninen ilmoitus, kun huoneen lämpötila nousee tietyn rajan yli.
2. Joukko tapahtumia voi muodostaa yhden kirjautumisen. On esimerkiksi mahdollista saada ilmoitus, kun oven lukko on avattu ja eteisen lamppu sytytetty.
3. Ilmoitus kertoo kirjautujalle tapahtumasta, esimerkiksi mikä on uusi huoneen lämpötila.

Kontrolloivalla laitteella on oltava tieto kontrolloitavasta verkotetusta laitteesta ja sen palveluista, jotta DMP:tä voitaisiin käyttää hyväksi. DMP ei ota kantaa siihen, kuinka tämä tietämys saadaan kontrolloivalle laitteelle. Tässä työssä DMP-viestejä käytetään DO-viesteissä [31] kontrollointiin ja tiedusteluun sekä SUBSCRIBE- ja NOTIFY-viesteissä [32] kirjautumiseen ja tapahtumista ilmoittamiseen. DMP-dokumentti asetetaan kyseisten SIP-viestien hyötykuormaksi ja Content-type-otsikko kertoo tällöin sisällön tyyppin (application/dmp). DMP on riippumaton sen siirtoon käytettävästä tavasta, joten esimerkiksi HTTP:tä voitaisiin käyttää SIP:n sijasta.

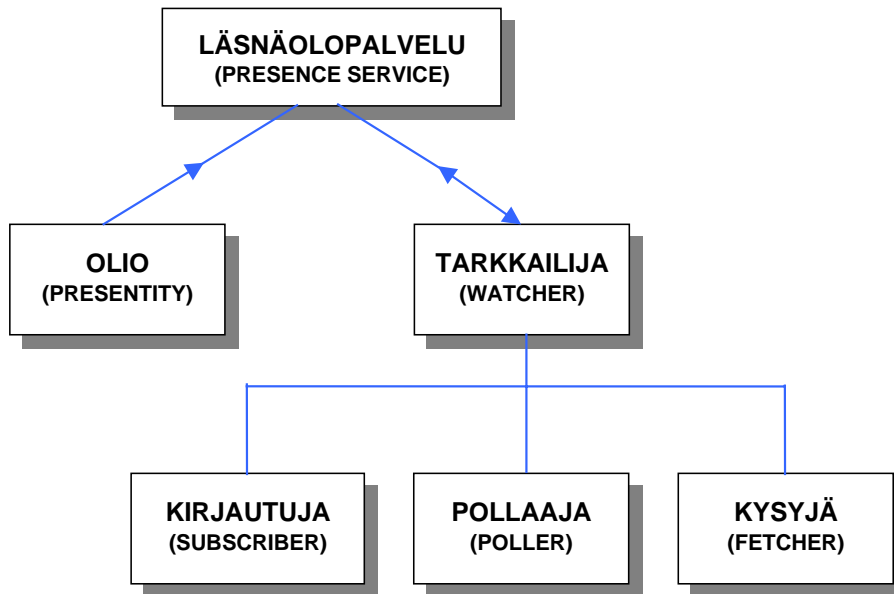
3.2.2 DO

DO on SIP:n laajennusviesti, jota käytetään verkotettuun laitteeseen liittyvään kontrollointiin ja tiedusteluun. Sen hyötykuormana käytetään DMP-tyyppistä pyyntöviestiä (Control), joka sisältää informaatiota verkotetusta laitteesta. Tämän pyynnön vastauksen (200 OK) hyötykuormassa palautetaan tieto pyynnön seurauksista DMP-tyyppisenä vastauksena (ControlResponse). Kyseiset pyyntö- ja vastausviestit (DO ja 200 OK) muodostavat transaktion, joka ei vaadi puheluhaaran luomista tai olemassaoloa. Tämä tarkoittaa sitä, että DO-viestit voivat käyttää olemassa olevaa puheluhaaraa tai luoda pelkästään uuden transaktion. Välityspalvelin käsittelee DO-viestin, kuten minkä tahansa pyyntöviestin, ja haaroittava palvelin, kuten ei-INVITE pyyntöviestin. Uudelleenohjauspalvelinta koskevat säännöt ovat samat kuin INVITE-viestille [31, s. 6].

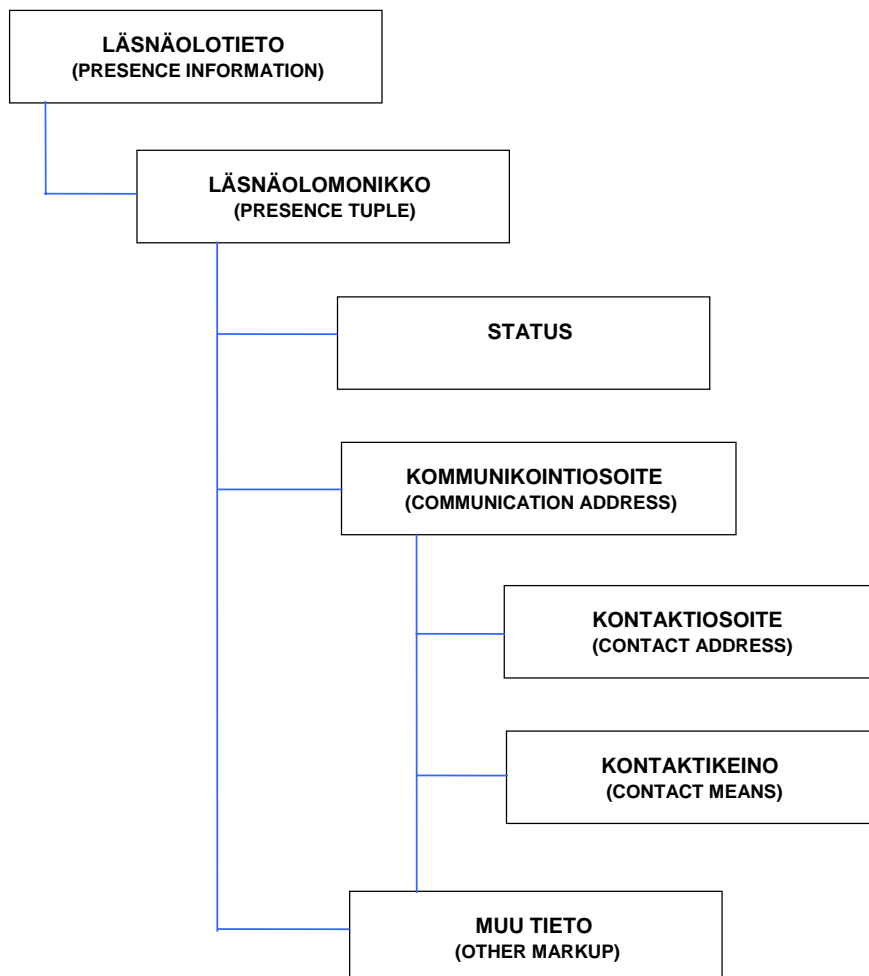
3.2.3 Läsnaolotieto

Kohdassa 2.4 kerrottiin, että verkotettujen laitteiden palveluihin liittyy läsnäolotietoa. Tämän tiedon käsittely on mahdollista toteuttaa SIP-protokollan laajennuksilla. Läheteessä [33] on määritelty malli, jossa käsitellään läsnäolotietoon liittyviä osapuolia, käsitteitä ja ominaisuuksia. Sen tarkoituksena on mahdollistaa keinot erilaisten läsnäolo- ja viestipalvelujen (presence and instant messaging) vertailemiseen ja kuvaamiseen. Tämä malli määrittelee kaksi palvelua: läsnäolopalvelun (PRESENCE SERVICE) ja viestipalvelun (INSTANT MESSAGE SERVICE). Läsnaolopalveluun liittyy kaksi osapuolta: olio (PRESENTITY) ja tarkkailija (WATCHER), kuten kuvasta 12 nähdään [33, s. 2–3]. Olio on mikä tahansa asia, joka tuottaa läsnäolotietoa läsnäolopalvelulle. Läsnaolopalvelu hyväksyy, tallentaa ja jakaa läsnäolotietoa. Tarkkailija pyytää tiettyihin olioihin liittyvää läsnäolotietoa. Tarkkailijat voidaan jakaa kirjautujiin (SUBSCRIBER), pollaajiin (POLLER) ja kysyjiin (FETCHER). Kirjautuja haluaa ilmoituksen yhteen tai useampaan olioon liittyvän läsnäolotiedon muuttuessa. Kysyjä tiedustelee läsnäolotietoa epäsäännöllisesti ja pollaaja säännöllisesti. Läsnaoloprotokolla määrittelee läsnäolopalvelun, olioiden ja tarkkailijoiden vuorovaikutuksen.

Läsnaolotiedon rakenne läsnäolopalvelussa käy ilmi kuvasta 13. Läsnaolomonikot sisältävät statuksen, joka ilmaisee olion saatavuuden kommunikointia varten valinnaisen kommunikointiosoitteen kautta. Läsnaolotiedon välittämiseen käytetään viestipalvelua. Muu tieto -kenttä sisältää ylimääräistä tietoa, joka liittyy olion läsnäolotietoon.



Kuva 12. Läsnäolopalveluun liittyvät osapuolet.



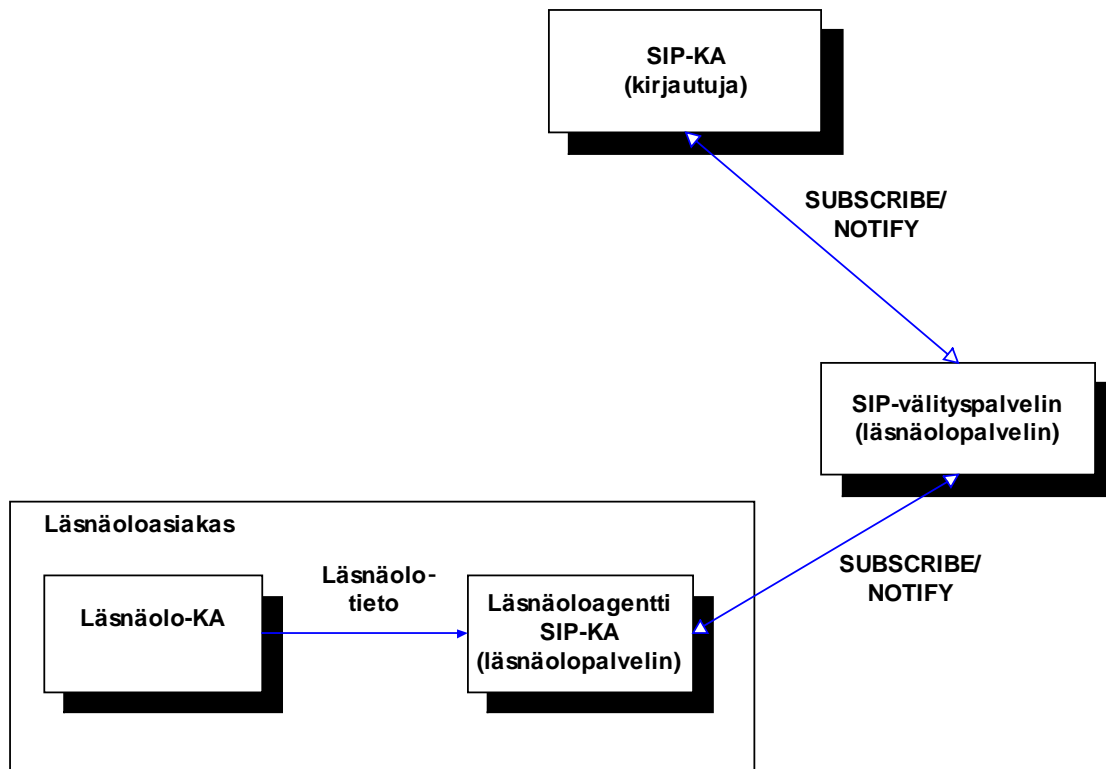
Kuva 13. Läsnäolotiedon rakenne.

SUBSCRIBE ja NOTIFY

SUBSCRIBE- ja NOTIFY-pyyntöviestejä käytetään kirjautumiseen ja asynkronisista tapahtumista ilmoittamiseen. Tässä työssä SUBSCRIBE-pyyntöä hyötykuormana käytetään DMP-tyyppistä pyyntöviestiä (Subscribe), joka kertoo verkotettuun laitteeseen liittyvästä kirjautumisesta. Tämä pyyntö kuitataan 2xx-luokan vastausviestillä. Kyseiseen kirjautumiseen liittyvistä tapahtumista ilmoitetaan NOTIFY-viesteillä, joiden hyötykuormana käytetään DMP-tyyppistä vastausta (Notify). NOTIFY kuitataan 2xx-luokan vastauksella.

SIP-laajennukset läsnäolotietoja varten [34] saadaan yhdistämällä SUBSCRIBE- ja NOTIFY-laajennusviestit läsnäoloon liittyvien käsitteiden kanssa eli käyttämällä SIP:aa läsnäoloprotokollana. Kuvassa 14 nähdään näihin laajennuksiin liittyvät osapuolet. Läsnäolo-KA:t voidaan ajatella olioiksi, jotka ilmoittavat läsnäolostaan läsnäoloagenteille. Siten läsnäoloagentti on läsnäolopalvelu, joka on tietoinen eri läsnäolokäyttjäagentteihin liittyvästä läsnäolotiedosta. Se vastaanottaa SUBSCRIBE-pyyntöjä ja lähettää NOTIFY-ilmoituksia liittyen tähän tietoon, joten sillä on SIP-KA:n ominaisuudet ja se on osoitettavissa SIP-URL-osoitteella. Läsnäoloasiakas on läsnäoloagentti, joka sijaitsee fyysisesti samassa paikassa läsnäolo-KA:n kanssa. Läsnäolopalvelin voi toimia joko läsnäoloagenttina tai SIP-välityspalvelimena. Välityspalvelimena toimiesseen se ohjaa SUBSCRIBE-viestit eteenpäin oikealle läsnäoloagentille ja läsnäoloagenttina se luonnollisesti prosessoi nämä kirjautumiset.

Tämä malli ei ota kantaa siihen, mitä eri osapuolet ovat käytännössä. Läsnäolotieto voidaan esimerkiksi liittää ihmiseen, jolloin eri läsnäolo-KA:t kuvaavat ihmisen eri laitteita, joista hänet voi tavoittaa, kuten kännykästä, lankapuhelimesta tai tietokoneesta. Kukin näistä laitteista tuottaa ihmisen läsnäolotietoa, joka ilmenee läsnäolomonikkona. Täten ihmiseen liittyvä läsnäolo koostuu näiden kaikkien läsnäolokäyttjäagenttien tuottamista läsnäolomonikoista, joista muodostuvaa läsnäolodokumenttia on mahdollista tiedustella kyseisen henkilön läsnäoloagentilta.



Kuva 14. Läsnäolotietoon liittyvien SIP-laajennusten osapuolet.

3.3 OSGi

OSGi on yhteisö, jonka tarkoituksena on luoda palveluyhdyskäytävälle avoimia spesifikaatioita, joiden avulla on mahdollista sijoittaa palveluita paikallisverkkoon ja sen laitteisiin suuralueverkkoa käyttäen [35, s. 110]. Palveluyhdyskäytävä toimii solmukohtana kotiverkon ja suuralueverkon välillä. Siihen voidaan yhdistää useat paikallisverkot ja niiden laitteet käyttämällä joko kiinteitä tai langattomia teknologioita. OSGi-palveluyhdyskäytävä on Java-pohjainen palvelujen suoritusalue, jonka toiminnasta kyseisen alustan operaattori on vastuussa. Palveluntarjoaja voi tuottaa lisäarvopalveluita kotiverkon laitteille toimittamalla sovelluksia yhdyskäytävän operaattorille. Tämä arkkitehtuuri mahdollistaa esimerkiksi kotiympäristön sovellusten etäohjelmoinnin ja konfiguraation [9]. OSGi:a voidaan hyödyntää myös muissa paikallisissa ympäristöissä, kuten esimerkiksi autoissa. OSGi-yhteisö on perustettu vuonna 1999, ja tällä hetkellä siihen kuuluu yli 80 jäsenyhtiötä, jotka osallistuvat spesifikaation kehittämiseen. Kyseisen vaatimusmäärittelyn pohjalta on tehty referenssitoteutuksia myös julkiseen levitykseen.

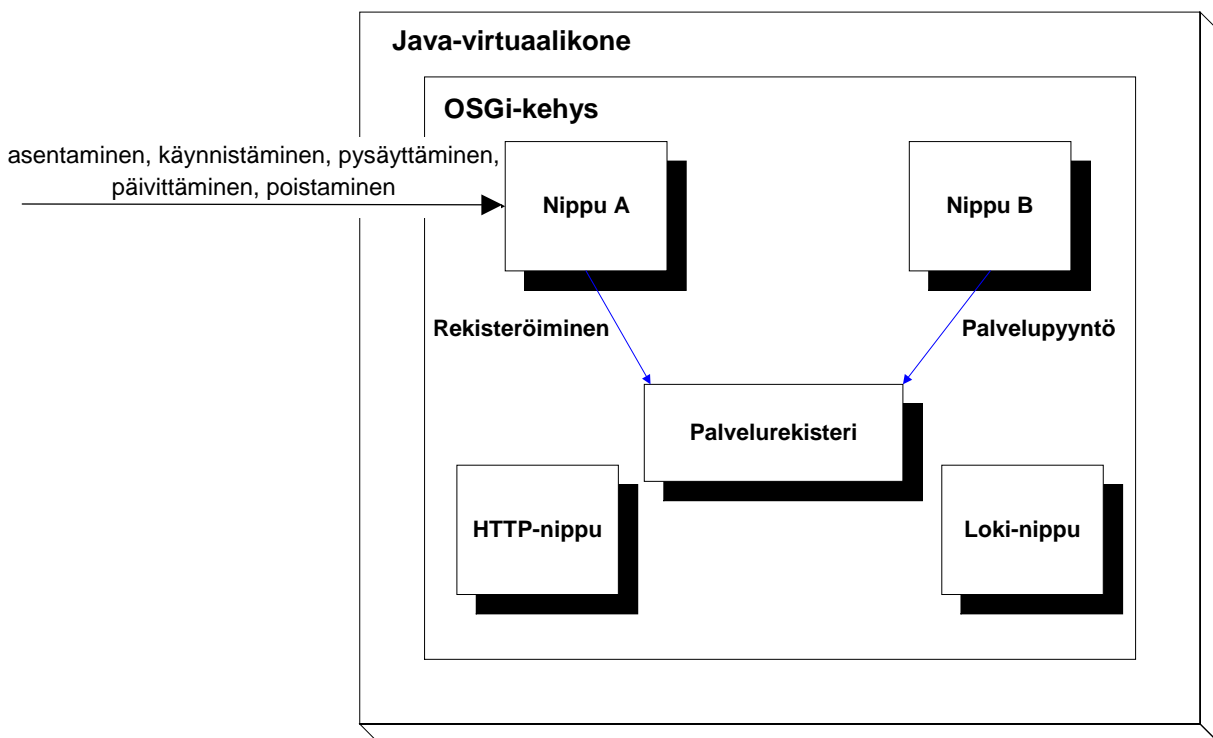
3.3.1 Kehys

OSGi-spesifikaation ydin sisältyy palveluyhdyskäytävään sijoittuvaan kehykseen (framework), joka voidaan luokitella väliohjelmistoksi. Se koostuu joukosta rajapintoja ja sijaitsee Java-virtuaalikoneen yläpuolella ajonaikaisessa ympäristössä, kuten todettiin

aikaisemmin kuvassa 4. Tämän OSGi-alustan avulla voidaan dynaamisesti asentaa ja hallita palvelusovelluksia, joita kutsutaan nipuiksi (bundle).

Jokainen OSGi-yhteensopiva laite voi asentaa, poistaa ja päivittää alustan nippuja. Nämä sovellukset voivat rekisteröidä palveluita ja Java-paketteja (package) kehykselle, joita muut niput voivat hyödyntää. Kehys kontrolloi nippujen asentamista, poistamista ja päivittämistä dynaamisella ja skaalautuvalla tavalla sekä hallitsee nippujen ja niiden palveluiden välisiä riippuvuuksia. Kehys määrittelee nippujen ohjelmoijille mallin, joka erottaa palvelurajapinnan sen toteutuksesta. Näin voidaan kehittää sovelluksia, joiden palvelut sidotaan kehukseen niiden rajapinnan perusteella ja toteutuksen valinta voidaan tehdä ajonaikaisesti. Tämä tekee kehyksestä skaalautuvan, mistä on hyötyä, kun alusta on suorituksessa erityyppisissä laitteistoissa. Kehys hallitsee palveluita palvelurekisterin avulla, jonka avulla niput pystyvät palvelujen rekisteröimisen ja hyödyntämisen lisäksi saamaan ilmoituksen palvelun tilan muuttuessa. Tämä tarkoittaa sitä, että nippuja on mahdollista päivittää ja muuttaa ilman OSGi-alustan uudelleenkäynnistystä.

Kuvassa 15 nippu A rekisteröi palvelurajapinnan toteutuksen palvelurekisteriin, jolta nippu B pyytää kyseisen palvelun käyttöönsä. Palvelu välitetään nipulle B palveluviit-
tausoliona. Tämä olio kapseloi palveluun liittyvät ominaisuudet, ja itse palveluolio saadaan OSGi-kehykseltä sen avulla.

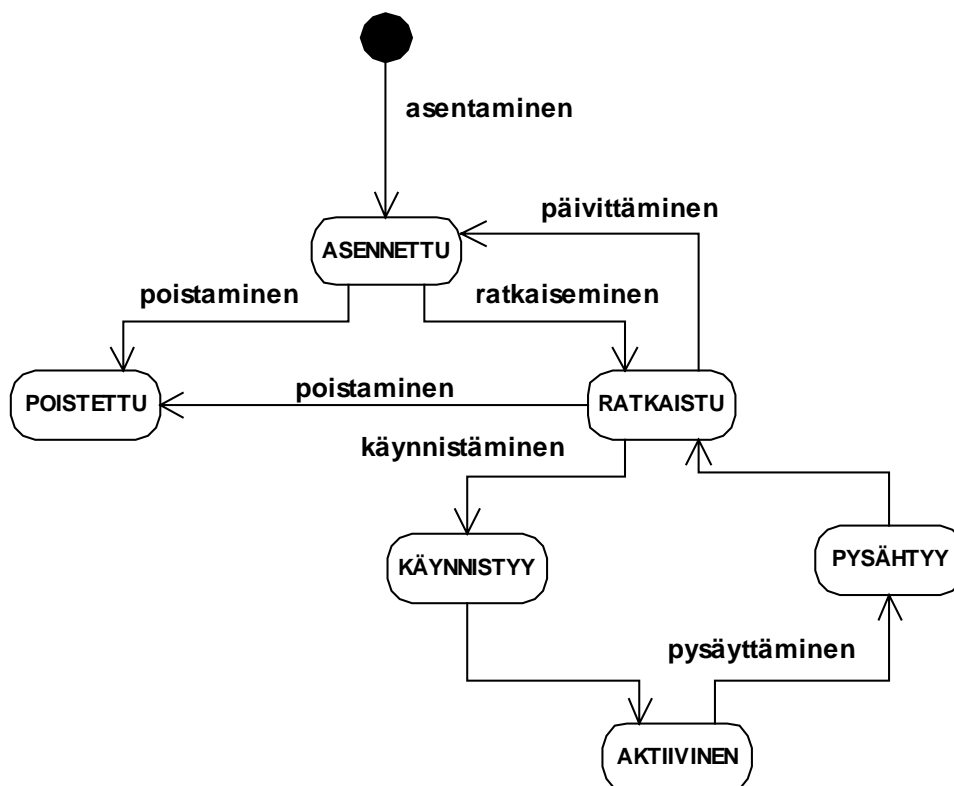


Kuva 15. OSGi-kehys, palvelurekisteri ja niput.

3.3.2 Nippu

Nippu on kokonaisuus, joka sisältää OSGi-alustalla suoritettavat sovellukset. Se on JAR-tiedosto koostuen Java-luokkatiedostoista sekä mahdollisesti muista resursseista, kuten kuvatiedostoista, joita käytetään palveluiden tarjoamiseen. Nippu sisältää manifest-tiedoston, joka kertoo kehykselle nipun ominaisuudet. Kehys tarvitsee tätä tiedostoa hallitakseen nipun elinkaarta sekä siihen liittyviä riippuvuuksia.

Kuvasta 16 nähdään nipun elinkaareen liittyvät eri tilat [2, s. 25]. Ennen kuin nippu voidaan käynnistää, kehyksen täytyy kyetä ratkaisemaan kyseisen nipun riippuvuudet muihin resursseihin, kuten esimerkiksi Java-paketteihin. Näistä riippuvuuksista kerrotaan nipun manifest-tiedostossa, jossa ilmaistaan myös nipun käynnistämiseen ja pysäyttämiseen tarvittavan luokkatiedoston sijainti nipun sisällä. Nipun käynnistämistä käytetään yleensä palveluiden rekisteröimiseen ja pysäyttämistä kyseisten palveluiden ja muiden resurssien vapauttamiseen. Nippu voidaan päivittää uudella versiolla kuitenkin siten, että alustalla on vain yksi versio kyseisestä nipusta. Uuden nipun täytyy olla yhteensopiva vanhan nipun kanssa, mikä tarkoittaa sitä, että sen täytyy toteuttaa vähintään sama palvelurajapinta ja tarjota samat Java-paketit kuin edellinen nippu. Nippua poistettaessa muita nippuja on mahdollista informoida kyseisestä tapahtumasta.



Kuva 16. Nipun elinkaari.

3.3.3 Ydinpalvelut

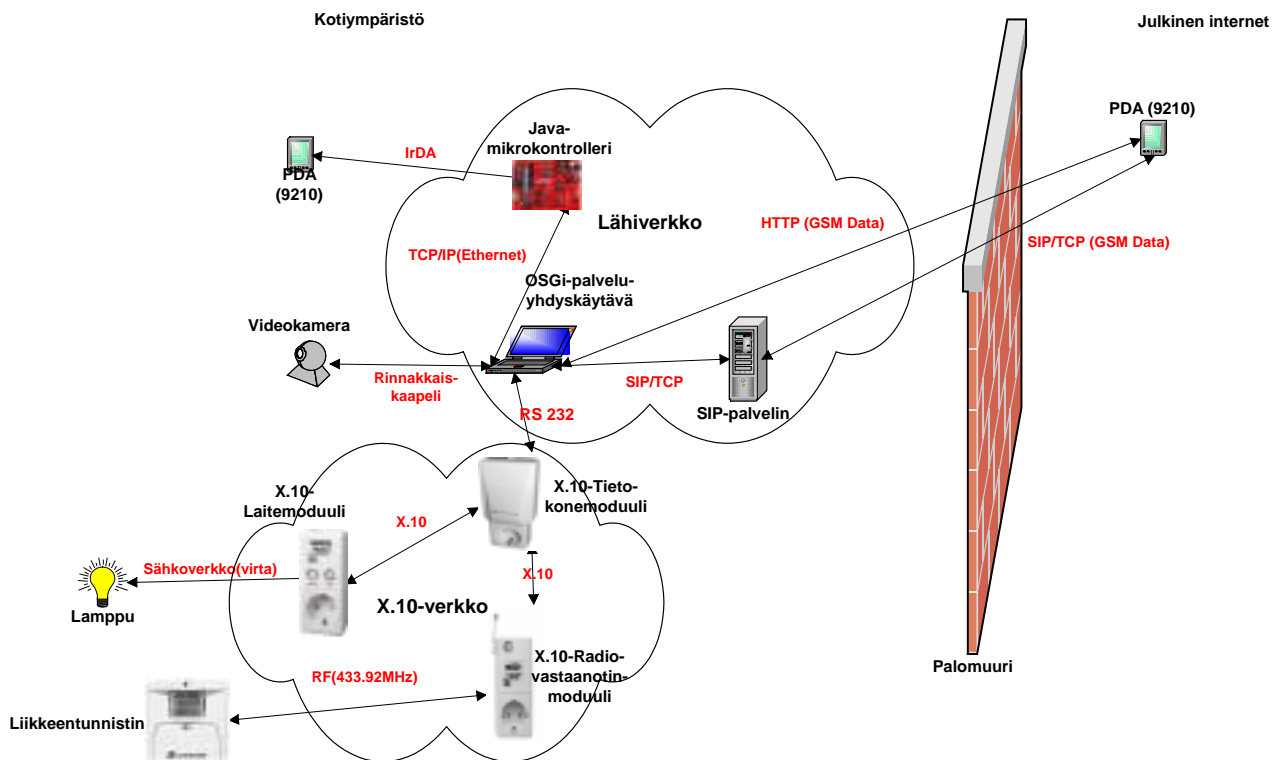
OSGi määrittelee joukon rajapintoja, joita voidaan kutsua ydinpalveluiksi (core service). Ne ovat yleiskäyttöisiä nippuja, joiden palvelut nippujen ohjelmoijat saavat käyttöönsä OSGi:n palvelurekisteriltä. HTTP-nippu toteuttaa palvelun, jonka avulla on mahdollista käyttää HTTP:aa kodin ja ulkopuolisen Internetin väliseen kommunikointiin (kuva 15). Lokipalvelua käyttämällä niput pystyvät tallentamaan niihin liittyvää lokitietoa, johon muut niput pääsevät käsiksi. Sitä käytetään pääasiassa tapahtumista ja virhetilanteista ilmoittamiseen. Laittepalvelu (device access manager) abstrahoi mallin laitteiden ja laiteajureiden hallitsemiseksi.

4. Järjestelmä ja käyttötapaukset

Tässä luvussa esitellään kodin verkotettujen laitteiden kontrolloimiseksi kehitetty järjestelmä [36], ohjelmiston arkkitehtuuri ja käyttötapaukset. Tämän diplomityön kontribuutio järjestelmässä esitellään käyttötapausten avulla.

4.1 Järjestelmä

Kodin laitteiden kontrollointiin kehitetty järjestelmä esitellään kuvassa 17. Kontrolloitavina verkotettuina laitteina käytettiin X.10-verkkoon liitettyä lamppua ja hälytysjärjestelmää. Kyseiset laitteet luokitellaan yksinkertaisiksi laitteiksi. Lamppu liitettiin X.10-verkkoon X.10-laitemoduulin avulla, ja sille oli mahdollista antaa ON/OFF-komentoja. Hälytysjärjestelmä muodostui palveluyhdyskäytävään rinnakkaiskaapelilla liitetystä videokamerasta, X.10-radiovastaanotinmoduulista ja liikkeentunnistimesta. X.10-verkko liitettiin palveluyhdyskäytävään RS-232-sarjaliitännällä X.10-tietokone-moduulia käyttäen. Hälytysjärjestelmällä on kolme eri tilaa (ON, IDLE ja ALARMING). IDLE-tilassa ei tapahdu hälytyksiä. ON-tilassa liikkeentunnistin ilmoittaa liikettä havaitessaan X.10-radiovastaanottimelle RF-signaalilla hälytyksestä, joka muutetaan X.10-signaaliksi. Tieto hälytyksestä saadaan palveluyhdyskäytävälle, ja hälytysjärjestelmä siirtyy ALARMING-tilaan. Hälytyksen tapahtuessa videokameralle annetaan komento JPG-kuvien ottamiseen, ja kyseiset kuvat tallennetaan OSGi-alustan kovalevyille. Palveluyhdyskäytävä toimii kotiympäristön ja julkisen Internetin välisenä solmukohtana, jonka kautta oli mahdollista kontrolloida kodin verkotettuja laitteita. Järjestelmässä käytettiin IBM:n OSGi-referenssitoteutusta Windows NT -käyttöjärjestelmässä.



Kuva 17. Toteutettu järjestelmä.

Kontrolloivana verkotettuna laitteena käytettiin Nokia 9210 Communicator PDA-laitetta, johon on toteutettu PersonalJava 1.1.1 -spesifikaation [37] mukainen Java-ohjelmointirajapinta EPOC-käyttöjärjestelmään. Kyseinen laite tukee myös IrDA-protokollaa. Java-mikrokontrollerissa on mahdollista suorittaa Java-ohjelmointikoodia, ja se tukee IrDA:a ja TCP/IP-protokollapinoa. Kotiympäristössä oltaessa hälytysjärjestelmää kontrolloitiin Java-mikrokontrollerin kautta käyttäen IrDA:a. Kodin ulkopuolelta tapahtuva VL:iden kontrollointi, tiedustelu ja kirjautuminen toteutettiin SIP-protokollalla ja sen laajennuksilla verkotettujen laitteiden tarjoamien palveluiden hyödyntämiseksi. SIP-viesteillä kommunikointiin käytettiin TCP:aa GSM-datapuheluiden avulla. Tilallisena SIP-välityspalvelimena kodin ja sen ulkopuolisen PDA:n välissä toimi Columbian yliopiston kehittämä SIPD Server 1.1.7, jota käytettiin Linux-käyttöjärjestelmässä. Sekä SIP-palvelimessa että OSGi-palveluyhdyskäytävässä käytettiin palomuuriohjelmistoja turvallisuuden takaamiseksi.

4.2 Ohjelmiston arkkitehtuuri

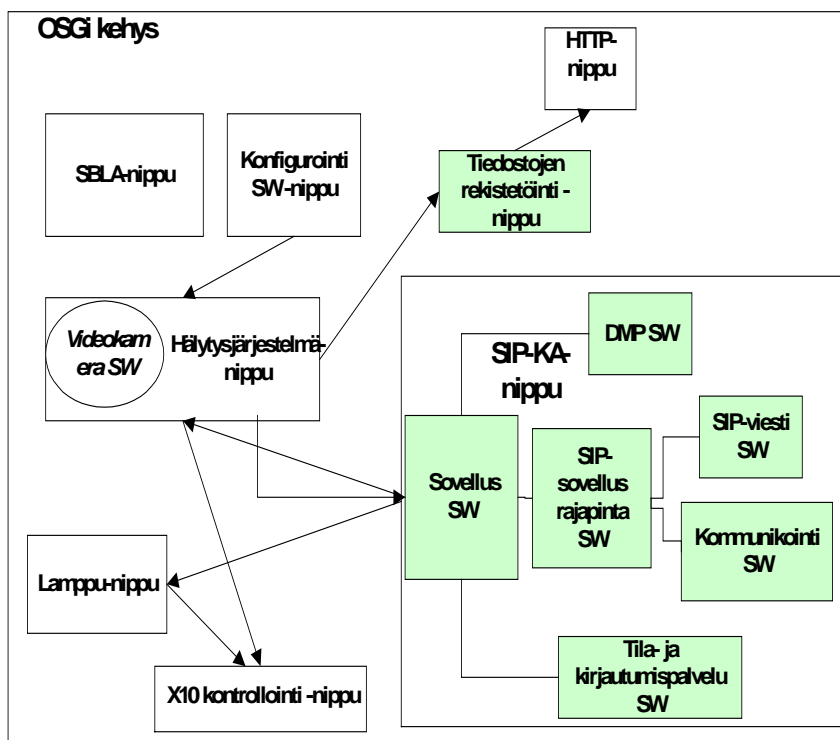
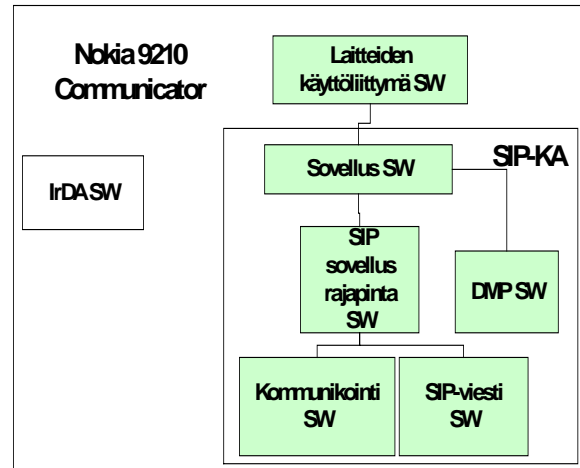
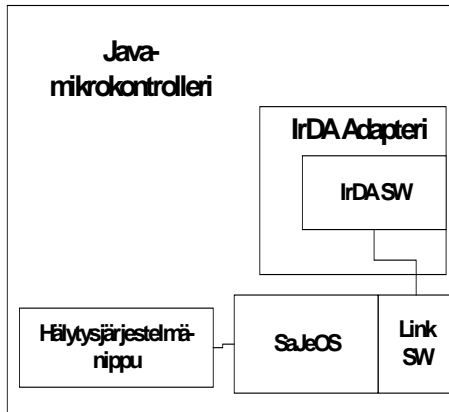
Järjestelmän toteuttamiseksi ohjelmistoja tehtiin Nokia 9210 Communicatorille, Java-mikrokontrollerille ja OSGi-alustalle, kuten kuvasta 18 huomataan. Tämän diplomityön osuus on merkitty tummalla värillä. Nuolet OSGi-alustalla tarkoittavat palveluriippuvuuksia nippujen välillä. IrDA SW, Link SW ja Konfigurointi SW -niput toteuttavat

fyysisen osoituksen kotiympäristössä PDA:n (9210) ja Java-mikrokontrollerin välillä käyttäen IrDA:a. X.10-Kontrollointi-nippu tarjoaa palvelun, jonka avulla on mahdollista kontrolloida X.10-verkkoa ja saada ilmoitus siihen kuuluvien laitteiden tapahtumista. Lamppu-nippu käyttää hyväksi tätä palvelua lampun tilan muuttamiseen ja tarjoaa SIP-KA:lle palvelun tämän toiminnon mahdollistamiseksi. Hälytysjärjestelmä-nippu kontrolloi hälytysjärjestelmää, ja sen tarjoama palvelu mahdollistaa hälytysjärjestelmän kontrolloinnin IrDA- ja SIP-protokollien avulla. Konfigurointi SW -nippu ja SIP-KA-nippu käyttävät hyväksi kyseistä palvelua. Hälytysjärjestelmä- ja Lamppu-nippu ilmoittavat läsnäolo- ja tilatietonsa SIP-KA-nipulle, joten ne toimivat verkotettujen laitteiden läsnäolo-käyttäjäagentteina. Kyseiset niput voidaan ajatella myös lampun ja hälytysjärjestelmän laitekontrollereiksi.

SIP-KA-nippu toimii SIP-KA:na OSGi-alustalla. Se pitää sijaintinsa päivitettyinä SIP-palvelimella ja vastaanottaa verkotettuihin laitteisiin liittyviä SIP-viestejä käyttäen SIP-laajennuksia verkotettuja laitteita varten. Kyseinen SIP-KA tietää lampun ja hälytysjärjestelmän läsnäolo- ja tilatiedot, joten se toimii lampun ja hälytysjärjestelmän läsnäoloagenttina. SIP-välityspalvelin ohjaa kirjautumiseen liittyvät viestit kodin OSGi-alustalle, joten se toimii viestejä välittävänä läsnäolopalvelimena. PDA:n SIP-KA pitää sijaintinsa päivitettyinä SIP-palvelimella ja tarjoaa verkotettujen laitteiden käyttöliittymille palvelun, jota hyväksikäyttämällä on mahdollista hyödyntää laitteiden palveluita kodin ulkopuolelta. Käyttöliittymät ladataan OSGi-alustalta PDA:lle HTTP:n avulla käyttötapauksessa 2 esitettävällä mekanismilla.

4.3 Käyttötapaukset

Seuraavaksi demonstroidaan kuvan 17 mukaisen järjestelmän toimintaa viidellä eri käyttötapauksella, joista neljä viimeistä ovat tämän diplomityön kontribuutiota. Eri laitteiden ja ohjelmistokomponenttien yhteistoimintaa valaisevat UML:n sekvenssidiagrammit ja SIP-viestikaaviot. SIP-viesteihin on otettu mukaan vain työn kannalta olennaisimmat otsikot, ja hyötykuormien DMP-viestit esitetään supistettuna.



Kuva 18. Ohjelmiston arkkitehtuuri.

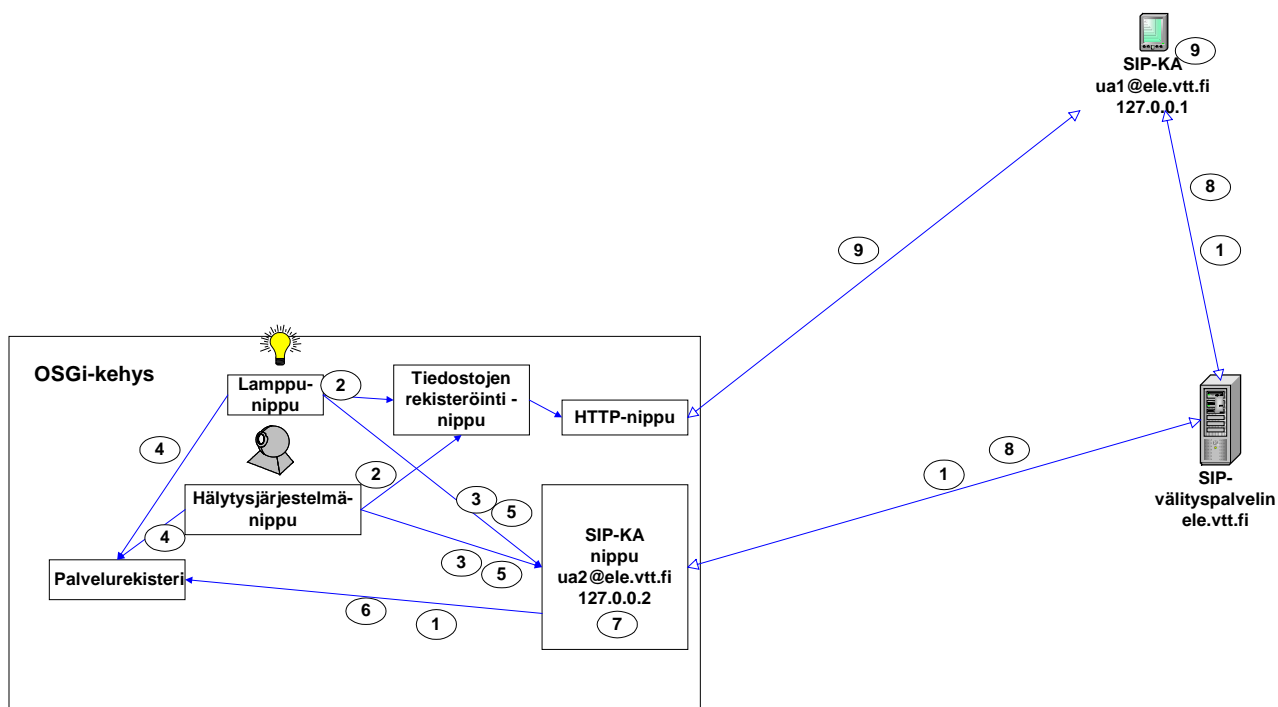
4.3.1 Laitteen fyysinen osoittaminen

Tässä käyttötapauksessa henkilö on kotiympäristössä ja kontrolloi hälytysjärjestelmää osoittamalla PDA-laitteella Java-mikrokontrolleria. Osoittamista kutsutaan fyysiseksi, koska tiedonsiirto laitteiden välillä tapahtuu IrDA:n [38] avulla. IrDA on lyhyen kantaman infrapuna-aaltoja hyväksi käyttävä protokolla. Mikrokontrollerin ohjelmisto ilmoittaa saamistaan komennoista OSGI-alustan Konfigurointi SW -nipulle, joka huolehtii hälytysjärjestelmän uuden tilan asettamisesta laitekontrollerina toimivan Hälytysjärjestelmä-nipun avulla. Näin kontrolloiva PDA-laite toimii hälytysjärjestelmän ohjaimena.

4.3.2 Rekisteröityminen ja käyttöliittymän lataaminen

Kun henkilö on kodin ulkopuolella, ei ole mahdollista käyttää kodin verkotettujen laitteiden palveluita samalla tavalla kuin edellisessä käyttötapauksessa. Tämä ongelma ratkaistiin toteuttamalla kullekin laitteelle käyttöliittymä, joka kapseloi kyseisen laitteen ominaisuudet ja palvelut. Nämä käyttöliittymät ladattiin kontrolloivaan laitteeseen, kuten seuraavat vaiheet osoittavat.

Kuvasta 19 nähdään tämän käyttötapauksen eri vaiheet. Vaiheessa 1 OSGi-alustalla ja PDA:ssa olevat SIP-KA:t (ua1@ele.vtt.fi, ua2@ele.vtt.fi) ilmoittavat sijaintinsa SIP-välityspalvelimelle. OSGi-alustalla oleva SIP-KA rekisteröi palvelunsa OSGi-kehiksen palvelurekisterille. PDA rekisteröityy lähettämällä REGISTER-pyyntö SIP-välityspalvelimelle ja saa vastauksena 200 OK -viestin (kuva 20). Expires-otsikko kertoo rekisteröinnin keston. OSGi-alustan SIP-KA:n rekisteröityminen tapahtuu identtisesti. Vaiheessa 2 lampun ja hälytysjärjestelmän niput asennetaan OSGi-alustalle. Kyseiset niput asettavat käyttöliittymäluokkansa HTTP:n avulla ladattaviksi käyttämällä Tiedostojen rekisteröinti -nipun palvelua (HTTPFileService-olio), joka nähdään kuvassa 21 (addFile-metodi). Tämän jälkeen edellisestä metodista paluuarvona saadut käyttöliittymätiedostojen HTTP-osoitteet ilmoitetaan SIP-KA:lle (addUILocations-metodi) vaiheessa 3. Osoitteet talletetaan parametrina annettavan laitetunnisteen (deviceID:n) perusteella, jonka avulla kaikki laitteet ovat osoitettavissa kodin ulkopuolelta. Kohdassa 4 kyseiset niput rekisteröivät palvelunsa OSGi-kehikselle. Näiden palveluiden avulla on mahdollista kontrolloida kyseisiä laitteita.



Kuva 19. Käyttötapaus 1.

Kohdassa 5 lampun ja hälytysjärjestelmän niput ilmoittavat läsnäolostaan SIP-KA:lle (devicePresence-metodi) ilmoittamalla laitetunnuksen, muuttujan ja muuttujan tilan eli tilatiedon. Vaiheessa 6 SIP-KA pyytää kyseisten verkotettujen laitteiden palveluita OS-Gi-kehykseltä (BundleContext-olio), jotka lampun ja hälytysjärjestelmän niput ovat aiemmin rekisteröineet (kohta 4). Palvelu saadaan palveluviittausolion (ServiceReferencen) avulla. Vaiheessa 7 SIP-sovellus (SIPApplication) käyttää hyväksi tila- ja kirjautumispalvelun (StateSubscriptionServicen) rajapintaa ja tallentaa verkotettujen laitteiden tilatiedot (addDeviceState). Kyseinen palvelu sijaitsee SIP-KA:ssa, jota käytetään verkotettujen laitteiden tila- ja läsnäolotietojen ja kirjautumisten tallentamiseen. Näin SIP-KA:lla on hallussa lampun ja hälytysjärjestelmän tila- ja läsnäolotiedot, käyttöliittymien HTTP-osoitteet ja kyseisten laitteiden kontrolloimiseen tarvittavat oliot.

SIP-välityspalvelin



PDA



1

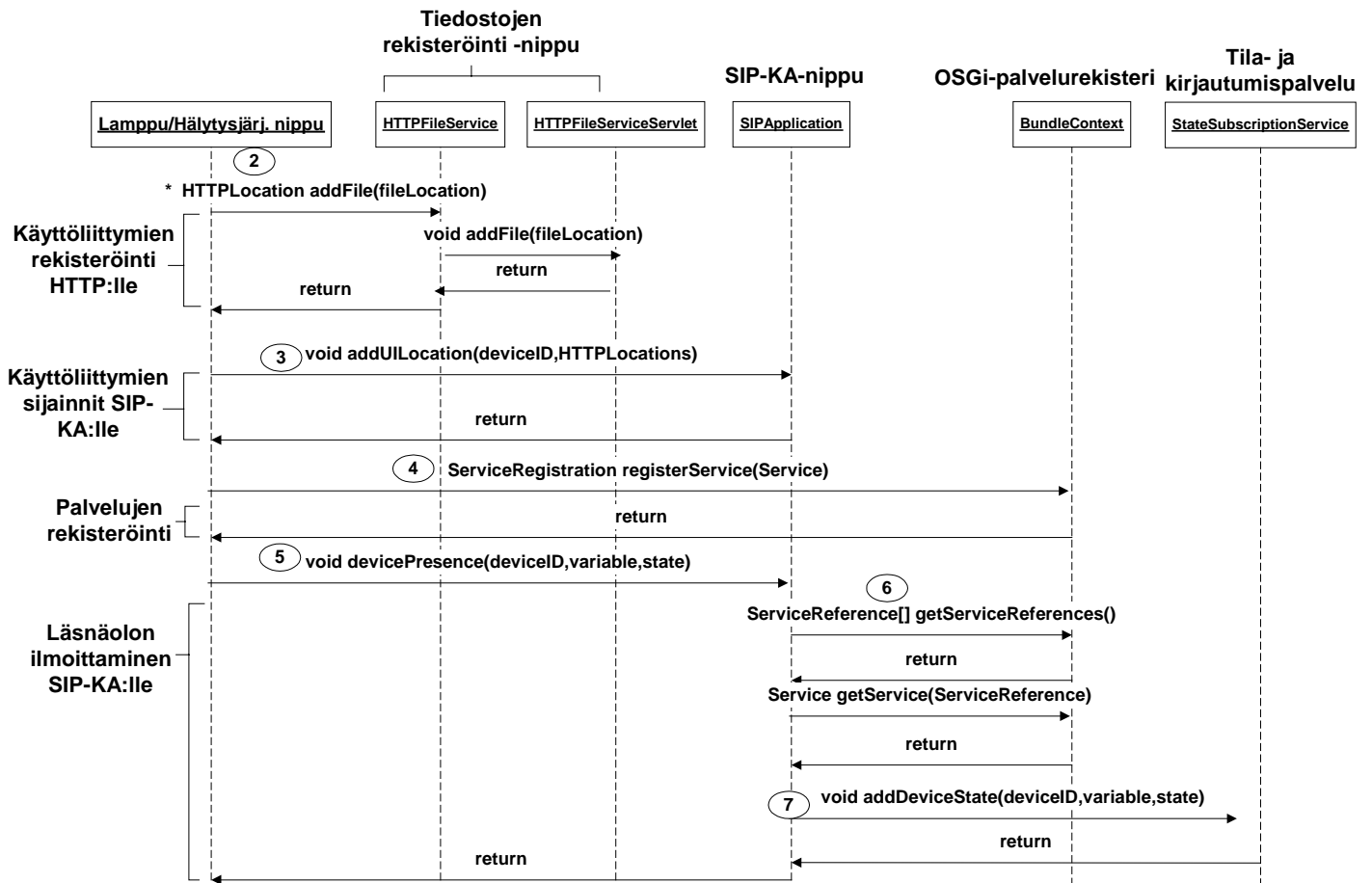
←

```
REGISTER sip:ele.vtt.fi;transport=TCP SIP/2.0
Via: SIP/2.0/TCP 127.0.0.1:4825
From: sip:ua1@ele.vtt.fi
To: sip:ua1@ele.vtt.fi
Call-ID: f62Qa11k1dUVQFgkbic9@ele.vtt.fi
CSeq: 1 REGISTER
Content-length: 0
Contact: sip:ua1@127.0.0.1:5060;transport=TCP
Expires: 5000
```

→

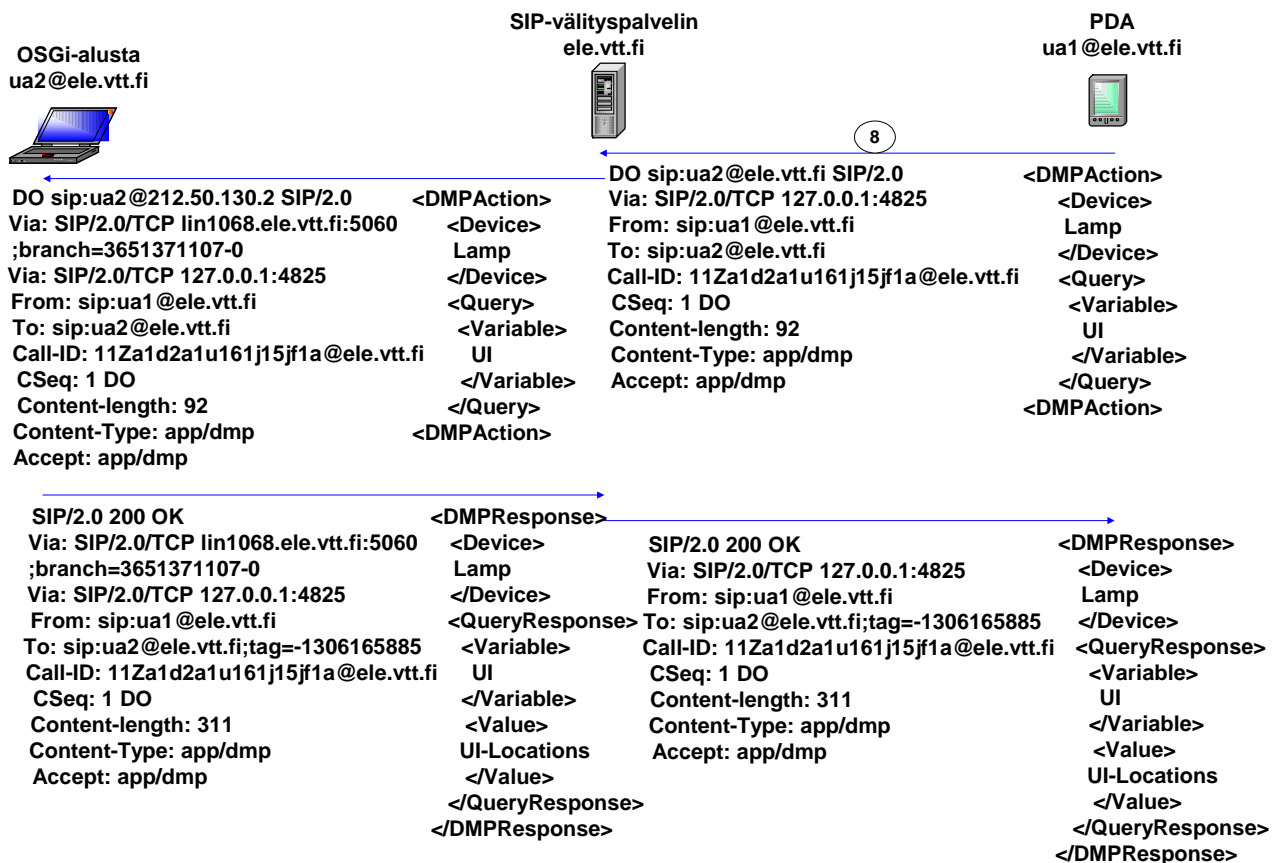
```
SIP/2.0 200 OK
Via: SIP/2.0/TCP 127.0.0.1:4825
From: sip:ua1@ele.vtt.fi
To: sip:ua1@ele.vtt.fi
Call-ID: f62Qa11k1dUVQFgkbic9@ele.vtt.fi
CSeq: 1 REGISTER
Content-length: 0
Contact: <sip:ua1@127.0.0.1:5060>;q=1.0;action=proxy;expires="Wed, 02 Jan 2002 11:36:53 GMT"
Expires: Wed, 02 Jan 2002 11:36:53 GMT
```

Kuva 20. PDA:n SIP-KA:n rekisteröityminen.



Kuva 21. Käyttötapauksen 1 sekvenssidiagrammi.

Kohdassa 8 ulkopuolinen käyttäjä (PDA) pyytää kodin lampun käyttöliittymää. Laitteeseen liittyvä tieto välitetään DO-pyyntöviestin DMP-tyyppisenä hyötykuormana (Query), kuten kuvasta 22 nähdään. Tähän DMP-tyyppiseen pyyntöviestiin sisältyy lampun ainutlaatuinen laitetunniste (Lamp), jonka perusteella käyttöliittymä ladataan. Hyötykuorman tyyppi (app/dmp) käy ilmi Content-type-otsikosta ja vastauksessa hyväksyttävien hyötykuormien tyypit Accept-otsikosta. DO-viesti lähetetään kodin SIP-KA:lle käyttäen SIP-välityspalvelinta, joka osaa ohjata viestin oikeaan osoitteeseen aikaisemman rekisteröinnin perusteella. Kodin SIP-KA muuntaa vastaanottamansa DO-viestin hyötykuorman sen ymmärtämäksi DMP-olioksi ja muodostaa vastaukseksi 200 OK -viestin, jonka hyötykuormassa (QueryResponse) on lampun käyttöliittymän luokkatiedostojen HTTP-osoitteet. Koska vastaanotetussa DO-viestissä oli useampi kuin yksi Via-otsikko, 200 OK -vastaukseen lisätään tag-parametri To-otsikkoon. Näin PDA osaa erottaa eri SIP-käyttäjäagenteilta tulleet vastaukset toisistaan haaroittavan välityspalvelimen tapauksessa. PDA tunnistaa 200 OK:n vastaukseksi lähetettyyn DO-viestiin To-, From-, Cseq- ja Call-ID-otsikoiden perusteella. DO ja 200 OK muodostavat siis transaktion.

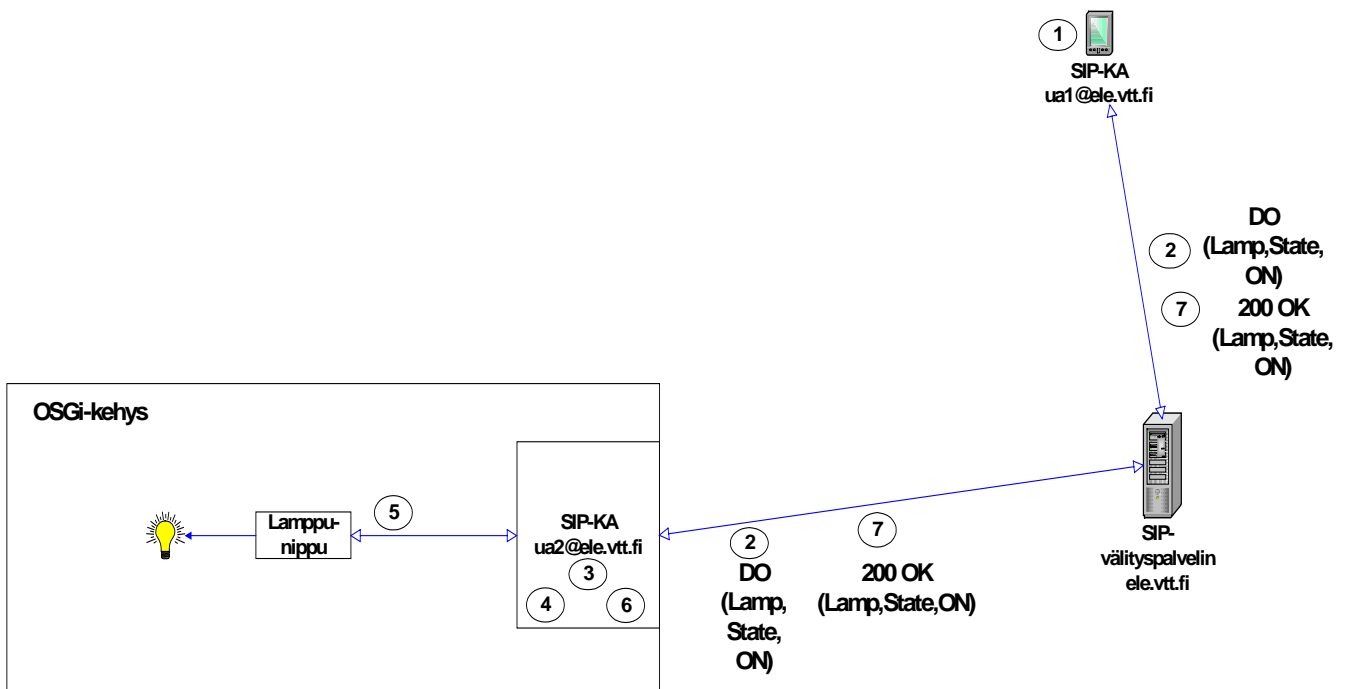


Kuva 22. Käyttöliittymän hakuun liittyvät SIP-viestit.

Kohdassa 9 PDA:n SIP-sovellus lataa lampun käyttöliittymäluokat OSGi-alustalta saamiensa osoitteiden perusteella HTTP:n GET-kutsuilla, tallentaa tiedostot ja luo uuden käyttöliittymäolion kyseisistä luokista käyttämällä Javan dynaamista luokanlatausta apuna. Näin lamppua kontrolloiva käyttöliittymä on ladattu ja luotu PDA-laiteeseen ja käyttäjän on mahdollista hyödyntää lampun palveluita. Käyttöliittymään sisältyvät verkottuneen laitteen ominaisuudet ja sen tarjoamat palvelut, kuten aikaisemmin mainittiin. Tämän tietämyksen perusteella muodostetaan DMP-tyyppisiä dokumentteja, joita käytetään SIP-viestien hyötykuormana verkotettujen laitteiden palveluiden hyödyntämiseen.

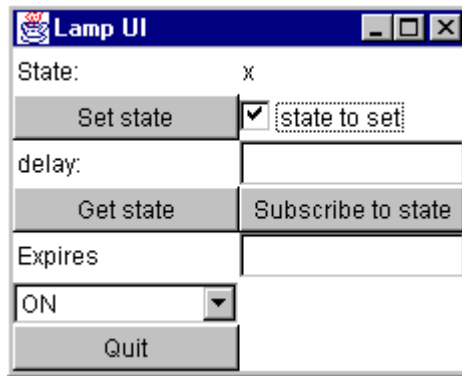
4.3.3 Kontrollointi

Toisen käyttötapauksen jälkeen on mahdollista hyödyntää verkotettujen laitteiden palveluita kodin ulkopuolelta. Tässä ja seuraavissa käyttötapauksissa lampun ja hälytysjärjestelmän tilalla tarkoitetaan kyseisten laitteiden "State"-nimisen muuttujan arvoa, jota voidaan pitää myös tilatietona. Kontrolloinnin vaiheet selviävät kuvasta 23 ja lampun käyttöliittymä on kuvassa 24.

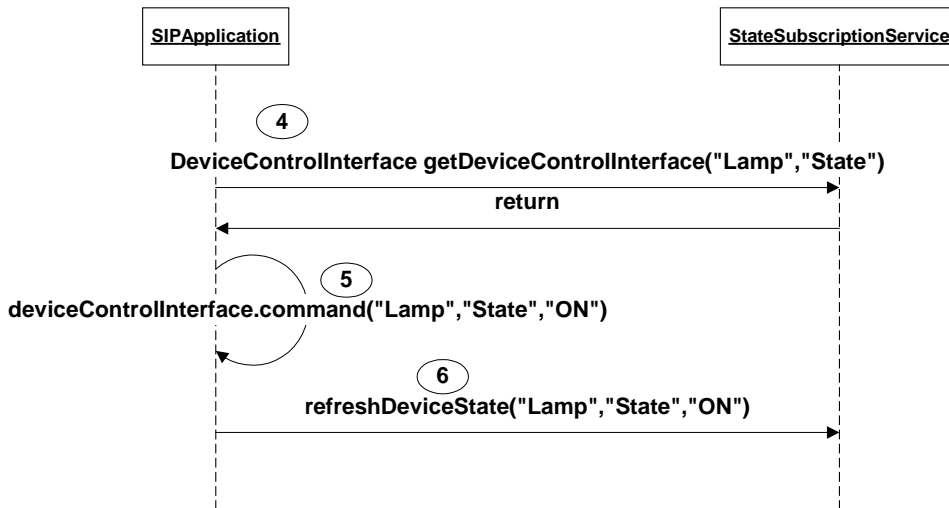


Kuva 23. Käyttötapaus 3.

Kohdassa 1 käyttäjä painaa "Set state" -näppäintä, jolloin lähetetään uusi DO-viesti kodin SIP-KA:lle SIP-välityspalvelimen kautta (kohta 2). DO- ja 200 OK-viestien sisällöt ovat kuten edellisessä käyttötapauksessa lukuun ottamatta hyötykuormaa. Pyyntöviestin hyötykuorma kertoo lampulle tarkoitetun käskyn ja vastauksen hyötykuorma lampun uuden tilan. Kuvassa 25 nähdään OSGi-alustan SIP-sovelluksen (SIPApplicationin) ja tila- ja kirjautumispalvelun (StateSubscriptionService) välinen kommunikointi DO-viestin vastaanottamisen jälkeen (kohta 3). Kohdassa 4 SIP-sovellus pyytää lampun tilan kontrollointiolion (`getDeviceControlInterface`-metodin), joka on aiemmin rekisteröity tila- ja kirjautumispalvelulle käyttötapauksessa 2. SIP-sovellus asettaa lampun uuden tilan kutsumalla kontrollointiolion `command`-metodia (kohta 5) ja päivittää lampun uuden tilan tila- ja kirjautumispalvelulle kohdassa 6 (`refreshDeviceState`-metodi). Tämän jälkeen lähetetään vastauksena 200 OK (kohta 7), joka kertoo lampun uuden tilan. PDA:n SIP-sovellus vastaanottaa viestin ja ilmoittaa käyttäjälle kuittauksen saapumisesta.



Kuva 24. Lampun käyttöliittymä.



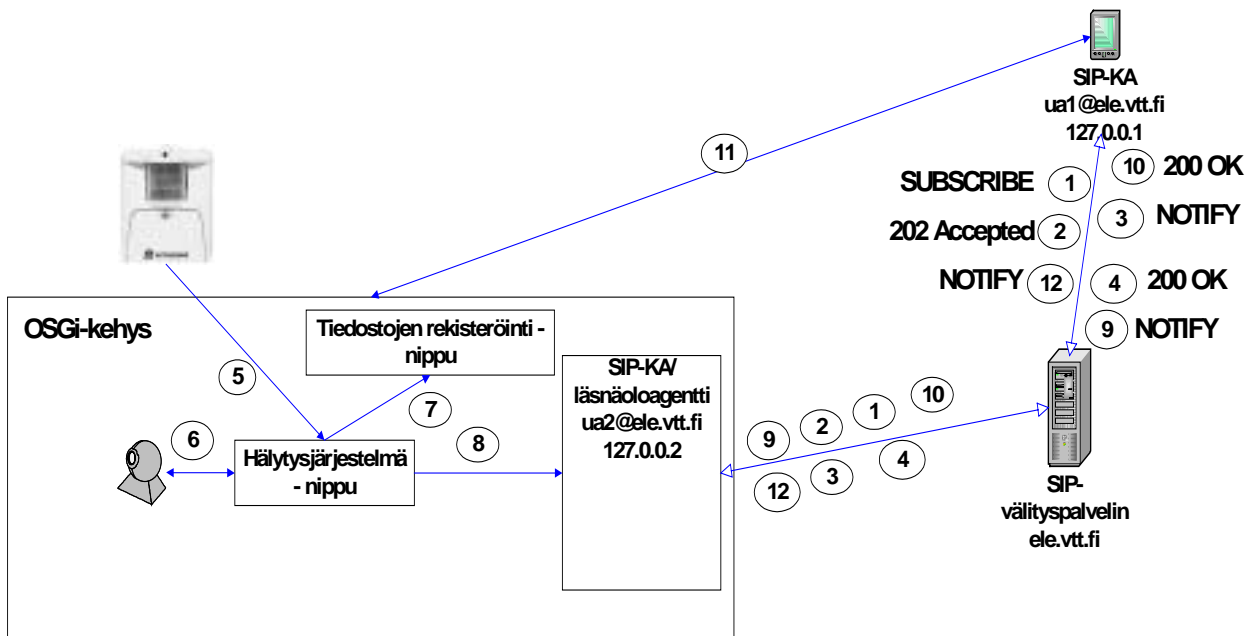
Kuva 25. Käyttötapauksen 3 sekvenssikaavio.

4.3.4 Tiedustelu

Tässä käyttötapauksessa oletetaan, että hälytysjärjestelmän käyttöliittymä on ladattu PDA-laitteeseen. Käyttäjä tiedustelee hälytysjärjestelmän tilaa painamalla "Get state" -näppäintä, jonka seurauksena lähetetään DO-viesti kodin SIP-KA:lle välityspalvelimen kautta. DO:n hyötykuorman DMP-dokumentti (Query) kertoo, minkä laitteen tilaa tiedustelu koskee. OSGi-alustalla oleva SIP-KA vastaanottaa kyseisen viestin ja kysyy tila- ja kirjautumispalvelulta hälytysjärjestelmän sen hetkisen tilan. Tiedusteluun liittyvä vastaus palautetaan 200 OK -viestin hyötykuormassa (QueryResponse). DO- ja 200 OK -viestien toiminta on kuten käyttötapauksessa 2. PDA:n SIP-KA tunnistaa 200 OK:n vastaukseksi DO-viestiin ja ilmoittaa hälytysjärjestelmän tilan käyttäjälle (käyttöliittymälle).

4.3.5 Kirjautuminen ja ilmoituksen saaminen

Tässä käyttötapauksessa ulkopuolinen käyttäjä kirjautuu hälytysjärjestelmän tilanmuutoksille ja haluaa ilmoituksen hälytyksen tapahtuessa. Kuva 26 kertoo tapahtumista eri vaiheissa ja kuva 27 tapaukseen liittyvistä SIP-viesteistä. Kohdassa 1 ulkopuolinen käyttäjä haluaa kirjautua hälytystä varten. Hälytysjärjestelmän käyttöliittymä on lähes samanlainen kuin lampun (kuva 24) vastaava. Käyttäjä asettaa kirjautumisen keston Expires-kenttään ja valitsee ALARMING-tilan hälytyksen ehdoksi listasta ja painaa ”Subscribe to state” -näppäintä. SUBSCRIBE-viesti lähetetään välityspalvelinta hyväksi käyttäen kodin SIP-KA:lle, joka toimii hälytysjärjestelmän läsnäoloagenttina. Verkottuneeseen laitteeseen liittyvä tieto sisältyy DMP-tyyppiseen hyötykuormaan (Subscribe), kuten aikaisemminkin. Kirjautumisen kesto ilmoitetaan sekä Expires-otsikossa että hyötykuormassa, koska on mahdollista, että SIP-viestejä ja DMP-dokumentteja käsitellään erikseen vastaanottavassa SIP-KA:ssa. Läsnäoloagentti hyväksyy pyynnön välittömästi 202 Accepted -vastauksella (kohta 2). Hyväksytyyn kirjautumisen kesto ilmoitetaan Expires-otsikossa, joka voi olla pienempi tai yhtä suuri kuin pyydetty kesto. SUBSCRIBE- ja 202 Accepted -viestit muodostavat uuden puheluhäärän, ja kirjautuja voi päivittää kirjautumisen lähettämällä uuden SUBSCRIBE-viestin, jossa on ilmoitettu uusi kesto ja CSeq-otsikon järjestysnumero on päivitetty (korotettu yhdellä). Kirjautuminen voidaan peruuttaa lähettämällä SUBSCRIBE, jonka kestoksi on asetettu nolla.



Kuva 26. Käyttötapaus 4.

202 Accepted -viestin lähettämisen jälkeen kodin SIP-KA muodostaa NOTIFYn (kohta 3), jonka hyötykuormassa kerrotaan hälytysjärjestelmän nykyinen tila (ON-tila). NOTIFY lähetetään SUBSCRIBE-viestin lähettäjän SIP-osoitteen perusteella SIP-välityspalvelimen kautta. PDA vastaa kyseiseen pyyntöön 200 OK:lla (kohta 4). NOTIFY on siis pyyntöviesti läsnäoloagentilta kirjautujalle, joka 200 OK:n kanssa muodostaa uuden puheluhäärän. Näin ollen NOTIFYn aloitusrivi sekä To- ja From-otsikot poikkeavat SUBSCRIBEn vastaavista ja CSeq-otsikon järjestysnumero on riippumaton kyseisestä kirjautumisviestistä. NOTIFYn Call-ID-otsikko on kuitenkin sama kuin SUBSCRIBEssa, jonka avulla PDA:n SIP-KA tunnistaa ilmoitukset kuuluviksi tiettyyn kirjautumiseen. Kirjautuja hyväksyy kuitenkin vain ne ilmoitukset, joiden From-otsikon tag-parametri on sama kuin 202 Accepted -vastausviestin To-otsikon tag-parametri. Tällä varmistetaan se, että ilmoitusten lähettäjänä toimii vain yksi läsnäolo-agentti, sillä SUBSCRIBE-viesti on voinut kulkeutua usealle SIP-KA:lle.

Kohdassa 5 liikkeentunnistin huomaa liikettä ja hälytysjärjestelmä laukeaa. Videokameraa käytetään JPG-kuvien tallentamiseen (kohta 6) OSGi-alustan kovalevyille. Tiedostojen rekisteröinti -nipun tarjoamaa palvelua käytetään apuna asettaessa JPG-kuvat ladattaviksi HTTP-protokollan kautta (kohta 7). Tämän jälkeen SIP-sovellukselle ilmoitetaan tapahtumasta (notifyDeviceState-metodi) kuvan 28 sekvenssikaavion mukaisesti (kohta 8). Tila- ja kirjautumispalvelua (StateSubscriptionService) käytetään VL:n tila- ja läsnäolotiedon sekä kirjautumisten tallentamiseen, joten tälle palvelulle ilmoitetaan hälytyksestä (refreshDeviceState-metodi). Tämän metodikutsun yhtenä parametrina annetaan JPG-kuvien HTTP-osoitteet (sideInformation-kenttä). Tila- ja kirjautumispalvelu vertaa hälytysjärjestelmän uutta tilaa vastaavien kirjautumisten vaatimuksiin ja muodostaa uuden notifiaktion niille SIP-KA:eille, joiden kohdalla ehto täyttyy. Koska kirjautuja on asettanut hälytyksen ehdoksi käyttöliittymän avulla ALARMING-tilan, luodaan ilmoitus tapahtumasta.

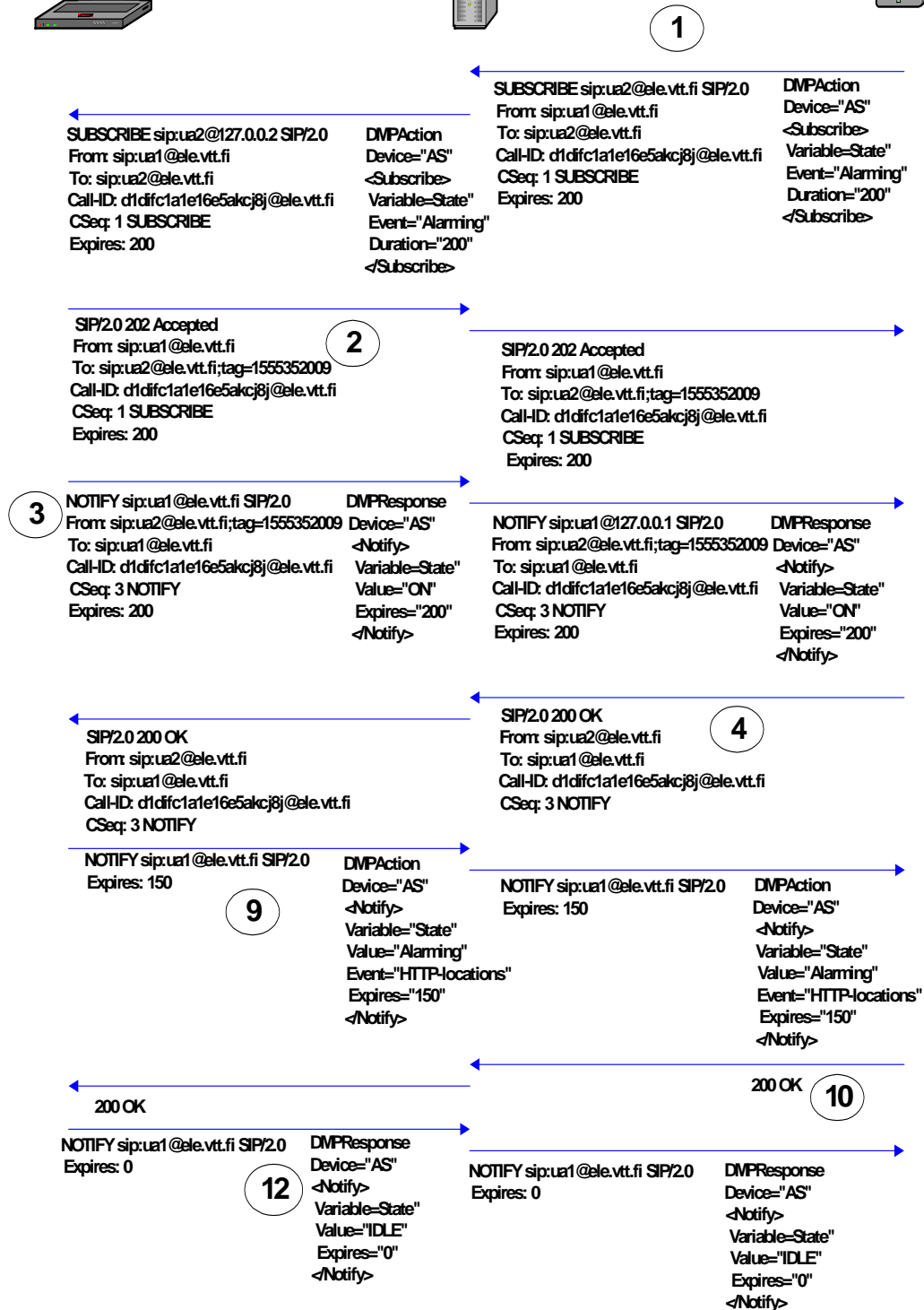
OSGi-alusta
ua2@ele.vtt.fi



SIP-välityspalvelin
ele.vtt.fi



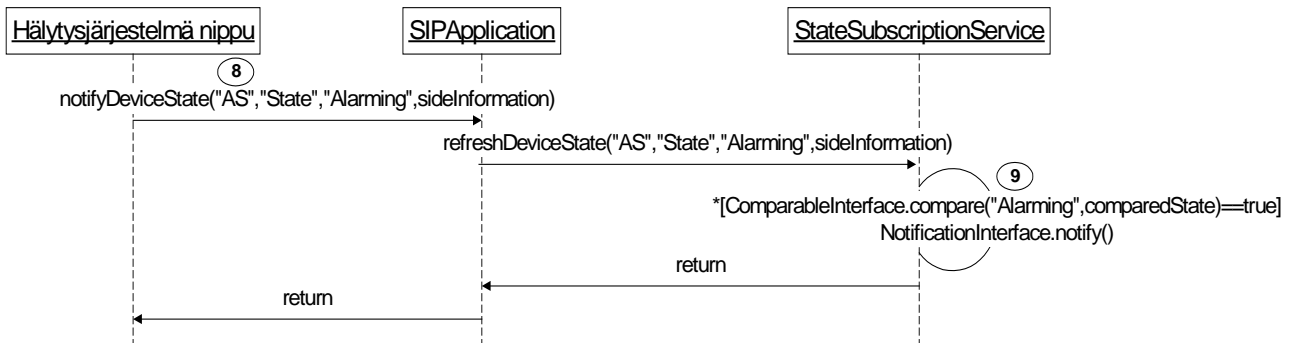
PDA
ua1@ele.vtt.fi



Kuva 27. Käyttötapausten 4 SIP-viestit.

Kohdassa 9 luodaan hälytyksen seurauksena uusi NOTIFY-viesti, joka lähetetään kirjautujalle. Ilmoituksen (NOTIFY) DMP-dokumentissa kerrotaan hälytysjärjestelmän laukeamisesta ja siihen liittyvän multimediatiedon HTTP-osoitteet. NOTIFYssa ilmoitetaan myös kirjautumisen jäljellä oleva kesto. PDA vastaa pyyntöön 200 OK -viestillä (kohta 10). Tämän jälkeen PDA:n SIP-sovellus käyttää HTTP-protokollaa JPG-kuvien hakemiseen OSGi-alustalta, jotka näytetään käyttäjälle kuvaesityksenä (kohta 11).

Kohdassa 12 kirjautuminen päättyy läsnäoloagentilla, josta ilmoitetaan PDA:lle NOTIFY-viestillä, jonka kestoksi on asetettu nolla sekuntia. Tähän pyyntöön ei vastata 200 OK -viestillä, koska SIP-välityspalvelin käsittää NOTIFYn pätevyuden loppuvan välittömästi, sillä Expires-otsikon kestoksi on asetettu nolla. Näin kyseinen palvelin ei jää odottamaan vastausviestejä vaan sulkee TCP-yhteyden palvelimen ja PDA:n väliltä heti lähetettyään NOTIFY-viestin määränpäähän. Tämä voitaisiin välttää lähettämällä ilmoitukset suoraan kirjautujalle SIP-välityspalvelimen sijaan. Tällöin kirjautujan tulisi ilmoittaa IP-osoitteensa esimerkiksi Contact-otsikon avulla.



Kuva 28. Käyttötapausten 4 sekvenssikaavio.

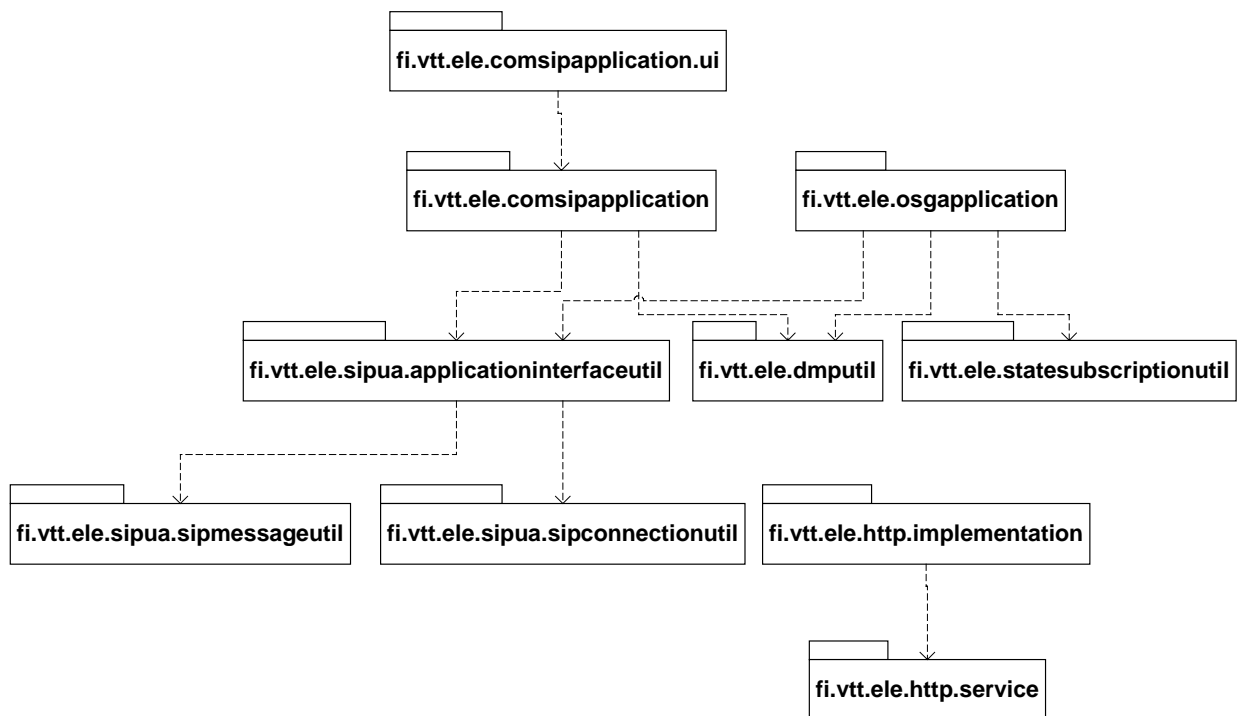
Tässä käyttötapauksessa kirjautuja saa ilmoituksen aina hälytyksen tapahtuessa (tilatiedon uusi arvo täyttää kirjautumisen ehdon) ja kirjautumisen päättyessä. Ilmoitus luodaan myös verkottuneen laitteen läsnäolotiedon muuttuessa, joten SIP-KA on verkotettujen laitteiden tilatietojen hallitsija ja läsnäoloagentti. On esimerkiksi mahdollista kirjautua verkotettujen laitteiden tilanmuutoksille, vaikka kyseisen laitteen tilatieto ei ole läsnäoloagentilla. Ilmoitus luodaan, kun tilatieto tulee saataville eli läsnäolo muuttuu.

5. Toteutus

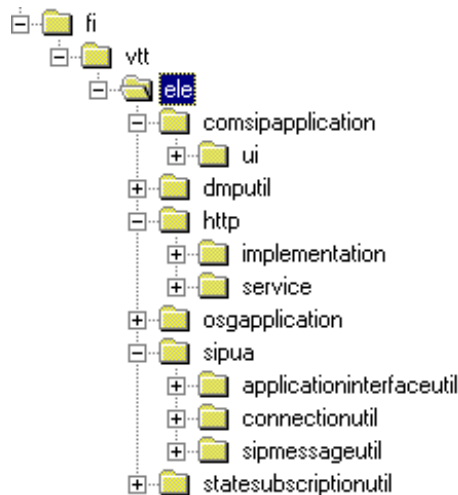
Diplomityötä varten toteutettiin koejärjestelmä, jossa tutkittiin SIP:n ja OSGi:n soveltuvuutta kodin verkotettujen laitteiden kontrolloimiseksi sekä suoritettiin edellä mainitut käyttötapaukset. Ohjelmiston kehityksen yhteydessä suoritettiin katselmointeja, vaikkakaan toteutusta ei ollut tarkoitus kehittää valmiin tuotteen tasolle.

Ohjelmistonkehitys suoritettiin emulaattorissa ennen siirtymistä varsinaiseen kohdeympäristöön. Nokia Communicator 9210 SDK:n Crystal Emulator -emulaattorin avulla oli mahdollista kehittää kyseisen PDA-laitteen Java-pohjaisia sovelluksia PC-ympäristössä. Tämä säästi huomattavasti lopullisissa tietoliikenneyhteyksissä ja helpotti työasemalla kehitetyn ohjelmiston siirtämistä itse sulautettuihin laitteisiin.

Tämän diplomityön kontribuutio oli toteuttaa SIP-KA ja käyttöliittymät lampun ja hälytysjärjestelmän kontrolloimiseksi. Toteutus tehtiin Java-kielellä, koska sekä OSGi-alusta että PDA-laite tukivat kyseistä ohjelmointikieltä. Koko järjestelmän arkkitehtuuri nähtiin kuvassa 18. SIP-KA:n pakettidiagrammi on kuvassa 29, jossa nuolet tarkoittavat riippuvuuksia ohjelmistokomponenttien välillä. Pakettien fyysinen sijainti hakemistorekenteena nähdään kuvassa 30. Tässä luvussa kuvataan kyseiset ohjelmistokomponentit.



Kuva 29. Ohjelmiston pakettidiagrammi.

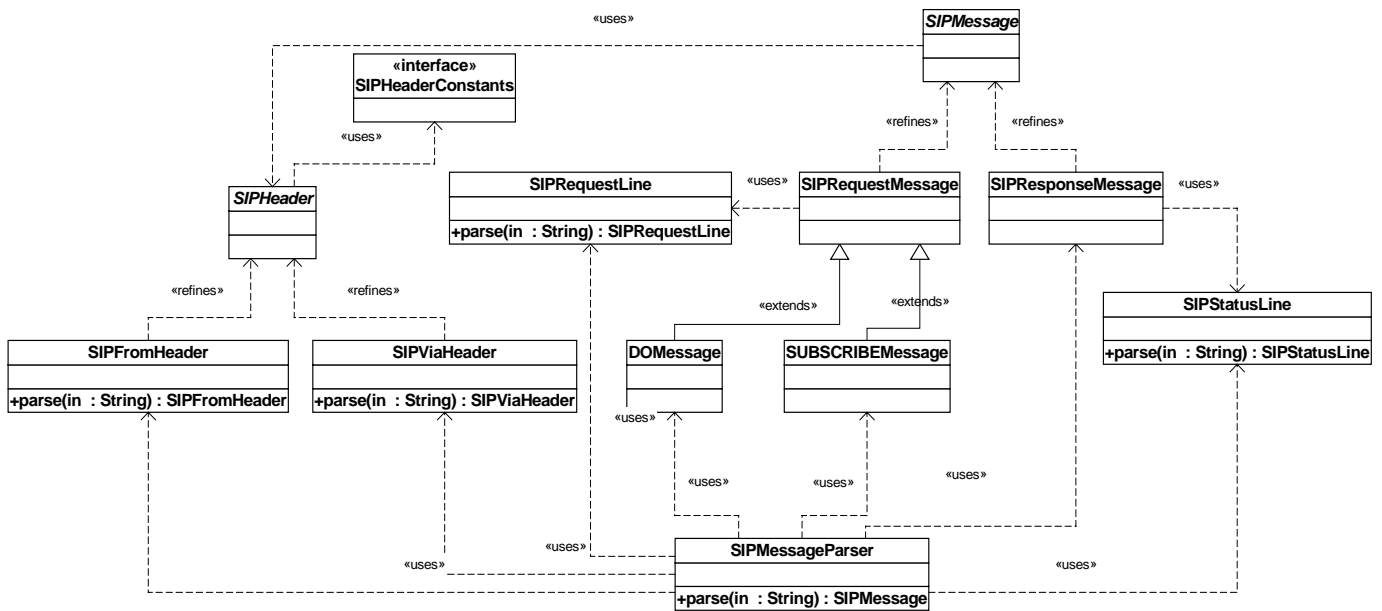


Kuva 30. SIP-KA:n hakemistorakenne.

5.1 SIP-viestit

SIP-viestien laajennukset verkotettuja laitteita varten on toteutettu paketissa `fi.vtt.ele.sipua.sipmessageutil`. Tämän paketin avulla on mahdollista luoda ja jäsentää tavuvirrasta REGISTER, DO, SUBSCRIBE ja NOTIFY pyyntöviestejä sekä kaikkia SIP-spesifikaatiossa [1] määriteltyjä vastausviestejä. Paketti tukee myös seuraavia SIP-otsikoita: Accept, AcceptEncoding, Allow, CallID, Contact, ContentEncoding, ContentLength, ContentType, CSeq, Expires, From, RecordRoute, Route, Require, Timestamp, To, Unsupported ja Via. Toteutuksen tärkeimmät luokat nähdään kuvan 31 luokkadiagrammista. SIP-otsikot (esimerkiksi `SIPFromHeader` ja `SIPViaHeader`) on toteutettu luokkina, jotka laajentavat abstraktia `SIPHeader`-luokkaa. `SIPMessage` on generiisen SIP-viestin ominaisuudet sisältävä luokka, jota pyyntö- (`SIPRequestMessage`) ja vastausviesti- (`SIPResponseMessage`) luokat toteuttavat. `SIPRequestLine`- ja `SIPStatusLine`-luokat kapseloivat pyyntö- ja vastausrivin ominaisuudet omaksi kokonaisuudekseen. Toiselta SIP-KA:lta tai SIP-palvelimelta tulleet viestit muunnetaan tavuvirrasta tietyiksi `SIPMessage`-olioiksi (esimerkiksi `DOMessage` tai `SUBSCRIBEMessage`) käyttämällä `SIPMessageParser`-luokkaa (`parse`-metodi). Tämä luokka käyttää hyväksi otsikoiden toteutuksia, koska niihin on kapseloitu tietämys tietyn otsikon jäsentämisestä. Myös pyyntö- ja vastausrivin tietämystä käytetään hyväksi (`parse`-metodit).

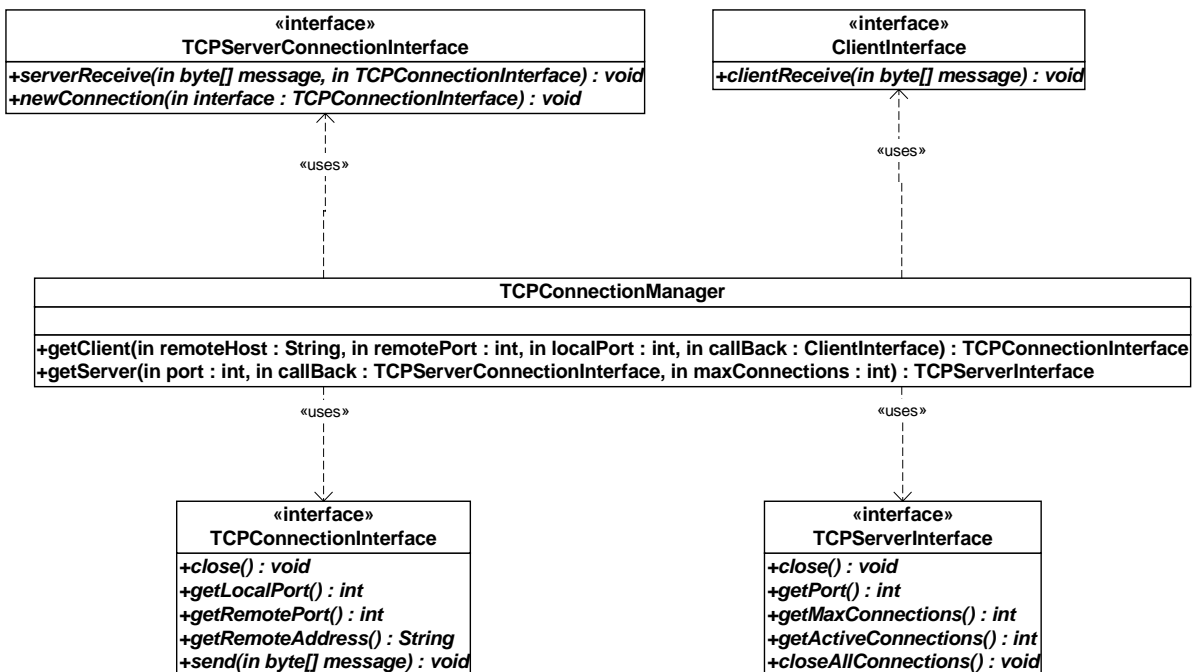
SIP-protokollan hyvä ominaisuus on sen laajennettavuus. Uusia pyyntöviestejä on mahdollista luoda periyttämällä uusi luokka `SIPRequestMessage`-luokasta. Vastausviestejä luodaan lisäämällä tietämys laajennusviestistä `SIPResponseMessage`-luokkaan. Uudet otsikot asetetaan toteuttamaan abstraktia `SIPHeader`-luokkaa.



Kuva 31. SIP-viestien tärkeimmät luokat.

5.2 Kommunikointi

Järjestelmän SIP-käyttäjäagenttien ja SIP-palvelimen välinen kommunikointi tapahtuu käyttämällä TCP-protokollaa. SIP-viestien välityksen mahdollistava ohjelmisto-komponentti on toteutettu paketissa fi.vtt.ele.sipua.connectionutil. Paketin tarjoama rajapinta mahdollistaa uusien TCP-asiakasyhteyksien luomisen etäterминаaleihin ja TCP-palvelimien asettamisen yhteyksien vastaanottamiseen.

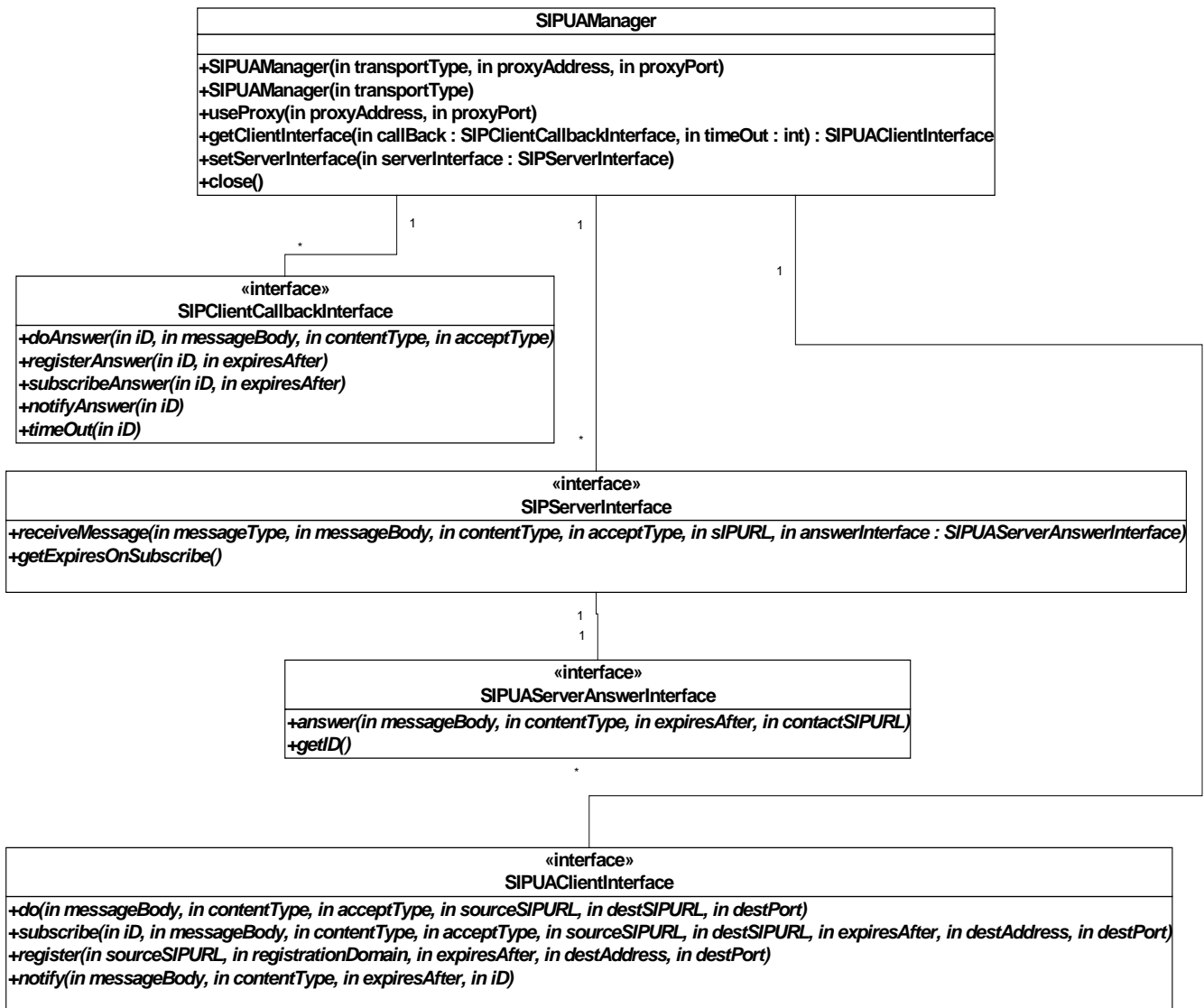


Kuva 32. Kommunikointiin käytettävän paketin rajapinnan luokkadiagrammi.

TCPCoannectionManager-luokan metodeja käytetään yhteyksien luomiseen ja vastaanottamiseen, kuten kuvan 32 luokkadiagrammista nähdään. Sen getClient- metodia käytetään uusien asiakasyhteyksien luomiseen. Kyseisen metodin parametriksi annetaan ClientInterface-rajapinnan toteutus, jonka avulla on mahdollista vastaanottaa palvelimen lähettämiä viestejä. Yhteyttä hallitaan paluuarvona saadulla oliolla, joka toteuttaa TCPConnectionInterface-rajapinnan. Tiettyä isäntäkoneen porttia kuuntelemalla on mahdollista vastaanottaa uusia asiakasyhteyksiä ja toimia siten yhteyden palvelimena. Tähän tarkoitukseen käytetään getServer-metodia, jolla on myös mahdollista määrittellä yhteyksien maksimimäärä kyseiseen porttiin. Uudet yhteydet vastaanotetaan TCPServerConnectionInterface-rajapinnan toteutuksella, joka annetaan parametriksi edellä mainitulle metodille. Tietyn portin palvelimen yhteyksiä hallitaan getServer-metodin paluuarvona saadulla TCPServerInterface-oliolla.

5.3 Sovellusrajapinta

fi.vtt.ele.sipua.applicationinterfaceutil-paketti käyttää hyväksi viesti- ja kommunikointipakettien palveluja ja tarjoaa rajapinnan SIP-sovellusten ohjelmoimiseen, joten se kapseloi osan SIP-protokollan toiminnasta (kuva 33). Sekä PDA:n että OSGi-alustan SIP-sovellukset hyödyntävät sen tarjoamaa rajapintaa.

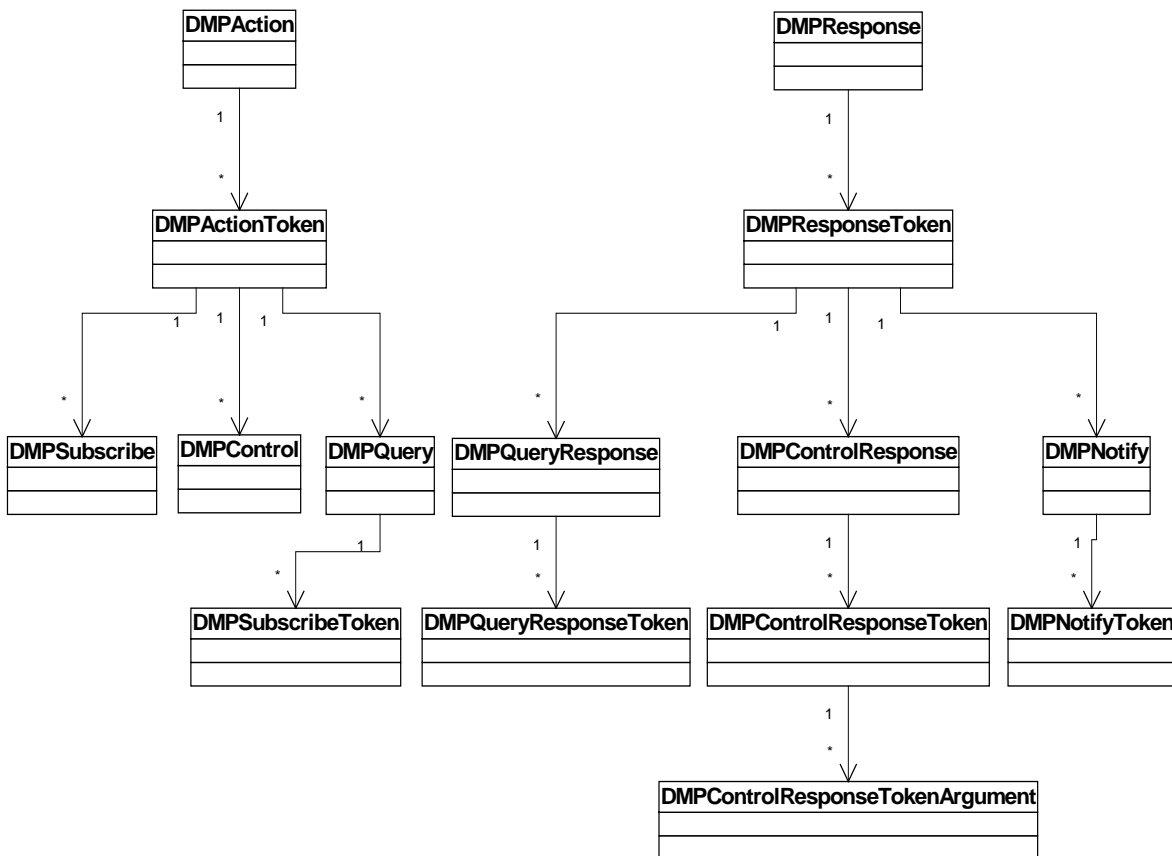


Kuva 33. Sovellusrajapinta.

Uusia SIP-KA-asiakkaita ja -palvelimia muodostetaan SIPUAManager-luokan avulla. SIP-KA-asiakas luodaan getClientInterface-metodilla. Paluuarvona saadaan SIPUAClientInterface-tyyppinen olio, jota käytetään uusien pyyntöviestien lähettämiseen. Sovelluksen ohjelmoija saa vastausviestit toteuttamalla SIPClientCallbackInterface-tyyppisen rajapinnan ja antamalla sen parametriksi getClientInterface-metodille. Uusi SIP-KA-palvelin muodostetaan setServerInterface-metodilla. Sen parametriksi annetaan SIPServerInterface-tyyppinen olio, jonka receiveMessage-metodin avulla vastaanotetaan pyyntöviestejä muilta SIP-käyttjäagenteilta. Edellä mainitun metodin parametrina saatua SIPUAServerAnswerInterface-tyyppistä toteutusta käyttämällä vastataan pyyntöviesteihin. SIP-KA voidaan konfiguroida käyttämään tiettyä välityspalvelinta SIPUAManager-luokan useProxy-metodin avulla.

5.4 DMP

Käyttötapausten yhteydessä nähtiin, kuinka DMP-tyyppisiä dokumentteja käytettiin SIP-viestien hyötykuormana. DMP-paketin (fi.vtt.ele.dmputil) palveluita hyödyntämällä on mahdollista luoda DMP-viestejä ja jäsentää muiden SIP-käyttjäagenttien lähettämät DMP-dokumentit sovellusten ymmärtämiksi DMP-olioiksi. DMP-viestien rakenne on kuvattu lähteessä [30, s. 4–8], jonka perusteella toteutus on tehty. Kuvassa 34 esitetään paketin luokkadiagrammi.



Kuva 34. DMP-paketin luokkadiagrammi.

5.5 Tila- ja kirjautumispalvelu

OSGi-alustan SIP-KA käyttää tila- ja kirjautumispalvelua (fi.vtt.ele.statesubscriptionutil) verkotettujen laitteiden tila- ja läsnäolotietojen sekä kirjautumisien tallentamiseen. Tilatieto koostuu laitetunnuksesta, laitteen muuttujasta ja muuttujan tilasta. Laitetunnus on ainutlaatuinen tunniste, jonka perusteella eri laitteet tunnistetaan. Laitteella voi olla useita muuttujia, joihin liittyy tila, kuten esimerkiksi mikroaaltouunilla on teho ja ajastin. Läsnäolotieto käsitetään tietyn laitteen muuttujan läsnäolona tila- ja kirjautumispalvelussa.

Tilätietoa lisättäessä yhtenä parametrina on mahdollista antaa verkotettujen laitteiden kontrollointi-rajapinnan toteutus. SIP-KA käyttää kyseistä oliota mahdollistaakseen ulkopuolisten käyttäjien kodin verkotettujen laitteiden kontrolloinnin. Kirjautumiset liitetään tietyn laitteen tilätietoon. Kirjautumista lisättäessä annetaan parametrina tilätietoon liittyvä ehto, jonka on toteuduttava, jotta ilmoitus (notifikaatio) muodostetaan. Ehto saadaan kirjautujalta. Kyseisen ehdon avulla päätetään kirjautumiseen liittyvän ilmoituksen luomisesta. Näin ulkopuolisen käyttäjän on mahdollista saada ilmoitus muuttujan saavuttaessa tietyn tilan tai aina, kun tila muuttuu. Ilmoitus luodaan myös aina, kun tilätieto lisätään tai poistetaan tila- ja kirjautumispalvelusta. Kaikkiin kirjautumisiin liittyy tietty kesto, jonka ajan kirjautumiset pidetään tallennettuina. Kirjautumisen keston päättyessä luodaan notifikaatio.

5.6 SIP-sovellukset

OSGi-alustalla oleva SIP-sovellus (fi.vtt.ele.osgapplication) käyttää hyväkseen sovellusrajapintaa, DMP- ja tila- ja kirjautumispalvelua sekä verkotettujen laitteiden rajapintoja muodostaakseen SIP-KA:n toiminnallisuuden. Se tulkitsee ulkopuolisten SIP-käyttäjäagenttien lähettämät viestit laitteille spesifisiksi komennoiksi eli toimii verkotettujen laitteiden ja SIP-KA:n välisenä rajapintana. Sovellus tarjoaa OSGi-alustalle palvelun, jota käyttämällä laitteenn on mahdollista ilmoittaa SIP-KA:lle laitteeseen liittyvät tilätiedot, kontrollointirajapinnan toteutus ja käyttöliittymäluokkien sijainnit. Sovellus ilmoittaa sijaintinsa SIP-välityspalvelimelle käyttäen REGISTER-viestejä. Lisäksi se vastaanottaa verkottuneisiin laitteisiin liittyviä DO- ja SUBSCRIBE-pyyntöviestejä sekä lähettää niihin liittyviä NOTIFY-ilmoituksia.

PDA:n SIP-sovellus (fi.vtt.ele.comsipapplication) käyttää sovellusrajapintaa ja DMP-pakettia muodostaakseen kontrolloivan laitteen SIP-KA:n. Kyseinen SIP-KA ilmoittaa sijaintinsa SIP-välityspalvelimelle. Sovellus tarjoaa käyttöliittymille rajapinnan, jonka avulla on mahdollista lähettää verkottuneisiin laitteisiin liittyviä DO- ja SUBSCRIBE-viestejä.

5.7 Käyttöliittymät

Käyttöliittymä (fi.vtt.ele.comsipapplication.ui) kapseloi verkottuneen laitteen tietämyksen ja sisältää kyseisen laitteen tarjoamat palvelut. Käyttöliittymät hyödyntävät PDA:n SIP-sovelluksen tarjoamaa rajapintaa SIP-viestien lähettämiseen ja vastaanottamiseen. Ne toteuttavat jokaiselle käyttöliittymälle yhteisen rajapinnan, jonka avulla kyseiseen laitteeseen liittyvät palvelut on mahdollista tarjota ulkopuoliselle käyttäjälle. Käytännössä lampun ja hälytysjärjestelmän käyttöliittymät ovat käännettyjä luokkatiedostoja,

joita asetetaan HTTP:n kautta ladattavaksi käyttämällä hyväksi Tiedostojen rekisteröinti -nipun tarjoamaa palvelua. Näin tietyn laitteen käyttöliittymä on mahdollista ladata kontrolloivaan verkottuneeseen laitteeseen tietämällä vain rajapina, jonka kyseinen käyttöliittymä toteuttaa.

5.8 Tiedostojen rekisteröinti

Tiedostojen rekisteröinti -nippu (fi.vtt.ele.http.implementation) käyttää OSGi:n määrittelemää HTTP-palvelua tarjotakseen rajapinnan (fi.vtt.ele.http.service), jonka avulla voidaan asettaa mikä tahansa tiedosto HTTP:n kautta ladattavaksi. Hälytysjärjestelmän ja lampun niput käyttävät tätä palvelua mahdollistaakseen käyttöliittymäluokkien ja JPG-kuvien saatavuuden kodin ulkopuolelta.

6. Evaluointi

Tässä luvussa evaluoidaan toteutusta suhteessa vaatimuksiin, jotka määriteltiin järjestelmälle johdannossa. Tämän lisäksi pohditaan SIP:n ja OSGi:n soveltuvuutta kodin verkotettujen laitteiden palveluiden hyödyntämiseen ja järjestelmän suhdetta olemassa oleviin ratkaisuihin.

6.1 Nimeäminen ja osoitteistus

Kodin verkottuneet laitteet erotettiin toisistaan ainutlaatuisten laitetunnisteiden avulla. Ulkopuolinen käyttäjä latasi käyttöliittymän PDA-laitteelleen käyttääkseen kyseisen laitteen palveluita. Hän tarvitsi tähän laitetunnisteen ja kodin SIP-osoitteen (sip:ua2@ele.vtt.fi). Ladattuun käyttöliittymään sisältyi laitteeseen liittyvä tietämys ja kyseisen laitteen tarjoamat palvelut.

Lähteessä [39, s. 4] esitetään vaihtoehto, jossa jokaisella laitteella on oma SIP-osoite. Tällöin verkotettujen laitteiden nimeämiseen käytettäisiin muotoa sip:laite@sijainti. Esimerkkeinä näissä tapauksissa laitteiden nimeämisestä olisivat seuraavat SIP-osoitteet: sip:kahvinkeitin@koti.net, sip:d=lamppu,r=makuuhuone@koti.net. Järjestelmän toteutuksessa laitteen osoitteistus sisältyi sekä SIP-osoitteeseen että hyötykuormaan, koska kodin OSGi-alustalla oli vain yksi SIP-KA, joka välitti lamppuun ja hälytysjärjestelmään liittyviä SIP-viestejä. Tämä johtui siitä, että kontrolloitavat laitteet olivat yksinkertaisia, jolloin ne eivät voineet sisältää SIP-KA:n ominaisuuksia.

6.2 Liikkuvuus

Kehitetyssä järjestelmässä verkottuneet laitteet ilmoittivat tietonsa kodin OSGi-alustalle asennetulle SIP-KA:lle. Kyseinen SIP-KA rekisteröi sijaintinsa oman alueen (ele.vtt.fi) SIP-välityspalvelimelle, joten kodin laitteet olivat osoitettavissa käyttäjäagentin SIP-osoitteen avulla. Voidaan kuitenkin olettaa, että laite siirretään toiseen kotiin, jolloin sen tiedot ilmoitetaan uuden kodin SIP-KA:lle. Tällöin ulkopuolisen käyttäjän on tiedettävä kyseisen uuden SIP-KA:n osoite. SIP-protokollan suurin etu tässä järjestelmässä on tuki SIP-KA:n liikkuvuudelle.

Kehitetty ratkaisu sopii hyvin yksinkertaisille laitteille, kuten lamppuille, joiden ei tarvitse sisältää SIP-protokollan tai TCP/IP-protokollapinon ominaisuuksia. On myös olemassa älykkäitä verkotettuja laitteita, jotka sisältävät SIP-KA:n ja joihin osoitetaan kodin ulkopuolelta laitteen omalla SIP-osoitteella, kuten edellisessä luvussa kerrottiin.

Tässä tapauksessa laitteet olisivat täysin liikkuvia ja ulkopuolisen käyttäjän tulisi tietää vain haluamansa laitteen SIP-osoite.

6.3 Käyttöliittymä

Käyttöliittymäluokkien lataamiseen käytettiin HTTP:ta, koska OSGi-spesifikaatio määrittelee HTTP-palvelun, jota oli luontevaa käyttää apuna luokkatiedostojen siirron mahdollistamiseksi. Tiedostojen lataamisen jälkeen käyttöliittymäolio muodostettiin käyttämällä apuna Javan dynaamista luokanlatausta. Tämä käyttöliittymäratkaisu kattaa kuitenkin vain osan päätelaitteista, koska kaikissa PDA-laitteissa ei ole Java-alustaa.

Käyttöliittymän tarkoituksena oli kapseloida verkottuneen laitteen ominaisuudet yhdeksi kokonaisuudeksi, jonka avulla oli mahdollista käyttää laitteen tarjoamia palveluita. Käyttöliittymä oli mahdollista ladata dynaamisesti ulkopuoliselle käyttäjälle, jonka jälkeen kommunikointi kontrolloivan laitteen ja palveluyhdyskäytävän välillä tapahtui SIP-viesteillä, jotka perustuivat käyttöliittymän sisältämään tietämykseen. Kohdassa 2.4 mainittuja palveluihin liittyviä tietotyyppejä oli mahdollista hyödyntää. Lampun virta voidaan ajatella tilatiedoksi, hälytysjärjestelmän saatavuus kommunikointiin läsnäolo-tiedoksi ja hälytyksen seurauksena tuotetut JPG-kuvat muuksi tiedoksi. Määritelmän "muu tieto" voi käsittää hyvin suuren skaalan erityyppisiä palveluita, kuten ääni- tai videovirtoja. Näin ollen järjestelmän soveltuvuutta kyseisten palveluiden hyödyntämiseksi ei voida arvioida luotettavasti. Verkotettujen laitteiden sijaintitiedoiksi voidaan luokitella palveluyhdyskäytävän SIP-KA:n välityspalvelimelle ilmoittama sijainti, joka on kuitenkin läpinäkyvä ulkopuoliselle käyttäjälle.

Olisi ollut myös suhteellisen yksinkertaista toteuttaa käyttöliittymä, joka olisi sisältänyt kaikkien kodin verkotettujen laitteiden käyttöliittymien sijainnit. Tällöin ulkopuolisen käyttäjän olisi tarvinnut ladata vain kyseinen käyttöliittymäolio, jonka avulla hän olisi voinut ladata loput käyttöliittymätiedostot.

6.4 Teknologiariippumattomuus

Järjestelmässä käytettiin yhtä olemassa olevaa teknologiaa (X.10) kodin laitteiden kanssa kommunikointiin. X.10-verkkoon oli liitettyä lampun ja liikkeentunnistin, joista jälkimmäinen oli osa hälytysjärjestelmää. Kummankin verkottuneen laitteen puolesta asennettiin OSGi-alustalle nippu, joka tarjosi palvelun laitteen kontrollointiin ja ilmoitti tilatiedot ja käyttöliittymän sijainnit SIP-KA:lle. SIP-sovellus muunsi SIP-viestit tarjoamaansa palveluun sopiviksi, ja VL:n nippu tulkitse tämän rajapinnan viestit laitespesifiselle rajapinnalle ja päin vastoin. Tällöin SIP-käyttäjäägenttien väliset SIP-viestit

ovat läpinäkyviä laitekontrollerina toimivalle nipulle. Koska kehitetty ratkaisu kuuluu sovelluskerrokseen, se ei ota kantaa siirto- tai saantiteknologioihin. Toteutuksen suhdetta muihin sovelluskerroksen teknologioihin on vaikea arvioida.

6.5 Kommunikointivaatimukset

Johdannossa mainitut kommunikointivaatimukset pystyttiin toteuttamaan. Kodin verkotettujen laitteiden tilatietojen kontrollointi ja tiedustelu toteutettiin DO-viestillä ja kirjautuminen sekä asynkronisten ilmoitusten luominen SUBSCRIBE- ja NOTIFY-viesteillä. Lisäksi SIP-viesteissä käytettiin uudenlaista hyötykuormaa (DMP.tä) näiden kommunikointivaatimusten mahdollistamiseksi. Nämä ominaisuudet oli yksinkertaista toteuttaa SIP:n hyvän laajennettavuuden vuoksi. Protokollaan on helppo lisätä sekä uusia viestejä että otsikoita. Hyötykuormana voidaan käyttää erityyppistä informaatiota, mutta viestin koko ei saa kasvaa liian suureksi, koska SIP ei ole tarkoitettu suurten tietomäärien siirtoon [29, s. 140].

6.5.1 Läsnaolotieto

Määritelmän mukaisella läsnäolotiedolla [33] käsitetään usein henkilön saatavuutta eri laitteiden avulla. Kehitettyssä järjestelmässä läsnäolo liittyy tiettyyn verkotettuun laitteeseen ihmisen sijaan, joten kyseistä aihetta käsiteltiin eri näkökulmasta. Laitteen saatavuus liittyy käyttöliittymään ja laitteen tilatietoihin. Käyttöliittymän saatavuus voidaan ajatella osaksi laitteen läsnäoloa, koska verkotetun laitteen palvelut sisältyivät käyttöliittymään. Käyttöliittymän lataamisen jälkeen tarkkailijan oli mahdollista kirjautua laitteen tilan muutoksille. Kun halutun tiedon läsnäolo muuttui, tarkkailija sai ilmoituksen tapahtumasta. SUBSCRIBE- ja NOTIFY-viestit soveltuivat hyvin kirjautumiseen ja asynkronisiin ilmoituksiin. Järjestelmässä ei käsitelty määritelmän mukaista läsnäoloon liittyvää dokumenttia [33], vaan saatavuus ilmoitettiin supistetusti SIP-viestin hyötykuormassa.

6.6 Skaalautuvuus

Ratkaisu ei ole kovin hyvin skaalautuva, sillä jokaisella verkotetulla laitteella täytyy olla maailmanlaajuisesti ainutlaatuinen laitetunniste, jos kontrolloitavien laitteiden liikkuvuus kotien välillä halutaan tehdä mahdolliseksi. Aiemmin esitetty mahdollisuus, jossa jokaista laitetta kohden on olemassa SIP-osoite, on paremmin skaalautuva. Tällöin ei tarvittaisi laitetunnistetta, vaan tiettyyn laitteeseen viittaaminen tapahtuisi SIP-osoitteen avulla (tapaan sip:laite@sijainti). Mikäli kyseinen SIP-KA sijaitsisi itse laitteessa, tämä vaatisi laitteelta omaa prosessoria ja Internet-ominaisuuksia, mikä ei ole mahdollista

kaikissa tapauksissa (vrt. yksinkertainen lamppu). SIP-KA voisi sijaita myös toisessa älykkäässä laitteessa, joka suorittaisi SIP-viesteihin liittyvät toiminnot verkotetun laitteen puolesta. Tällöin SIP-KA:n täytyisi prosessoida useiden laitteiden SIP-viestit, jotta ratkaisu olisi skaalautuva, sillä kodeissa on useita yksinkertaisia laitteita.

6.7 Järjestelmän suhde alan viimeisimpään kehitykseen

Järjestelmän laitteet voidaan nimetä SIP-osoitteilla. Kohdassa 2.5 mainituissa ratkaisussa ei ole mahdollista käyttää vastaavanlaista osoitteistusta, joka mahdollistaisi kodin älykkäiden IP-laitteiden liikkuvuuden, kuten kohdassa 6.2 todettiin. Kyseiset ratkaisut tukevat kuitenkin kontrolloivan laitteen liikkuvuutta. Jos kontrolloivan laitteen ja palveluyhdyskäytävän välillä käytetään HTTP:tä, kontrolloivassa päätelaitteessa on oltava HTTP-palvelin, jotta asynkroninen viestinvälitys on mahdollista. Soneran Talonmies-tuote [22] tukee käyttäjän liikkuvuutta ja asynkronista kommunikointia, kuten kehitetty järjestelmäkin.

Lähteessä [24] esitellyssä ratkaisussa käyttöliittymät luodaan ajon aikaisesti kontrolloiville laitteille UIML-kuvausten perusteella. Tämä ratkaisu on paras olemassa olevista järjestelmistä, koska se mahdollistaa riippumattomuuden kontrolloivan laitteen ohjelmointitympäristöstä. Tässä työssä kehitetyssä järjestelmässä vaaditaan Java-virtuaalikoneen olemassaolo PDA-laitteessa.

Tässä diplomityössä esitettävän järjestelmän suurin ero olemassa oleviin ratkaisuihin on SIP:n soveltuvuus verkkoistuntojen kuvaamiseen. Vaikka järjestelmässä ei demonstroitu istuntojen muodostamista, tiedetään, että SIP:n SDP-hyötykuorman avulla voidaan kuvata esimerkiksi RTP-virran muodostaminen kodin videokameran ja kontrolloivan laitteen välille. SIP:han on alun perin kehitetty verkkoistuntojen muodostamista varten.

6.8 SIP:n soveltuvuus

Verkotetun laitteen käyttöliittymä oli ladattava kodin OSGi-alustalta kontrolloivaan laitteeseen, ennen kuin VL:n palveluiden käyttäminen oli mahdollista. Tämän jälkeen SIP soveltui hyvin kyseiseen laitteeseen liittyvään kontrollointiin, tiedusteluun ja kirjautumiseen. Nämä kommunikointiin liittyvät ominaisuudet oli helppo toteuttaa SIP:n hyvän laajennettavuuden vuoksi. Myös läsnäoloon liittyvää käsitettä oli mahdollista soveltaa SIP:n avulla. Jos verkotetut laitteet olisi nimetty SIP-osoitteilla, ratkaisu olisi ollut skaalautuvampi ja tukenut paremmin liikkuvuutta, kuten todettiin aikaisemmin. Todettiin, että SIP on riippumaton sovelluskerroksen alapuolella käytettävistä teknologioista. Sen sijaan SIP:n suhdetta muihin sovelluskerroksen teknologioihin ei osattu

arvioida. Työ ei koskenut turvallisuutta, luotettavuutta eikä tunnistusta, joten SIP:n soveltuvuudesta aihepiiriin ei saatu täysin kattavaa kuvaa.

Ehdotettuja SIP:n laajennuksia voidaan käyttää myös tarjottaessa palveluja kodin verkotetuille laitteille. Tällöin ulkopuolisen palveluntarjoajan palvelut olisivat riippumattomia kodin laitteiden sijainnista ja ominaisuuksista [40] eli palvelut olisivat portattavia ja keskitetty tiettyyn paikkaan.

6.8.1 SIP ja HTTP

SIP ja HTTP ovat molemmat sovelluskerroksen protokollia. Käyttöliittymän lataaminen tapahtui käyttämällä HTTP:tä. SIP:n käyttämiselle HTTP:n sijaan on olemassa useita syitä.

Suurin syy on SIP:n tarjoama tuki liikkuvuudelle. Lisäksi yhden SIP-osoitteen kautta on mahdollista saada yhteys useaan SIP-KA:iin, joten läsnäoloon liittyviä käsitteitä on mahdollista soveltaa SIP:n avulla. Kyseinen protokolla on alunperin suunniteltu tukemaan verkkoistuntojen luomista. Tämä on myös suuri etu HTTP:hen verrattuna. SIP on riippumaton siirtoon käytettävästä tavasta, mutta HTTP-viestejä lähetetään TCP:n avulla. Näin ollen SIP-viesteillä kommunikointi on kevyemmin toteutettavissa. Kommunikointivaatimukset olisi voitu toteuttaa HTTP:n GET- ja POST-viesteillä. HTTP toteuttaa kuitenkin asiakas-palvelin-mallin, joten se ei sovellu hyvin asynkroniseen kommunikointiin. HTTP:n suurin etu SIP:hen verrattuna on sen luonteva käyttö suurten tietomäärien siirtoon.

6.9 OSGi:n soveltuvuus

OSGi-kehiksen ominaisuuksia tutkittiin suppeammin kuin SIP-protokollaa, ja tarkoituksena oli yhdistää SIP:n ja OSGi:n parhaat ominaisuudet verkotettujen laitteiden palveluiden hyödyntämiseksi. OSGi:n parhaat puolet olivat kodin verkkojen ja niiden laitteiden yhdistäminen, nippujen palvelujen dynaamisuus ja tuki käyttöliittymätiedostojen sekä JPG-kuvien jakamiselle HTTP-palvelun avulla. OSGi toimi siis hyvin kodin laitteiden palveluyhdyskäytävänä, ja SIP:n ja OSGi:n yhdistäminen teki edellä kuvatut käyttötapaukset mahdolliseksi. Oma mielipiteeni on, että pelkästään toisen teknologian soveltaminen käyttötapauksien ratkaisemiseen ei ole mahdollista ilman toisen jättämien aukkojen paikkaamista jollakin uudella teknologialla. Voisi olla myös hyvä ajatus lisätä SIP-protokollaa tukevat rajapinnat OSGi-spesifikaation ydinpalveluksi. Tämä yhdistäisi kummankin teknologian ominaisuudet luontevalla tavalla.

7. Yhteenveto

Diplomityön aiheena oli tutkia SIP-protokollan ja OSGi:n soveltuvuutta kodin verkotettujen laitteiden kontrollointiin ja palveluiden hyödyntämiseen painottaen SIP:tä. Tutkimusta rajattiin tietyin kriteerein ja kehitettävälle järjestelmälle asetettiin tavoitteet. Järjestelmä koostui kotona olevasta OSGi-palveluyhdyskäytävästä, siihen liitetystä lampusta ja hälytysjärjestelmästä, SIP-palvelimesta ja kontrolloivasta laitteesta. Oma kontribuutioni oli toteuttaa SIP-käyttäjäagentti kodin OSGi-alustalle ja kontrolloivaan Nokia 9210 Communicator -laitteeseen. Kyseisen PDA-laitteen avulla oli mahdollista käyttää kodin verkotettujen laitteiden tarjoamia palveluita. Järjestelmän toimintaa demonstroitii käyttötapauksilla. Tämän jälkeen suoritettiin evaluointi, joka perustui järjestelmälle asetettuihin vaatimuksiin ja käyttötapauksen suorittamisesta saatuihin kokemuksiin. Tämän perusteella todettiin SIP:n soveltuvan parhaiten verkotettujen laitteiden kommunikointivaatimusten toteuttamiseen kyseisen protokollan hyvän laajennettavuuden vuoksi. SIP:n muut hyvät puolet ovat sen tuki laitteiden liikkuvuudelle, riippumattomuus sovelluserroksen alapuolisista teknologioista ja mahdollisuus laitteiden nimeämiseen SIP-osoitteilla. OSGi:n paras puoli oli sen toimivuus kodin ja ulkopuolisen Internetin yhdyskäytävänä, johon oli mahdollista liittää dynaamisesti laitteita ja jota voitiin käyttää palveluiden tarjoamiseen. Kehitetyn järjestelmän etujen olemassa oleviin ratkaisuihin verrattuna todettiin olevan kodin älykkäiden IP-laitteiden liikkuvuus ja mahdollisuus verkkoistuntojen luomiseen SIP:n avulla.

Lähdeluettelo

1. Handley, M., Schulzrinne, H., Schooler E. & Rosenber, J. 6.1.2002. Session Initiation Protocol. Internet Engineering Task Forum RFC 2543. URL: <http://www.ietf.org/rfc/rfc2543.txt?number=2543>.
2. Open Services Gateway Initiative Specification (OSGi). 6.1.2002. URL: http://www.osgi.org/resources/spec_download.asp.
3. Edens, G. T. 2001. Home Networking and the CableHome Project at CableLabs. IEEE Communications Magazine 39, nro. 6, s. 112–121.
4. Holtz, R. et al. 2.2.2002. Guide to Home Networks. URL: <http://www.ce.org/networkguide/default.asp>.
5. Rose, B. Home Networks: A Standards Perspective. IEEE Communications Magazine 39, nro. 12, s. 78–85.
6. X-10 Technology and Research Forum. 17.1.2002. URL: <http://www.x10.org>.
7. The Bluetooth SIG. 17.2.2002. URL: <http://www.bluetooth.org>.
8. Aragon, M., Romano, S., Tummins, J. & Stachelek, M. 20.2.2002. Residential Gateway Viability. URL: <http://citeseer.nj.nec.com/cachedpage/240325/1>.
9. Aihkisalo, T. 2001. Kotiautomaatio-ohjelmistojen etäylläpito ja -kehitys. Diplomityö. Oulun yliopisto, sähkötekniikan osasto, Oulu. 80 s.
10. Lernel, M., Vanecek, G., Vidovic, N. & Vrsalovic, D. 2000. Middleware Networks Concept, Design and Deployment of Internet Infrastructure. Kluwer Academic Publishers, Boston/Dordrecht/London. 375 s.
11. Moyer, S., Marples, D. & Tsang, S. 2001. A Protocol for Wide-Area Secure Networked Appliance Communication. IEEE Communications Magazine 39, nro. 10, s. 52–59.
12. Gillet, S. E., Lehr, W. H., Wroclawski, J. T. & Clark, D. D. 2001. Do Appliances Threaten Internet Innovation. IEEE Communications Magazine 39, nro. 10, s. 46–51.
13. HAVi Home Audio/Video Interoperability. 7.2.2002. URL: <http://www.havi.org>.

14. IEEE Standardi 1394. 1995.
15. Hasedawa, A. & Nakajima, T. 2001. A User Interface System for Home Appliances with Virtual Network Computing. International Workshop on Distributed Computing Systems, April 16–19, Phoenix, Arizona, USA. S. 229–234.
16. Richardson, T., Stafford-Fraser, Q., Wood, K. R. & Hopper, A. 1998. Virtual Network Computing. IEEE Internet Computing 2, nro. 1, s. 33–38.
17. Corcoran, P. M., Papai, F. & Zoldi, A. 1998. User Interface Technologies for Home Appliances and Networks. IEEE Transactions on Consumer Electronics 44, nro. 3, s. 679–685.
18. Wendorft, R. G., Udink, R. T. & Bodlaender, M. P. 2001. Remote Execution of HAVi Applications on Internet-Enabled Devices. IEEE Transactions on Consumer Electronics 47, nro. 3, s. 485–495.
19. Simple Object Access Protocol (SOAP) 1.1 W3C. URL: <http://www.w3.org/TR/SOAP>.
20. Yoshida, R., Inoue, A., Hiraishi, J., Shigeno, H. & Matsushita, Y. 2002. EXWeb: Remotely Operating Devices in the Home Network. 4th IEEE International Workshop on Networked Appliances, January 15–16, Gaithersburg, Maryland, USA. S. 267–274.
21. Jini 12.2.2002. URL: <http://www.jini.org>.
22. Sonera Talonmies 22.2.2002. URL: <http://www.sonera.net/in/talonmies>.
23. User Interface Markup Language 17.1.2002. URL: <http://www.uiml.org>.
24. Plomp, J., Keränen, H., Yli-Nikkola, H. & Rantakokko, T. 2002. Supporting past, present and future interaction with home appliances. International ITEA Workshop on Virtual Home Environments, Paderborn, Saksa, February 20–21. S. 51–60.
25. VoiceXML 25.2.2002. URL: <http://www.voicexml.org>.
26. Buker, U. & Sturm, O. 2002. Secure Remote Control of Home Appliances. International ITEA Workshop on Virtual Home Environments, Paderborn, Saksa, February 20–21. S. 183–185.

27. Third Generation Partnership Program (3GPP). 6.1.2002. URL: <http://www.3gpp.org>.
28. Rosenberg, J. et al. 2001. SIP Extensions for Instant Messaging. IETF draft. URL: <http://www.ietf.org/internet-drafts/draft-ietf-sip-message-00.txt>.
29. Schulzrinne, H. & Rosenberg, J. 2000. The Session Initiation Protocol: Internet-Centric Signaling. IEEE Communications Magazine 38, nro. 10, s.134–141.
30. Khurana, S., Kurung, P. & Dutta, A. 6.1.2002. Device Messaging Protocol (DMP): An XML based format for Wide Area Communication with Networked Appliances. IETF draft. URL: <http://www.argreenhouse.com/iapp/draft-khurana-dmp-appliances-00.txt>.
31. Tsang, S., Moyer, S., Marples, D., Schulzrinne, H. & Roychowdhury, A. 6.1.2002. SIP Extensions for Communicating with Networked Appliances. IETF draft. URL: <http://www.argreenhouse.com/iapp/draft-tsang-sip-appliances-do-00.txt>.
32. Roach, A. 7.1.2002. Event Notification in SIP. IETF draft. URL: <http://beatles.csel.it/mirrors/drafts/draft-roach-sip-subscribe-notify-03.txt>.
33. Day, M., Rosenberg, J. & Sugano, H. 7.1.2002. A Model for Presence and Instance Messaging. IETF RFC 2778. URL: <http://www.ietf.org/rfc/rfc2778.txt?number=2778>.
34. Rosenberg, J. et al. 7.1.2002. SIP Extensions for Presence. IETF draft. URL: <http://www.ietf.org/internet-drafts/draft-ietf-simple-presence-04.txt>.
35. Marples, D. & Kriens, P. 2001. The Open Services Gateway Initiative: An Introductory Overview. IEEE Communications Magazine 39, nro. 12, s. 110–114.
36. Latvakoski, E. J., Pääkkönen, P., Pakkala, D., Välitälo, P., Remes, J. & Tikkala, A. 2002. Interaction of All IP Mobile Internet Devices with Networked Appliances in Residential Home. Hyväksytty IWSAWC'2002. URL: <http://www.mkg.sfc.keio.ac.jp/IWSAWC>. Julkaistaan: The Proceedings of IEEE International Conference on Distributed Computing Systems Workshops (IEEE Computer Society) -julkaisussa heinäkuussa 2002.
37. Personal Java Specification 1.1.1. 7.1.2002. URL: <http://java.sun.com/products/personaljava/spec-1-1-1/index.html>.
38. Infrared Data Association IrDA. 7.1.2002. URL: <http://www.irda.org>.

39. Moyer, S. et al. 17.1.2002. Framework Draft for Networked Appliances using the Session Initiation Protocol. IETF draft. URL:
<http://www.argreenhouse.com/iapp/draft-moyer-sip-appliances-framework-02.pdf>.
40. Moyer, S., Marples, D., Tsang, S. & Ghosh, A. 2002. Service Portability of Networked Appliances. IEEE Communications Magazine 40, nro. 1, s.116–121.



Tekijä(t) Pekka Pääkkönen			
Nimeke Kodin verkotettujen laitteiden palveluiden hyödyntäminen			
Tiivistelmä Diplomityö käsittelee kodin verkotettujen laitteiden palveluiden hyödyntämistä kotiympäristön ulkopuolelta. Palveluiksi käsitetään mikä tahansa kodin laitteen tarjoama tieto, jota ulkopuolinen käyttäjä voi hyödyntää. Tämän tiedon käsittely aiheuttaa usein toimintoja kontrolloitavassa laitteessa. Suurimmat ongelmat aiheeseen liittyen ovat lukuisat kodin laitteiden verkotukseen käytettävät teknologiat ja laitteiden kontrolloinnin turvallisuus. Työn tavoitteena on toteuttaa kotona olevien yksinkertaisten laitteiden palveluiden hyödyntäminen kodin ulkopuolelta käyttäen liikkuvaa päätelaitetta. Työssä tutkitaan SIP- ja OSGi-teknologioiden soveltuvuutta tähän aihepiiriin. Tutkimusta on rajattu siten, että palveluiden hyödyntämisen turvallisuuteen, tunnistukseen ja luotettavuuteen ei oteta kantaa, ja SIP:n osuutta painotetaan suhteessa OSGi:een. Näihin teknologioihin liittyen tutkitaan laitteiden nimeämistä, osoitteistusta, liikkuvuutta, käyttöliittymiä, teknologiariippumattomuutta, kommunikaatiovaatimuksia, skaalautuvuutta ja läsnäoloa. Tätä varten toteutettiin järjestelmä, jonka avulla oli mahdollista käyttää kodin lampun ja hälytysjärjestelmän tarjoamia palveluita liikkuvalla päätelaitteella. Ohjelmisto toteutettiin Java-kielellä sen alustariippumattomuuden ja verkkotuen vuoksi. Järjestelmän toimintaa demonstroidaan viidellä käytötapauksella, jotka havainnollistavat kyseisten laitteiden käyttöliittymän lataamista ja sen jälkeen tapahtuvaa kommunikointia. Käyttötapausten kulkua kuvataan SIP-viestikaavioilla ja ohjelmiston rakenne selitetään UML-menetelmää apuna käyttäen. Lopuksi suoritettujen evaluoinnin perusteella todettiin SIP:n olevan helposti laajennettava ja tukevan laitteiden liikkuvuutta ja nimeämistä. Se soveltui myös hyvin kommunikointivaatimusten toteuttamiseen. OSGi:n parhaaksi puoleksi huomattiin sen toimivuus kodin ja ulkopuolisen Internetin välisenä dynaamisena palvelualustana.			
Avainsanat SIP, OSGi, service gateways, presence			
Toimintayksikkö VTT Elektroniikka, Kaitoväylä 1, PL 1100, 90571 OULU			
ISBN 951-38-6073-6 (URL: http://www.inf.vtt.fi/pdf/)		Projektinnumero E2SU00085	
Julkaisuaika Syyskuu 2002	Kieli suomi	Sivuja 69 s.	Hinta
Projektin nimi PLA_MIDA		Toimeksiantaja(t)	
Avainnimeke ja ISSN VTT Tiedotteita – Research Notes 1455-0865 (URL: http://www.inf.vtt.fi/pdf/)		VTT Tietopalvelu PL 2000, 02044 VTT Puh. (09) 456 4404 Faksi (09) 456 4374	

Published by



Vuorimiehentie 5, P.O.Box 2000, FIN-02044 VTT, Finland
Phone internat. +358 9 4561
Fax +358 9 456 4374

Series title, number and
report code of publication

VTT Research Notes 2157
VTT-TIED-2157

Author(s) Pekka Pääkkönen			
Title The utilization of services provided by networked home appliances			
Abstract <p>This diploma thesis deals with the utilization of services provided by networked home appliances from outside of the home environment. Services are considered to be any information provided by a home appliance which the outside user can utilize. The processing of this information often produces functions in the controlled home devices. The biggest problems related to this subject are the wide variety of technologies used to network home appliances and the safety related to the control of these devices. The goal of this thesis is to implement the utilization of services from outside of the home environment offered by simple home networked appliances using a nomadic terminal. The applicability of SIP- and OSGi-technologies are studied in relation to this subject matter. The research is limited in that it does not deal with security, identification and the reliability of the utilized services and the stress on SIP's part in this study is emphasized in its relation to OSGi. Related to these technologies research is conducted on the naming, addressing, mobility, user interfaces, technological independence, requirements for communication, scalability and presence of various devices. To enable this, a system was implemented with which services provided by a home's lamp and alarm system were able to be utilized with a nomadic terminal. Software was implemented with the Java programming language because of its support for platform independence and networking. The operation of the system was demonstrated with five use cases which illustrated the downloading of the user interfaces of the devices in question and the communication which takes place after it. The progress of the use cases is illustrated with the aid of SIP message charts and the structure of the software is depicted with UML. Based on an evaluation performed at the end, SIP was recognized to be easily expandable and to support the mobility and naming of the devices. It was also found to be suitable for the implementation of the communication requirements. OSGi's best feature was noticed to be its functionality as a dynamic service gateway between the home and the outside Internet.</p>			
Keywords SIP, OSGi, service gateways, presence			
Activity unit VTT Electronics, Kaitoväylä 1, P.O.Box 1100, FIN-90571 OULU, Finland			
ISBN 951-38-6073-6 (URL: http://www.inf.vtt.fi/pdf/)		Project number E2SU00085	
Date September 2002	Language Finnish	Pages 69 p.	Price
Name of project PLA_MIDA		Commissioned by	
Series title and ISSN VTT Tiedotteita – Research Notes 1455-0865 (URL: http://www.inf.vtt.fi/pdf/)		VTT Information Service P.O.Box 2000, FIN-02044 VTT, Finland Phone internat. +358 9 456 4404 Fax +358 9 456 4374	

VTT TIEDOTTEITA – RESEARCH NOTES

VTT ELEKTRONIIKKA – VTT ELEKTRONIK – VTT ELECTRONICS

- 1777 Röning, Juha, Kalaoja, Jarmo, Okkonen, Ari & Kauniskangas, Hannu. Reaaliaikaisten konenäkösovellusten kehittäminen. 1996. 72 s. + liitt. 40 s.
- 1816 Pyhälä, Timo. Ohjelmistokomponenttien rajapintojen kuvaaminen. 1997. 55 s. + liitt. 22 s.
- 1825 Heimala, Päivi, Hokkanen, Ari, Keinänen, Kari, Keränen, Kimmo, Tenhunen, Jussi & Lehto, Ari. Mikroanturisyteemien tutkimusohjelma 1994–1996. 1997. 47 s.
- 1908 Tuominen, Arno. Joustavat ohjelmistoratkaisut tehtäväkriittisessä hajautetussa järjestelmässä. 1998. 74 s.
- 1911 Holappa, Mikko S. CORBAn soveltaminen joustavan valmistusjärjestelmän perusohjelmistoon. 1998. 95 s.
- 1913 Salmela, Mika. Testausympäristön konfigurointityökalun käytettävyyden parantaminen. 1998. 56 s.
- 1914 Korpipää, Tomi. Hajautusalustan suunnittelu reaaliaikasovelluksessa. 1998. 56 s. + liitt. 4 s.
- 1927 Lumpus, Jarmo. Kenttäväyläverkon automaattinen konfigurointi 1998. 68 s. + liitt. 3 s.
- 1933 Ihme, Tuomas, Kumara, Pekka, Suihkonen, Keijo, Holsti, Niklas & Paakko, Matti. Developing application frameworks for mission-critical software. Using space applications as an example. 1998. 92 p. + app. 20 p.
- 1965 Niemelä, Eila. Elektroniikkatuotannon joustavan ohjauksen tietotekninen infrastruktuuri. 1999. 42 s.
- 1985 Rauhala, Tapani. Javan luokkakirjasto testitapauseditorin toteutuksessa. 1999. 68 s.
- 2042 Kääriäinen, Jukka, Savolainen, Pekka, Taramaa, Jorma & Leppälä, Kari. Product Data Management (PDM). Design, exchange and integration viewpoints. 2000. 104 p.
- 2046 Savikko, Vesa-Pekka. EPOC-sovellusten rakentaminen. 2000. 56 s. + liitt. 36 s.
- 2065 Sihvonen, Markus. A user side framework for Composite Capability / Preference Profile negotiation. 2000. 54 p. + app. 4 p.
- 2088 Korva, Jari. Adaptiivisten verkkopalvelujen käyttöliittymät. 2001. 71 s. + liitt. 4 s.
- 2092 Kärki, Matti. Testing of object-oriented software. Utilisation of the UML in testing. 2001. 69 p. + app. 6 p.
- 2095 Seppänen, Veikko, Helander, Nina, Niemelä, Eila & Komi-Sirviö, Seija. Towards original software component manufacturing. 2001. 105 p.
- 2114 Sachinopoulou, Anna. Multidimensional Visualization. 2001. 37 p.
- 2129 Aihkisalo, Tommi. Remote maintenance and development of home automation applications. 2002. 85 p.
- 2130 Tikkanen, Aki. Jatkuva-aikaisten multim mediasovellusten kehitysalusta. 2002. 55 s.
- 2157 Pääkkönen, Pekka. Kodin verkotettujen laitteiden palveluiden hyödyntäminen. 2002. 69 s.

Tätä julkaisua myy
VTT TIETOPALVELU
PL 2000
02044 VTT
Puh. (09) 456 4404
Faksi (09) 456 4374

Denna publikation säljs av
VTT INFORMATIONSTJÄNST
PB 2000
02044 VTT
Tel. (09) 456 4404
Fax (09) 456 4374

This publication is available from
VTT INFORMATION SERVICE
P.O.Box 2000
FIN-02044 VTT, Finland
Phone internat. + 358 9 456 4404
Fax + 358 9 456 4374