



# Vuorovaikutteiset puhe-sovellukset

Markku Turunen & Jaakko Hakulinen

Tampereen yliopisto

Tietojenkäsittelytieteiden laitos

Tampere Unit for Computer-Human Interaction

Speech-based and Pervasive Interaction Group

{Markku.Turunen,Jaakko.Hakulinen}@cs.uta.fi

## Tiivistelmä

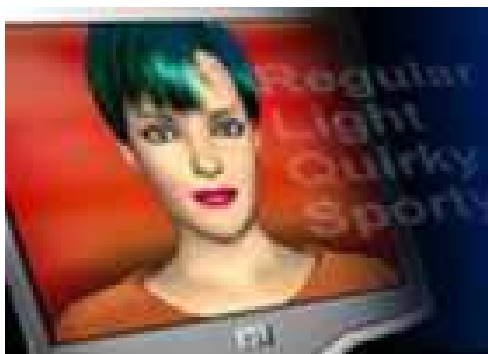
Puheteknologiaan perustuvat sovellukset nähdään usein seuraavana askeleena ihmisen ja teknologian välisessä vuorovaikutuksessa. Esittelemme tässä artikkelissa keskeisimmät puhe-sovelluksiin liittyvät seikat käyttöliittymien ja ohjelmistokehityksen näkökulmasta. Kehittämällä vuorovaikutuksen hallintaan sopivia menetelmiä meidän on mahdollista hyödyntää puheteknologiaa uusilla alueilla ja mahdollistaa tietokoneiden entistä helpompi ja tehokkaampi käyttö erilaisille käyttäjille. Havainnollistamme puhepohjaisen vuorovaikutuksen hallintaan ja sovelluskehitykseen liittyviä asioita konkreettisilla esimerkeillä tutkimusryhmämme tuottamista puhe-sovelluksista. Esittelemme lopuksi näkökulmia tulevaisuuden puhe-sovelluksiin ja niiden asettamiin tutkimushaasteisiin.

## 1 Johdanto

Puheteknologiaan eli puheentunnistukseen, puhesynteesiin ja puhujantunnistukseen pohjautuvia sovelluksia hyödynnetään enenevässä määrin. Synteettistä puhetta on käytetty jo pitkään erityisryhmien sovelluksissa. Erityisesti näkövammaisille synteetitknologia on tärkeä osa arkea, sillä se tarjoaa mahdollisuuden päästä käsi- mm. www-palveluihin ja muihin tietopalveluihin. Synteettinen puhe on tullut laajemmalti tutuksi television välityksellä ohjelmissa, joissa juontaja on korvattu automatisoidulla palvelulla. Yksinkertaisimmillaan synteesi lukee konemaisella äänellä käyttäjien lähettämiä tekstiviestejä, kun taas toisessa ääripäässä yhdistetään

luonnollisen kuuloista puhetta ja näyttävää fotorealistista grafiikkaa nk. puhuvien päiden muodossa. Tunnetuin näistä sovelluksista lienee kuvan 1 maailmanlaajuista huomiota herättänyt naispuolinen uutisankkuri **Ananova**. Helsingin Sanomat innostui uutisoimaan asiasta kesällä 2000 otsikolla *"Syrjäyttääkö Ananova Arvi Lindin?"* (nykytiedon valossa Lindin urakka- resta muodostui mittavampi, sillä molemmat ovat eläkeläisiä).

Puheentunnistuksen sovellukset ovat synteetin tavoin usein erityisalueilla ja erityisryhmien käytössä. Puheentunnistusta popularisoivien kirjoitusten kesto-suosikki on viitata Kubrickin ja Clarcken visioon puhetta täydellisesti ymmärtävästä inhimillisestä HAL-9000 tietokonees-



Kuva 1: Virtuaalinen uutisankkuri Anonova (kuva lähteestä <http://www.ananova.com/>).

ta. Oman sukupolvellemme puheentunnitukseen perehdytti ihmeauto KITT. Valittavasti puheentunnistus, ei ole täydellistä, vaan pikemmin rajoittunutta ja virhealtista. Oikein hyödynnettyä se kuitenkin mahdollistaa suuren joukon käytännön sovelluksia. Keskeistä tällöin on vuorovaikutuksen suunnittelu. Päämääränä on käyttäjille luonteva, helppokäyttöinen käyttöliittymä, joka ottaa huomioon puheteknologian mahdollisuudet ja rajoitukset.

Esittelemme seuraavaksi keskeisimmät puheteknologiaan ja vuorovaikutukseen liittyvät seikat, jotka on otettava huomioon puheteknologiaa sovellettaessa. Tämän jälkeen esittelemme keskeisimmät vuorovaikutuksen hallintaan kehitetyt menetelmät. Havainnollistamme näitä konkreettisilla esimerkeillä tutkimusryhmämme tuottamista aikataulusovelluksista, sähköpostisovelluksista ja jatkokaikaa tietotekniikan sovelluksista. Tämän jälkeen tarkastelemme puhe-sovellusten toteuttamista, ja luomme lyhyen katsauksen sovellustemme pohjana olevaan yleiseen sovellusarkkitehtuuriin. Lopuksi esittelemme vision tulevaisuuden puhe-sovelluksista.

## 2 Puhepohjainen vuorovaikutus

Puhe on ihmisten välisessä vuorovaikutuksessa tehokkain ja suosituin menetelmä ja puheteknologian sanotaan usein tekevän ihmisen ja tietokoneen välisestä vuorovaikutuksesta luonnollisempaa. Puheella onkin useita ominaisuuksia, jotka potentiaalisesti tekevät tietotekniikan hyödyntämisestä entistä luontevampaa ja tuottavampaa. Puheteknologialla on myös useita rajoituksia, jotka tulee ottaa huomioon sovelluksia suunniteltaessa.

### 2.1 Puheen keskeiset ominaisuudet

Puheen keskeisenä ominaisuutena pidetään *luonnollisuutta*. Motiivina puheen tuomiseen osaksi ihmisen ja tietokoneen välistä vuorovaikutusta on ihmisten välisen vuorovaikutuksen etujen hyödyntäminen. Tyypillisessä dialogijärjestelmässä luonnollisuuteen pyritään käyttämällä vuorovaikutusmetaforana ihmisten välistä keskustelua samalla aihealueella. Esimerkiksi paikallisliikenteen aikataulujärjestelmässä voidaan mallintaa matkustajan ja asiakaspalvelun välistä vuoropuhe-

lua. Luonnollisuus on kuitenkin kaksite-räinen miekka. Ensinnäkin monet luonnollisena pitämämme asiat, kuten pääl-lekkään puhuminen, aiheuttavat merkittä-viä ongelmia puheentunnistukselle. Toi-sekseen on todettu, että ihmisen ja tietoko-noon välinen vuorovaikutus eroaa merkit-tävästi ihmisten välisestä vuorovaikutuk-sesta. Esimerkkinä aikataulujärjestelmälle soittavat ihmiset ovat harvoin konetta koh-taan niin kohteliaita, että lopettaisivat pu-helun muutoin kuin yksinkertaisesti sulke-malla luurin. Käytetty kieli voi myös olla merkittävästi erilaista koneelle puhuttaes-sa, kuten olemme huomanneet aikataulu-järjestelmien kohdalla [25].

Puhe on parhaimmillaan erittäin *te-hokas* vuorovaikutusmenetelmä, ja se on-kin todettu tehokkaimmaksi menetelmäk-si ihmisten välisessä ongelmanratkaisus-sa [2]. Tehokkuuteen liittyy monia tekijöitä, kuten *nopeus* ja *ilmaisuvoimaisuus*. Puheen avulla on mahdollista ilmaista asioita, joiden ilmaiseminen muilla tavoin on erittäin vaikeata. Puhepohjaisen vuorovaikutuksen tehokkuus tulee hyvin esille myös siirryttäessä pois perinteisistä graafisten käyttöliittymien hallitsemis-ta työpöytäympäristöistä. Graafisessa suorakäyttöliittymässä kuvakkeet ovat koko ajan nähtävillä ja muokattavissa mikä aiheuttaa haasteita esim. kannettavien lait-teiden pienillä näytöillä.

Puheella nähdään olevan tuottavuut-ta ja myös turvallisuutta lisäävä vaikutus etenkin useita samanaikaisia tehtäviä sisäلتävissä tilanteissa. Muut tehtävät voivat liittyä esimerkiksi koneiden ohjaukseen tai autolla ajamiseen, jolloin käyttäjän kä-det ja silmät ovat käytössä näissä tehtä-vissä. Kyseessä voi olla myös moniais-tinen (*multimodaalinen*) käyttöliittymä, jossa eri aisteja käytetään eri tarkoituk-siin. Yleisesti moniaistisuuden hyödyntä-minen mahdollistaa monipuoliset vuoro-

vaikutusratkaisut, ja puhe onkin usein kes-keisessä osassa näissä sovelluksissa.

Vaikka puhe ei olisikaan tehokkain käytettävissä oleva vuorovaikutusmene-telmä, se voi olla käyttäjien kannalta *suo-situin*. Osaksi tämä voi johtua ihmisten erilaisista ongelmanratkaisutavoista, sillä osa ihmisistä ajattelee pikemminkin pu-heen kuin kuvien avulla [1]. Myös van-huksille puhe voi olla luontevampi käyt-tötapa kuin perinteinen tietokoneen graa-finen suorakäyttö. Näissä tilanteissa pu-he voisi parantaa käyttäjien tyytyväisyyt-tä tietotekniikkaa kohtaan ja laajentaa sen käyttäjäjoukkoa.

## 2.2 Puheen rajoituksista

Edellä esitettyjen puheelle suotuisien ominaisuuksien lisäksi puheen käyttämi-seen liittyy joukko rajoituksia, joista osa liittyy teknologiaan, osa puheen luontee-seen sinänsä, ja osa ihmisten välisessä vuorovaikutuksesta opittuihin käytäntöi-hin. Näiden seikkojen tunteminen on kes-keistä käytännöllisiä puhepohjaisia järjes-telmiä suunniteltaessa.

Yhdeksi suurimmista ongelmista käyt-ännön puhe-sovelluksissa on koettu vuoro-vaikutuksessa tarvittavien kommentojen muistamisen tarve, tai yleisemmin vuoro-vaikutuksessa käytettävän kielen tunte-misen tarve. Lähitulevaisuudessa teknolo-gia ei mahdollista rajoittamatonta puheen tunnistamista eikä ymmärtämistä, minkä vuoksi käyttäjän ja järjestelmän välinen keskustelu on luonnollista puhetta rajoit-tuneempaa. Lisäksi on huomattava, että vaikka teknologia mahdollistaisikin hy-vin vapaamuotoisen puhutun kielen käyt-tämisen, täytyy käyttäjän tuntea järjestel-män mahdollisuudet ja rajoitukset. Edellä mainituista syistä johtuen puhe-käyttöliit-tymien keskeinen kysymys on, kuinka hy-vin järjestelmä tukee käyttäjän tietoisuutta siitä, mitä järjestelmälle voi sanoa.

Puhetulosteisiin liittyy keskeisesti puheen väliaikainen ja peräkkäinen luonne, joka ilmenee esimerkiksi silloin, kun halutaan tarkastella edeltäviä tapahtumia, kuten aikataulujärjestelmän antamia busien lähtöaikoja. Järjestelmän on pystyttävä tukemaan muistamista esimerkiksi keräämällä jo sanottuja asioita, mutta kuitenkin niin, ettei puheen hitaudesta muodostu lisäongelmia. Hienovarainen palautteen antaminen onkin yksi puhekäyttöliittymien suunnittelun keskeisiä kulmakiviä.

Vaikka puhesynteesi sinänsä on vaikiintunutta ja luotettavaa teknologiaa, puhetulosteisiin liittyy monia ongelmia. Yleistäen voidaan sanoa, että synteettinen puhe kuulostaa hyvin ymmärrettävältä, mutta ei kovinkaan luonnolliselta. Synteettisen puheen kuuntelun on myös todettu lisäävän kognitiivista kuormitusta, mikä voi olla keskeistä tarkkaavaisuutta vaativissa tehtävissä, kuten autolla ajettaessa [9]. Näistä syistä johtuen nauhoitetun puheen käyttäminen puhesynteesin sijaan onkin suositeltavaa silloin, kun se on mahdollista. On kuitenkin sovellusalueita, kuten sähköpostien kuunteleminen, joissa ei ole mahdollista käyttää nauhoitettua puhetta sisällön vapaamuotoisuuden vuoksi. Toisin kuin puhesynteesi, puheentunnistus on virhealtis teknologia. Tarkastelemus siihen liittyviä rajoituksia myöhemmin kohdassa "virheiden hallinta".

Puheen luonteeseen ja teknologiaan liittyvien rajoitusten lisäksi puhepohjaiseen vuorovaikutukseen liittyy monia opittuja tapoja. Näitä ovat mm. ihmisten taipumus puhua päällekkäin sekä taukojen ja yleisesti prosodisten elementtien (esim. äänenkorkeuden ja puhenopeuden) käyttäminen keskustelun rakenteen ohjaamiseen. Näiden mekanismien käyttäminen on ihmisten välisessä vuorovaikutuksessa pitkälle kehittyntä ja hienovaraisista, mutta niiden hyödyntäminen puhekäyttöliitty-

missä on vasta alussa. Voidaankin sanoa, että nämä aiheuttavat toistaiseksi enemmän ongelmia kuin tuovat hyötyä.

Merkittäväksi tekijäksi puhekäyttöliittymien yleistymiselle tulevaisuudessa saattaa muodostua puheen julkinen luonne, joka rajoittaa käyttömahdollisuuksia sekä syötteiden että tulosteiden kohdalla joillakin sovellusalueilla. Tämä tulee erityisesti esille silloin, kun käsitellään henkilökohtaista tietoa, kuten sähköposteja tai potilastietoja. Toistaiseksi myös (julkista) koneelle puhumista pidetään hyvin epäluontevana, ja siihen liittyy paljon enakkoluuloja. Odotettavissa kuitenkin on, puhekäyttöliittymien yleistyessä tämänkaltaiset opitut vuorovaikutusmekanismit muuttuvat ja kehittyvät käytön myötä samalla tavoin kuin on käynyt esimerkiksi tekstiviestien osalta.

Kuten edellä on kuvattu, puheen käyttämiseen liittyy paljon potentiaalia, mutta sillä on myös merkittäviä rajoituksia, erityisesti verrattuna ihmisten väliseen vuorovaikutukseen. Luonteva ja tehokas puhepohjainen ihmisen ja teknologian välinen vuorovaikutus vaatiikin osittain erilaisia menetelmiä ja lähestymistapoja kuin ihmisten välinen vuorovaikutus. Tutkimuksessa on syytä kiinnittää huomiota innovatiivisten ratkaisujen luomiseen ihmisten välisen vuorovaikutuksen jäljittelyn sijasta. Tämä korostuu erityisesti silloin, kun siirrytään pois perinteisistä tehtäväpohjaisista järjestelmistä kaikkialla läsnä oleviin, ympäristöön sulautuviin ja käyttäjien mukana kulkeviin sovelluksiin.

### 3 Vuorovaikutuksen hallinta

Puhesovellusten vuorovaikutukseen hallintaan on kehitetty ratkaisuja eri näkökulmista eri tieteenaloilla. Tutkimusaluee-

na puhekäyttöliittymien tutkimus on hyvin monitieteinen, ja siinä yhdistyy signaalinkäsittely, lingvistiikka, dialogijärjestelmien tutkimus, ohjelmistotekniikka ja vuorovaikutteinen teknologia. Näkökulmasta riippumatta keskeisiä seikkoja ovat käyttöliittymätasolla aloitteellisuuden hallinta, virheiden hallinta ja puhetulostoiden suunnittelu. Sovellustasolla keskeistä on vuoropuhelun (dialogin) mallintaminen.

### 3.1 Aloitteellisuuden hallinta

Useimmiten puhepohjainen vuorovaikutus nähdään dialogina käyttäjän ja järjestelmän välillä (käytämme jatkossa termiä *dialogi* tarkoittamaan käyttäjän ja järjestelmän välistä vuorovaikutusta puhejärjestelmissä). Keskeiseksi nousee tällöin se, kuinka dialogia mallinnetaan ja hallitaan. Käyttöliittymän kannalta tärkein piirre on aloitteellisuuden luonne, eli eteneekö dialogi järjestelmän ohjaamana, käyttäjän kysymysten ja kommentojen pohjalta vai näiden molempien lähestymistapojen yhdistelmänä.

Järjestelmän aloitteellisuuden pohjautuvassa dialogissa järjestelmä kysyy joukon kysymyksiä, joiden pohjalta se pystyy tuottamaan vastauksen käsillä olevaan tehtävään. Järjestelmän kannalta dialogin kulku on helposti ennustettavissa, ja puheentunnistus voidaan optimoida kuhunkin tilanteeseen kun tiedetään, mitä käyttäjä on sanomassa. Tämä parantaa tunnistustarkkuutta. Käyttäjän kannalta järjestelmän aloitteellisuus luo varmuuden siitä, että kaikki tarvittava tieto tulee kysytyä, etenkin jos järjestelmä antaa selkeitä ohjeita ja vaihtoehtoja siitä, mitä käyttäjän tulee sanoa seuraavaksi.

Järjestelmän aloitteellisuuteen pohjautuvat dialogit soveltuvat erityisen hyvin aloitteleville käyttäjille, jotka eivät tunne järjestelmää tai palvelun luonnetta

ennalta. Ne vastaavat myös hyvin luonnollisia vuoropuheluja esimerkiksi puhelintilauksissa ja erilaisissa tietopalveluissa. Näille alueille sijoittuukin suurin osa nykyisistä laajassa käytössä olevista sovelluksista.

Järjestelmän aloitteellisuuteen pohjautuvia puhejärjestelmiä kohtaan on esitetty paljon kritiikkiä. Keskeisenä ongelmana nähdään niiden kömpelyys kokeneille käyttäjille. Erityisesti, jos järjestelmä ottaa vastaan ainoastaan yhden syöteen kerrallaan, eikä järjestelmän puheenvuoroa pysty keskeyttämään, voi vuorovaikutuksesta muodostua turhauttavaa, kun saman tiedon voisi antaa yhdellä puhunnoksella kuuntelematta järjestelmän kysymyksiä. Tässä tulee esille yksi puhe-sovelluksien keskeisistä haasteista: aloittelijoille ja kokeneille käyttäjille parhaiten soveltuvat ratkaisut ovat usein tyystin erilaisia, minkä vuoksi puhekäyttöliittymän tulisi sopeutua käyttäjänsä. Tämä on keskeinen tekijä tutkimusryhmämme työssä, ja jäljempänä esitellään useita ratkaisuja tähän haasteeseen eri sovellusten alueella.

On myös aihealueita, joihin järjestelmän aloitteellisuus ei ole luonteva ratkaisu. Sähköpostien lukeminen on yksi tällainen alue. Sähköpostisovelluksissa vuorovaikutukseen vaikuttaa aina suuresti kulloinenkin käyttötilanne ja erityisesti sähköpostilaatikon sisältö. Periaatteessa käyttäjä voi haluta tehdä missä tahansa tilanteessa mitä tahansa, minkä vuoksi järjestelmän on vaikea ottaa aloitetta ja ohjata käyttäjää mihinkään tiettyyn suuntaan. Tämänkaltaisiin hyvin avoimiin tehtäviin sopii paremmin käyttäjän aloitteeseen pohjautuva dialogi.

Käyttäjän aloitteeseen pohjautuvat järjestelmät muistuttavat monilta osin komentorivikäyttöliittymiä. Järjestelmä suorittaa tällöin käyttäjän antamia kommentoja tai etsii vastauksen käyttäjän antamaan

kysymykseen. Käyttäjän kieli voi olla hyvin yksinkertaista ("*poista viesti*") tai hyvinkin monimuotoista ("*Öö, tuota, mun pitäis päästä hallille, siis niinku jäähallille tietty, ootas, Jaska hei, milloinkas se peli olikaan?, joo, siellä pitäis siis olla puol kolmen maissa huomisen jälkeen*"). Käyttäjän kannalta näiden järjestelmien viehätävyyden on niiden periaatteissa mahdollistamassa luonnollisessa vuorovaikutuksessa, jolloin käyttäjä voi antaa kaiken tiedon haluamallaan tavalla normaalia puhuttua kieltä käyttämällä. Tämä ei kuitenkaan ole puheentunnistuksen, luonnollisen kielen ymmärtämisen ja dialogin hallinnan näkökulmasta realistista sen paremmin nyt kuin lähitulevaisuudessa. Tämän vuoksi käyttäjän aloitteeseen pohjautuviin järjestelmiin liittyy paljon haasteita ja erityisesti monimutkaista kieltä ymmärättäviä palveluita on vähän todellisessa käytössä.

Lähes kaikki nykyinen tutkimus puhesovellusten alueella pyrkii yhdistämään järjestelmän ja käyttäjän aloitteeseen pohjautuvien dialogin hallintamenetelmien parhaita piirteitä, sillä sen paremmin käyttäjän kuin järjestelmän aloitteeseen pohjautuvat dialogit eivät sovi kaikkiin tilanteisiin ja kaikille käyttäjäryhmille. Tällöin puhutaan joustavasta aloitteellisuudesta jolloin käyttäjä tai järjestelmä voi olla aloitteellinen tilanteen vaatimuksesta riippuen. Tyypillinen esimerkki on virhetilanteen selvittäminen, jolloin muutoin käyttäjän aloitteeseen pohjautuva dialogin luonne muuttuu järjestelmän ottaessa aloitteen. Käytännössä virheiden hallinnan yhteydessä tapahtuvasta aloitteen vaihdosta onkin muodostunut synonyymi joustavalle dialogin hallinnalle, vaikka tarvetta olisi kehittyneemmille menetelmille esim. käyttäjän taito- ja tietotason mukaan tapatuvalle sopeuttamiselle. Näitä asioita tarkastellaan kohdassa 4 tarkemmin.

Teknisesti joustavaan aloitteeseen pohjautuvat dialogit ovat kaikkein haastavimpia toteuttaa, erityisesti mikäli käyttäjän sallitaan ottaa aloite milloin tahansa. Tämä asettaa erityisiä haasteita dialogin hallinnassa käytettäville menetelmille, sillä dialogien ennustettavuus on vähäistä. Lisäksi käytettävän kielen mallinnus muodostuu haastavaksi, sillä järjestelmän tulee ymmärtää sekä järjestelmän että käyttäjän aloitteeseen pohjautuvaa puhetyyliä. Tämä aiheuttaa monissa järjestelmissä liian paljon puheentunnistuksen virheitä. Kääntäen tunnistustarkkuutta voidaan tarvittaessa parantaa ottamalla aloite käyttäjältä järjestelmän hallintaan.

### 3.2 Virheiden hallinta

Virheiden hallinta muodostaa tärkeän osan puhesovellusten suunnittelua puheentunnistuksen virhealttiuden vuoksi. Tässä mielessä teknologian voidaankin sanoa hallitsevan sovellusten suunnittelua ja tehdyt ratkaisut ovat usein pikemmin teknologian sanelemia kuin parhaisiin käytäntöihin perustuvia [10]. Toisaalta, onnistuneella käyttöliittymäsuunnittelulla voidaan korvata puheentunnistuksen puutteita ja tuottaa toimivia käytännön sovelluksia [8]. Kaikki puheentunnistuksen virheet eivät ole samanarvoisia, ja tehtävän sekä käyttäjien vaatimuksilla on erittäin suuri merkitys. Tämän vuoksi teknologian arviointiin käytettävät menetelmät, kuten puheentunnistuksen tarkkuus, eivät välttämättä kerro paljoakaan teknologian soveltuvuudesta yksittäiseen tehtävään. Tämä tekee puhesovellusten testauksesta ja käytettävyyden arvioinnista erittäin haasteellista, ja alueella tarvitaan erityisesti subjektiivisia menetelmiä. Eräs mielenkiintoinen suuntaus on sovellusten palveluorientoinut arviointi [5].

Vuorovaikutuksen hallinnan näkökulmasta virheiden hallinta voidaan jakaa eri-

laisiin vaiheisiin, jotka kaikki tulisi ottaa huomioon dialogin onnistuneen jatkumisen turvaamiseksi. Tarkimmissa kuvauksissa on erotettu seitsemän eri vaihetta, jotka ovat *virheen havaitseminen, virheen syyn analysointi, virheen korjauksen suunnittelu, virheen korjaaminen, käyttäjälle tiedotus, paluu normaaliin dialogiin sekä virheiden ennaltaehkäisy* [19]. Kysymyksessä on siis erittäin monimuotoinen ilmiö, jonka hallinta tuottaa erityisiä haasteita dialogin kontrollointiin.

Pääosa virheiden hallinnasta keskittyy niihin menetelmiin, joilla virheitä korjataan. Yksinkertaisimmillaan käyttäjä pyydetään toistamaan (mahdollisella kehotuksella uudelleenmuotoilusta) puhuttu syöte, jolloin toivotaan puheentunnistuksen toimivan paremmin tällä kertaa esim. olosuhteisiin (ympäristön häiriöäänet), siirtokanavaan (puhelinlinjan häiriöt) tai käyttäjään (huolimaton lausunta) liittyvien tekijöiden muuttuessa. Ongelmana kuitenkin on, että virheet helposti kasaantuvat, jonka seurauksena joudutaan vuorovaikutukselle tuhoisaan virhekierteeseen. Käytännön nyrkkisääntönä voidaan pitää yhtä tai maksimissaan kahta uusintayritystä, jonka jälkeen useimmat käyttäjät turhautuvat ja luopuvat järjestelmän käytöstä. Virheitä täytyy tämän vuoksi pystyä hallitsemaan kehittyneemmällä menetelmällä.

Virheidenkorjaukseen kehitettyjä tekniikoita ovat mm. sopivan vaihtoehdon valinta puheentunnistimen palauttamista vaihtoehdoista tuloksista, tarkoittavat kysymykset, sekä erityistä huomiota ansaitsevat varmistukset. Varmistuksia on kahta tyyppiä, implisiittisiä ja eksplisiittisiä. Eksplisiittisessä varmistuksessa järjestelmä kysyy käyttäjältä, onko tunnistustulos oikea. Varmistus voi kohdistua joko koko syötteeseen (*"Haluatko lähtöajat Tampere-talolta Lielah-*

*teen?"*), tai sen osaan (*"Onko lähtöpaikka Tampere-talo?"*). Implisiittisessä varmistuksessa järjestelmä tiedottaa käyttäjälle ymmärtämästään, ja jättää mahdollisen virheen korjaamisen käyttäjän aloitteellisuuden varaan. Kuten eksplisiittisessä varmistuksissakin, implisiittisesti voidaan varmistaa yksi (*"Hatanpäälle seis- ka kahden minuutin päästä."*) tai useampi asia kerralla (*"Pääkirjastolta keskussai- raalaan kolmonen viiden minuutin päästä..."*).

Eksplisiittisten varmistusten avulla dialogi voidaan pitää hallittavana, mutta ne voivat tehdä vuorovaikutuksesta erittäin kömpelöä, sillä käyttäjän on aina vastattava niihin. Tämän vuoksi implisiittisiä varmistuksia pidetään usein parempana vaihtoehtona, sillä ne eivät vaadi käyttäjältä toimenpiteitä, ellei virheidenkorjausta haluta käynnistää. Implisiittiset varmistukset kuitenkin asettavat lisähaasteita puheentunnistukselle koska järjestelmän on ymmärrettävä käyttäjän tekemä virheidenkorjaus. Ne myös tekevät puhetulosteista helposti pitkiä ja kömpelöitä. Implisiittiset varmistukset asettavatkin suuren haasteen puhetulosteiden suunnittelulle.

### 3.3 Puhetulosteiden suunnittelu

Puheen piirteistä, erityisesti sen väliaikaisesta ja peräkkäisestä luonteesta johtuen puhetulosteiden suunnittelu on keskeisellä sijalla käytännön puhesovelluksissa. On tärkeää huomata, että puhetulosteet eivät ole erillinen osa sovelluksia, vaan liittyvät oleellisesti vuorovaikutuksen hallintaan. Esimerkiksi dialogin hallinnan tulee varautua implisiittisten varmistusten jälkeiseen käyttäjän aloitteellisuuteen. Yleisesti ottaen puhetulosteilla voidaan ohjata käyttäjää käyttämään sellaista kieltä, jota järjestelmä ymmärtää, ja jonka

kautta vuorovaikutuksesta muodostuu sujuvaa. Kääntäen tämä tarkoittaa sitä, että huonosti muotoillut järjestelmän puheenvuorot voivat johtaa käyttäjiä harhaan, ja tehdä jopa yksinkertaisista kyllä/ei kysymyksistä vaikeasti tulkittavia [6]. Puhutosteiden kannalta keskeistä on, että kaikki tilanteen kannalta olennainen tulee sanottua, mutta ei tämän enempää eikä vähempää [3].

### 3.4 Dialogin mallintaminen

Tarkasteltaessa vuorovaikutuksen hallintaa ohjelmistojen näkökulmasta keskeiseksi muodostuu dialogin hallinta ja siinä käytetyt mallit. Dialogin hallinnan tehtävänä on siirtää järjestelmä tilasta toiseen saadun syötteen perusteella. Syöte voi tulla käyttäjältä, tai se voi olla peräisin dialoginhallinnalta tai järjestelmästä itseltään. Tätä abstraktia käyttäytymistä voidaan mallintaa eri tavoin, kuten esimerkiksi tilakoneilla, lomakepohjaisilla malleilla tai erilaisilla ohjelmistoagenteilla.

Tilakoneet ovat erittäin suosittuja dialoginhallinnan malleja [11]. Solmut edustavat näissä malleissa dialogin tilaa (esim. järjestelmän puheenvuoroja), ja kaaret siirtymiä tilasta toiseen (esim. käyttäjän antamia syötteitä). Näin muodostunut verkko edustaa dialogin yleistä rakennetta, ja polut kaikkia mahdollisia dialogeja joita käyttäjä ja järjestelmä voivat käydä. Tilakoneet soveltuvat erityisen hyvin sellaisiin tehtäviin, joissa on selkeä tehtävä rakenne ja vähän siirtymiä tilojen välillä. Mikäli verkon koko kasvaa, ja erityisesti siirtymät tilojen välillä lisääntyvät, tilakoneista tulee vaikeita ylläpitää ja niiden selkeys hämärtyy. Erityisesti virheidenhallinta monimutkaistaa tilakoneita merkittävästi. Yleistäen voidaan sanoa, että tilakoneet soveltuvat parhaiten pienehköihin sovelluksiin, tai käytettäväksi osana muita dialoginhallintamalleja.

Tilakoneiden ohella suosituin dialoginhallintamallintamismenetelmä on lomapohjainen lähestymistapa [11]. Dialogi etenee, kunnes tarvittavat lomakkeen kentät saadaan täytettyä. Lomakemallin etuna on joustavuus: lomake voidaan periaatteessa täyttää missä tahansa järjestyksessä, käyttäjä voi antaa kullakin puheenvuorolla haluamansa määrän tietoa, ja aloitteellisuutta voidaan vaihdella joustavasti käyttäjän ja järjestelmän välillä. Tällainen vapaa joustava vuorovaikutus vaatii luonnollisesti paljon järjestelmän eri osilta, minkä vuoksi käytännön sovelluksissa on erilaisia rajoituksia.

Tyypillisiä lomakepohjaisia sovelluksia ovat tietopalvelut, kuten aikataulupalvelut. Tällöin lomakkeen kenttinä ovat tarvittavat tiedot, kuten lähtöaika, lähtöpaikka ja linja. Käyttäjä voi antaa kaiken tiedon kerralla (*"Milloin lähtee seuraava kakskutonen keskustorilta Multisiltaan?"*). Mikäli käyttäjä ei anna kaikkia tarvittavia tietoja, järjestelmä voi ottaa aloitteen ja esittää tarkentavia kysymyksiä (*"Mikä on lähtöpaikka?"*). Keskeisessä asemassa lomakepohjaisissa järjestelmissä on algoritmi, joka määrittelee kuinka dialogi etenee lomakkeen täyttämisen myötä. Tämä voi olla yleiskäyttöinen, kuten VoiceXML-selainten FIA-algoritmi (Form Interpretation Algorithm), jolloin sovelluskohtainen dialogi voidaan kuvata määrittelykielen avulla, kuten VoiceXML-dokumentteina [27].

Tilakoneiden ja lomakepohjaisten mallien lisäksi on kehitetty useita dialoginhallintamenetelmiä, jotka lähestyvät aihetta hyvinkin erilaisista näkökulmista. Paljon työtä on tehty mm. klassisen tekoälyn sääntöjärjestelmien ja tilastollisten menetelmien parissa [11]. Yhteistä monille menetelmille on lähestymistapa, jossa dialogi nähdään agenttien välisenä vuorovaikutuksena [11]. Joissakin tapauksissa,



kuten esimerkiksi seuraavassa kohdassa kuvattavissa tutkimusryhmämme kehittämissä puhe-sovelluksissa, agentit viittaavat myös ohjelmistotason komponentteihin. Yleisesti ottaen ohjelmistoagentteja käytetään erilaisten dialogimallien yhteydessä.

## 4 Puhe-sovellukset

Puheteknologiaa hyödyntävillä sovelluksilla on yllättävän pitkä historia, vaikka ne edelleen mielletään verrattain uudeksi teknologiaksi. Erityisesti suurilla kielialueilla, joissa puheentunnistus on kehittyneempää kuin pienillä kielialueilla, puhelinpohjaisia sovelluksia käytetään laajasti. Toisaalta esimerkiksi meillä puheentunnistusta hyödyntäviä sovelluksia ei vielä ole merkittävässä määrin laajassa käytössä. Perinteiset puhelinpohjaiset sovellukset tekevät kuitenkin meilläkin tuloaan, ja samaan aikaan kehitetään uusia sovelluksia erilaisille alueille.

Kaikkein perinteisimmät puhe-sovellukset ovat IVR-palveluita (Interactive Voice Response). Näissä palveluissa puheteknologiaa hyödynnetään itse asiassa melko vähän: vuorovaikutus perustuu puhelimen näppäinten käyttämiseen, nauhoitettuun puheeseen, ja joissakin järjestelmissä myös puhe-synteesiin. Motivaationa on usein ihmistyövoiman korvaaminen, ja joissakin tapauksissa myös aidosti uusien (tai ihmistyövoimalla kannattamattomien) palveluiden tarjoaminen. Tyypillisiä esimerkkejä ovat asiakaspalveluun tulevien puheluiden vastaanotto ja ohjaus (call steering) sekä operaattoreiden ylläpitämät puhelinvastaajat [16].

Puhelimen näppäinten käyttämiseen perustuvissa käyttöliittymissä on monia käytettävyysongelmia, ja niiden rajat tulevat nopeasti vastaan puheentunnistukseen verrattuna, vaikka ne saattavatkin olla eri-

tyisryhmien suosiossa [24]. Puheentunnistus voi lisätä sekä käyttäjien tyytyväisyyttä että kustannustehokkuutta puhelinpalveluissa [17]. Puheentunnista hyödyntävät puhelinpohjaiset tietopalvelut ovatkin seuraavaksi merkittävin puhe-sovellusten ryhmä. Tyypillisiä sovellusalueita ovat sääpalvelut, pankkipalvelut ja seuraavaksi tarkemmin esiteltävät aikataulupalvelut.

Aikataulujärjestelmät muodostavat hyvin suosittuun puhe-sovellusten alueeseen. Tähän on useita syitä. Aikataulupalvelut ovat käytännöllinen sovelluskohde, ja sopivat luonteensa vuoksi hyvin automatisoitaviksi. Samalla ne kuitenkin tarjoavat runsaasti haasteita vuorovaikutustutkimukselle, ja niistä saatuja tuloksia voidaan soveltaa muissa samanlaisissa palveluissa. Taulukko 1 havainnollistaa kolmea erilaista paikallisliikenteen aikataulupalvelua, joista Pysäkkimies on palvelut tamperelaisia joukkoliikenteen käyttäjiä vuoden 2003 lokakuusta alkaen. Muut järjestelmät, Interact ja Bussimies, ovat tutkimusprototyyppejä. Kaikkia järjestelmiä on testattu erilaisilla käytettävyyssuhteiden menetelmillä, ja niihin on tehty multimodaalisia laajennuksia. Tarkempi kuvaus järjestelmistä on luettavissa tutkimusartikkelista [25].

Taulukko 1 esittelee erilaisia lähestymistapoja vuorovaikutuksen mallintamiseen. Interact-järjestelmässä lähtökohtana on luonnollinen ihmisten välinen vuorovaikutus, ja dialogimallin taustana on aidossa olosuhteissa kerätty dialogiaineisto. Bussimies-järjestelmässä aloitteellisuuden joustavaan hallintaan ja virhetilanteiden hoitamiseen on kiinnitetty erityistä huomiota. Pysäkkimies puolestaan pyrkii virhesietoiseen dialogiin pitämällä aloitteen itsellään, kunnes keskeinen tietosisältö on tarjottu käyttäjälle. Tämän jälkeen aloite luovutetaan käyttäjälle mah-

Taulukko 1: Aikataulujärjestelmien esimerkkidialogeja.

	Interact:	Bussimies:	Pysäkkimies:
J <sub>1</sub>	Tervetuloa Helsingin automaattiseen bussi-aikataulupalveluun. Kuinka voin auttaa?	Tervetuloa toivottaa Bussimies, Tampereen bussi-aikataulujärjestelmä! Kuinka voin auttaa?	Tervetuloa Pysäkkimies-järjestelmään. Ohjeita saat sanomalla kerro ohjeet. Kuultuasi äänimerkin, kerro pysäkin nimi, esimerkiksi Keskustori.
K <sub>1</sub>	Koska lähtee seuraava bussi Hakaniemestä Hietalakeuhon?	Olen lähdössä keskustorilta ja menossa Hervantaan.	Aleksanterin kirkko.
J <sub>2</sub>	10:59.	Anteeksi en ymmärtänyt. Kerro mistä olet lähdössä.	Haataanko pysäkin Aleksanterin kirkko aikataulu?
K <sub>2</sub>	Koska lähtee ensimmäinen sellainen bussi keskustasta Murikkiväkeen joka lähtee myöhemmin kuin 10:58?	Keskustorilta.	Kyllä.
J <sub>3</sub>	Valitettavasti bussi 44 ei mene keskustasta Murikkiväkeen.	Mirne olet menossa?	Kello on nyt 10:10. Tänään seuraavat vuorot pysäkillä Aleksanterin kirkko ovat: Vihmaislinja viisi, nyt heli (listaus jatkuu)... Anna kommentosi, esimerkiksi kerro ohjeet.
K <sub>3</sub>	Koska lähtee seuraava - mitkä bussilinjat kulkevat keskustasta Murikkiväkeen?	Hervantaan.	Valitse linja.
J <sub>4</sub>	Anteeksi en ymmärtänyt.	Keskustorilta Hervantaan menevät linjat 13 ja 20. Haluatko kuulla niiden aikataulut?	Kerro linjan numero tai sano: "Listaa linjat".
K <sub>4</sub>	Mitkä bussit kulkevat keskustasta Murikkiväkeen?	Kyllä.	Kolmelta.
J <sub>5</sub>	10:57.	Keskustorilta Hervantaan arkisin klo 11:15 linja 13 (listaus jatkuu)...	Seuraavat linjan 13 vuorot pysäkillä Aleksanterin kirkko ovat Hermiaan minuutin päästä, Ilkurin (listaus jatkuu)...

dollisia lisätoimintoja varten. Kuten esimerkkidialogit osoittavat, virheidenhallinta ja palaute ovat keskeisessä asemassa paikattaessa puheentunnistuksen puutteita. Kokemustemme perusteella toimivien ratkaisujen suuren yleisön käyttöön tarkoitettavissa järjestelmissä on aloitteen pitämisen järjestelmällä. Tähän samaan johtopäätökseen ovat tulleet muutkin tutkijaryhmät [14].

Aikataulujärjestelmien ohella toinen suosittu puhe-sovellusten ryhmä on sähköpostipalvelut. Sähköposti itsessään on eräs tietojenkäsittelyn merkittävimpiä sovelluksia, ja puhekäyttöliittymä siihen tarjoaa houkuttelevan mahdollisuuden päästä käsiksi sähköposteihin milloin ja missä tahansa, esimerkiksi autolla ajaessa. Tutkimusryhmämme ensimmäinen sähköpostisovellus on vuodelta 1998, ja viimeisin versio kehitettiin osana monikansallista EU-hanketta [22]. Sähköposti on erittäin haasteellinen tutkimukselle mm. monikielisuuden ja sähköpostien vapaamuotoisuuden vuoksi, kuten oheisesta esimerkkidialogista käy ilmi:

*J: Tervetuloa käyttämään Athosmailia. Ei uusia viestejä. 17 vanhaa viestiä. Sinulla on 5 ryhmää. Ensimmäisessä ryhmässä viestit otsikolla "Reply: about deliverables". Toisessa ryhmässä... (listaus jatkuu).*

*K: "Valitse kolmas ryhmä"*

*J: Kolmas ryhmä, viestit lähettäjältä Kristiina Jokinen. Ensimmäisen viestin otsikko on "D6.1". Toisen... (listaus jatkuu).*

*K: "Listaa lähettäjät"*

*J: Viisi lähettäjä. Listaan lähettäjät. Adam Funk, Markku Turunen, Kristiina Jokinen, Kari Kanto, Maria Cheadle ja Veikko Surakka.*

*K: "Mitä viestejä minulle on Adam Funkilta?"*

*J: Toisessa ryhmässä on viisi viestiä lähettäjältä Adam Funk. Ensimmäisen otsikko on... (listaus jatkuu).*

*K: "Haluan lisätietoja ensimmäisestä viestistä"*

*J: Message number one from Adam Funk, subject is "AthosMail review, urgent". Message was sent on Wednesday at 14.24. It has 6 lines. Summary of message: "At-*

*hosMail review, urgent. Hi, I got this message from you."*

*K: "Lue viesti"*

*J: Message with subject "AthosMail review, urgent" is sent by Adam Funk. Lainaus edellisestä viestistä: "Onko kaikki siellä päässä jo valmiina?". Lainaus päättyy. "Hi, I got this message from you. Could you please translate what it means?". "Our components are finished. I made a draft of changes. You can find it at <http://www.umist.edu.uk/AthosMail>". Cheers, Adam. The message has been read.*

*K: "Poista se"*

*J: Poistetaanko viesti otsikolla 'AthosMail review, urgent', jonka on lähettänyt Adam Funk. Sano kyllä tai ei.*

*K: "Kyllä"*

*J: Viesti on merkitty poistettavaksi. Ei uusia viestejä. 17 vanhaa viestiä.*

*K: "Kiitti hei!"*

*J: Heippa, soitellaan.*

Pienten kielialueiden kannalta monikielisyys on keskeinen osa monia puhe-sovelluksia. Tietopalveluissa, kuten aikatauluissa, monikielinen sisältö palvelee eri käyttäjäryhmiä, kun taas luonteeltaan aidosti monikielisyyssovelluksissa, kuten sähköpostissa, tuki monikielisyydelle on käytännön välttämättömyys. Teknologia asettaa omat rajoituksensa: esimerkiksi puhesynteesin vaihtaminen monikielisyssä sähköpostissa ei kuulosta hyvältä lyhyissä jaksoissa, vaikka kieli pystyttäisiin tunnistamaan tarvittaessa aina sanatasolla riittävän luotettavasti. Samoin kielten erityispiirteet on otettava huomioon, esimerkiksi vieraskielisten nimien taipuminen ja lausuminen (vertaa "*Mitä viestejä minulle on Adam Funkilta?*" ja "*What messages do I have from Markku Turunen?*"). Usein nämä ilmiöt vaikuttavat koko järjestelmään, esimerkiksi nimien taipuminen sähköpostijärjestelmäs-

sä täytyy ottaa huomioon puheentunnistimen kielioppien ajonaikaisessa tuottamisessa, kielen tulkinnassa, dialoginhallinnassa ja vastausten tuottamisessa.

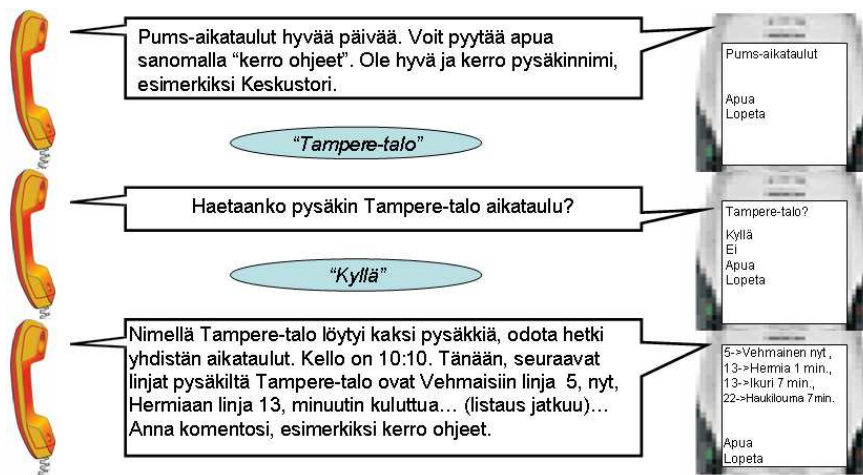
Pelkkään puheeseen perustuvien sovellusten lisäksi puhetta käytetään yhä enenevässä määrin multimodaalisissa järjestelmissä. Tällä pyritään yhdistämään eri syötteidenanto- ja tulostemenetelmien parhaat puolet. Lopputuloksena on parhaassa tapauksessa virhesietoinen järjestelmä, jossa käyttäjä voi valita joustavasti käyttämänsä modaliteetit. On kuitenkin syytä huomata, että multimodaalisuus sinänsä ei tee käyttöliittymästä parempaa, ja siihen liittyy paljon lunastamattomia lupauksia ja suoranaisia myyttejä [13].

Multimodaalisuus voi oikein käytettynä tehostaa vuorovaikutusta monilla tavoin. Esimerkki yksinkertaisesta multimodaalisesta puheikäyttöliittymästä on kuvassa 2 esiintyvä vaihtoehtoinen käyttöliittymä Pysäkkimies-sovellukseen.

Kuten kuvasta 2 näkyy, puhedialogissa voidaan hyödyntää matkapuhelimen näyttöä ja ohjainta dialogien tehostamiseksi ja vaihtoehtojen tarjoamiseksi. Tekniseltä kannalta kiinnostavaa on, että tämä voidaan toteuttaa täsmälleen saman dialogikuvauksen perusteella hyödyntämällä agenttitekniologiaa ja hajauttamista [15].

Eräs kiehtova multimodaalisuuden muoto on ihmisenkaltaisten, tai sarjakuvamaisten hahmojen käyttäminen osana puheikäyttöliittymää. Yksinkertaisimmillaan hahmot voivat olla pelkkä tehostekaino, mutta niiden käyttämisellä on perusteltu sijansa mm. opetuksessa. Kuvassa 3 esiintyy multimodaalinen opastaja, joka neuvoa aloittelevaa käyttäjää Bussimiesjärjestelmän käytössä. Tällä on saatu lupaavia tuloksia uutta sovellusta opeteltaessa [4].

Multimodaalisten sovellusten myötä puheen käyttäminen on levinnyt perin-



Kuva 2: Multimodaalinen älypuhelinikäyttöliittymä.

teisistä puhelin- ja työpöytäsovelluksista jokapaikan tietotekniikan sovelluksiin ja mobiilikäyttöön. Puhe soveltuu erityisen hyvin pieninäyttöisten ja näytöttömien laitteiden, kuten matkapuhelinten ja kämmenitietokoneiden käyttöliittymiin. Uutena käyttökohteena on ympäristöön sijoitettu teknologia. Samalla nämä ympäristöt asettavat entistä suurempia vaatimuksia puheteknologialle ja edellyttävät erityyppisiä vuorovaikutustekniikoita kuin perinteiset puhesovellukset. Kuva 4 havainnollistaa tutkimusryhmämme kehittämää toimistoympäristöön sijoitettua jokapaikan tietotekniikan sovellusta.

Kuvassa 4 havainnollistettu Ovimiesjärjestelmä kontrolloi sisäänpääsyä toimistotiloihin ja huolehtii vieraiden opastamisesta haluttuun paikkaan tai halutun henkilön luo. Järjestelmä hyödyntää puheentunnistuksen ja puhesynteesin lisäksi puhujantunnistusta ja erilaisia ympäristöön sijoitettuja antureita, kuten EMFi-teknologiaa [7]. Järjestelmä on sekä henkilökunnan että vieraiden käytössä, kuten seuraavista esimerkkidialogeista käy ilmi:

#### Ovimies vierailijoiden käytössä:

*K: (vierailija soittaa ovikelloa)*

*J: Olen Ovimikko. Sano etsimäsi henkilön tai paikan nimi äänimerkin jälkeen.*

*K: "Markku Turunen"*

*J: Tervetuloa. Lukko on nyt auki.*

*K: (Astuu sisään opastajahahmon eteen)*

*J: Etsimäsi henkilö Markku Turunen on huoneessa 432. Päästäksesi sinne käänny vasemmalle. Mene neuvotteluhuoneen viereen. Käänny oikealle. Mene sohvan viereen. Etsimäsi Markun huone on etuvasemmalla kahden metrin päässä.*

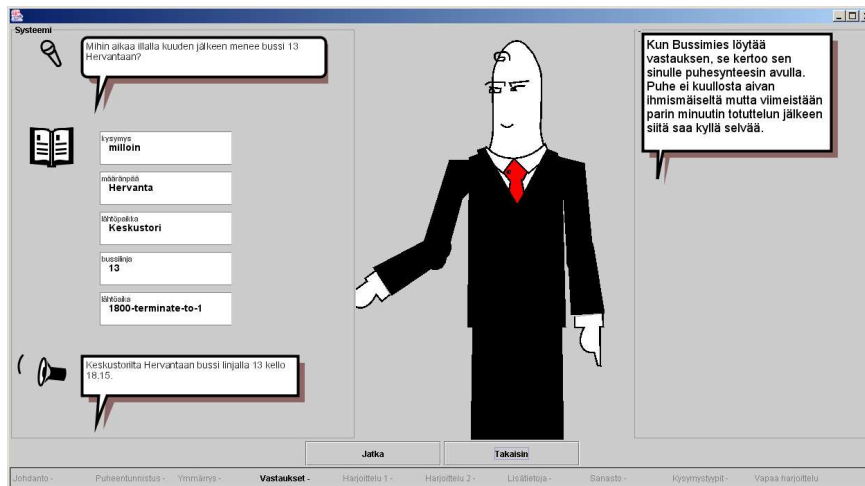
#### Ovimies henkilökunnan käytössä:

*J: (havaitsee liikkeen ja aloittaa tunnistuksen)*

*K: "Turusen Markku tässä huonenta."*

*J: Terve Markku. Lukko on auki.*

Jokapaikan tietotekniikan alueella on useita kiehtovia mahdollisuuksia joissa dialogit voivat olla huomattavasti monimuotoisempia kuin perinteisissä sovelluksissa. Näihin liittyvän vuorovaikutuksen mallintaminen vaatii uusia lähestymistapoja. Puheen ja graafisten elementtien lisäksi anturitieto ja ei-puheäänet,



Kuva 3: Multimodaalinen opastaja aikataulujärjestelmässä.

kuten esimerkiksi askeläänten käyttäminen työryhmän tietoisuuden lisäämiseen [12], avaavat uusia mahdollisuuksia puhepohjaiseen vuorovaikutukseen. Sovellusten kehittyessä myös eri palveluiden käyttö yhdessä on yhä tärkeämpää. Esittelemme tätä tarkemmin viimeisessä kohdassa tulevaisuuden puhesovellusten yhteydessä.

## 5 Puhesovellusten toteuttaminen

Useimmat puhesovellukset toteutetaan dialogijärjestelminä, jolloin niissä on muodossa tai toisessa kuvassa 5 esiintyvät komponentit puheentunnistukseen, luonnollisen kielen ymmärtämiseen, dialoginhallintaan, sovelluskohtaiseen tietoon, luonnollisen kielen generointiin ja puhesynteesiin.

Kuvan 5 esittämät ylätasoinen komponentit jakaantuvat usein pienempiin komponentteihin, jotka hoitavat erikoistuneita tehtäviä. Kaikkien tutkimusryhmämme

kehittämien sovellusten pohjana on yhteinen Jaspis-sovellusarkkitehtuuri [24]. Jaspis-arkkitehtuurin kehittämisessä ovat lähtökohtina modulaarisuus ja sopeutu- vuus erilaisiin vuorovaikutuskäytäntöihin, esimerkiksi erilaisten dialoginhallintamenetelmien ja monikielisyuden joustava hallinta ajonaikaisesti. Teknisesti arkkitehtuuri hyödyntää ohjelmistoagentteja ja järjestelmätason adaptiivisuutta, joka saavutetaan käyttämällä pieniä tilanteen mukaan valittavia komponentteja. Kuva 6 havainnollistaa Jaspis-pohjaisten aikataulusovellusten yleisarkkitehtuuria.

Kuten kuvasta 6 näkyy, Jaspis-pohjaiset sovellukset koostuvat agenteista, arvioijista ja managereista, jotka on koottu moduuleihin. Moduulien ja niiden sisältämien komponenttien lukumäärä ei ole rajoitettu, ja niillä on jaettu tietovarasto, johon kaikilla komponenteilla on pääsy. Esimerkiksi aikataulujärjestelmissä on viidestä kymmeneen moduulia ja joitakin kymmeniä agenteja järjestelmästä riippuen, kun taas sähköpostijärjestelmässä on liki kaksikymmentä moduulia ja yli sa-

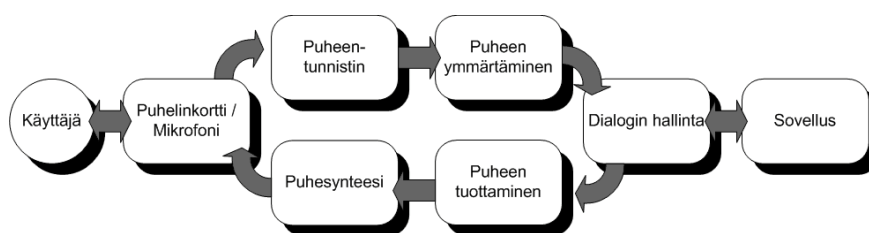


Kuva 4: Multimodaalinen opastusjärjestelmä.

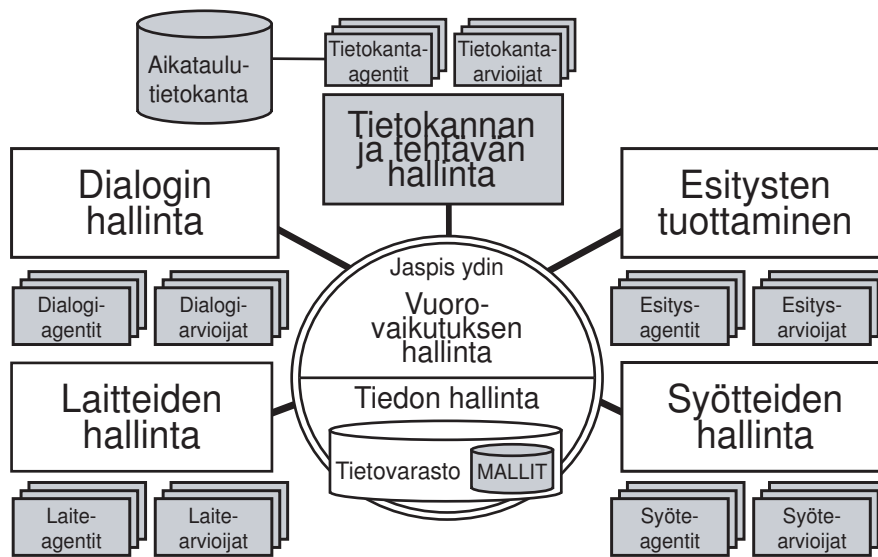
ta agenttia [23]. Alkuperäistä arkkitehtuuria [18] on myöhemmin laajennettu mm. jokapaikan tietotekniikan [20] sekä kannettavien laitteiden [26] tarpeisiin. Arkkitehtuurin täydellisempi esittely löytyy teoksesta [21] ja yleiskuvaus artikkelista [24].

## 6 Tulevaisuuden puhe-sovellukset

Puhe-sovellusten tutkimuksessa on meillä voimakas murroskausi perinteisten puhe-sovellusten tullessa laajamittaiseen käyttöön ja huomion siirtyessä uusille alueille kuten jokapaikan tietotekniikkaan. Näillä sovellusalueilla tarvitaan paitsi teknologisia edistysaskeleita (mm. ympäristön vaihteluille robusti puheen-



Kuva 5: Puhejärjestelmän osat.



Kuva 6: Aikataulujärjestelmien rakenne.

tunnistus), niin myös uusia menetelmiä vuorovaikutukseen, jossa dialogit voivat olla ajallisesti ja paikallisesti hajautettuja, ja palvelu useita yhtäaikaista käyttäjiä. Kuvassa 7 ja sitä seuraavassa esimerkidiialogissa on visio siitä, millainen voi olla vuorovaikutus tulevaisuuden puhesovellusten kanssa.

**Järjestelmä soittaa käyttäjän matkapuhelimeen kuvan 7 (vasemmalla) mukaisesti.**

*J: Kiirehdä, Esa Pekka! Sinulla on tapaaminen Leenan kanssa Vapriikkissa 20 minuutin päästä. Ota bussi 13 keskustorilta, se lähtee 5 minuutin päästä.*

*K: OK, kiitoksia.*

**Museolla järjestelmä toimii oppaana käyttäen vuorovaikutuksen tilaan sijoitettuja kaiuttimia ja mikrofoneja. Järjestelmä hyödyntää museon tietokantaa.**

*J: Näkemäsi haarniska on 1570-luvulta. Sen nahkakuviointi...*

*K: Onko täällä samantyyppisiä esineitä?*

*J: Kyllä, löydät samanlaisen haarniskan seuraavasta huoneesta. Pääset sinne...*

*Järjestelmä soittaa käyttäjän matkapuhelimeen yksityisen tiedon välittämiseksi.*

*J: Sinulla on tapaaminen lääkärisi kanssa. Ota bussi 27 keskussairaalalle. Se lähtee 20 minuutin päästä, joten ehdit katsoa näyttelyn loppuun.*

*K: Melkein unohdin tapaamisen, kiitoksia!*

**Järjestelmä kuuluttaa tuoreimman paikallisliikenteen häiriötiedotteen kaiuttimesta:**

*J: Olen pahoillani, bussi on myöhässä. Suosittelemme ottamaan seuraavan raitiovaunun, se lähtee 7 minuutin päästä ravintola "Höyryn" edestä.*

*K: Ei, otan mieluummin taksin.*

*J: Ok, etsin taksin, pieni hetki. (Hetken päästä järjestelmä jatkaa). Taksi on tulossa 2 minuutin sisällä, odota sitä museon edessä.*

*K: Kiitoksia!*



Kuva 7: Tulevaisuuden puhejärjestelmä.

Kuten esimerkistä käy ilmi, tulevaisuuden puhejärjestelmillä on useita haasteita edessä. Esimerkki ei kuitenkaan ole utopistinen, vaan monilta osin toteutettavissa nykyteknologialla ja sopivilla vuorovaikutustason ratkaisuilla. Keskeistä tulevaisuuden puhejärjestelmien kehittämisessä onkin käyttäjälähtöisten, innovatiivisten ratkaisujen kehittäminen ihmisen ja ympäröivän teknologian vuorovaikutuksen hallintaan.

## Viitteet

- [1] Bradford, J. The Human Factors of Speech-Based Interfaces: A Research Agenda. *ACM SIGCHI Bulletin*, 27(2): 61-67, 1995.
- [2] Chapanis, A. Interactive Human Communication. *Scientific American*, 232: 36-42, 1975.
- [3] Grice, P. Logic and conversation. In Cole, P. and Morgan, J.L. (editors): *Syntax and Semantics*, Vol. 3, Speech Acts, New York, Academic Press: 41-58, 1975.
- [4] Hakulinen, J., Turunen, M., Salonen, E.-P. Software Tutors for Dialogue Systems Proceedings of Text, Speech and Dialogue (TSD 2005), 2005: 412-419.
- [5] Hartikainen, M., Salonen, E.-P., Turunen, M. Subjective Evaluation of Spoken Dialogue Systems Using SERVQUAL Method. Proceedings of ICSLP 2004: 2273-2276, 2004.
- [6] Hockey, B., Rossen-Knill, D., Spejewski, B., Stone, M., Isard, S. Can You Predict Responses to Yes/No Questions? Yes, No, and Stuff. Proceedings of Eurospeech 1997: 2267-2270, 1997.
- [7] Kainulainen, A., Turunen, M., Hakulinen, J., Salonen, E.-P., Prusi, P., Helin, L. A Speech-based and Auditory Ubiquitous Office Environment. Proceedings of 10th International Conference on Speech and Computer (SPECOM 2005): 231-234, 2005.
- [8] Kamm, C. User Interfaces for Voice Applications. In *Voice Communication Between Humans and Machines*. Roe, D., Wilpon, J. (editors). National Academy Press, Washington D.C.: 422-442, 1994.
- [9] Lai J., Wood D., Considine M. The Effect of Task Conditions on the Comprehensibility of Synthetic Speech. Proceedings of Text, Speech and Dialogue (TSD 2005), 2005: 412-419.



- dings of CHI 2000, ACM Press: 321-328, 2000.
- [10] Mane, A., Boyce, S., Karis, D., Yankelevich, N. Designing the User Interface for Speech Recognition Applications. SIGCHI Bulletin, 1996, Volume 28 (4): 29-34, 1996.
- [11] McTear, M. Spoken Dialogue Technology: Enabling the Conversational Interface. ACM Computing Surveys, 34, 1, March 2002: 90-169, 2002.
- [12] Mäkelä, K., Hakulinen, J., Turunen, M. The Use Of Walking Sounds In Supporting Awareness. Proceedings of ICAD 2003: 144-147, 2003.
- [13] Oviatt, S. Ten Myths of Multimodal Interaction. Communications of the ACM, 42 (11): 74-81, 1999.
- [14] Raux, A., Langner, B., Bohus, D., Black, A. W. Eskenazi, M. Let's Go Public! Taking a Spoken Dialog System to the Real World. Proceedings of Interspeech 2005: 885-888, 2005.
- [15] Salonen, E.-P., Turunen, M., Hakulinen, J., Helin, L., Prusi, P., Kainulainen, A. Distributed Dialogue Management for Smart Terminal Devices. Proceedings of Interspeech 2005: 849-852, 2005.
- [16] Schmandt, C. Voice Communication with Computers. Van Nostrand Reinhold, New York, 1994.
- [17] Suhm, B., Bers, J., McCarthy, D., Freeman, B., Getty, D., Godfrey, K., Peterson, P. A comparative study of speech in the call center: natural language call routing vs. touch-tone menus. Proceedings of CHI 2002: 283-290, 2002.
- [18] Turunen, M., Hakulinen, J. Jaspis - A Framework for Multilingual Adaptive Speech Applications. Proceedings of ICSLP 2000: 719-722, 2000.
- [19] Turunen, M. Hakulinen, J. Agent-based Error Handling in Spoken Dialogue Systems. Proceedings of the Eurospeech 2001: 2189-2192, 2001.
- [20] Turunen, M., Hakulinen, J. Jaspis2 - An Architecture For Supporting Distributed Spoken Dialogues. Proceedings of the Eurospeech 2003: 1913-1916, 2003.
- [21] Turunen, M. Jaspis - A Spoken Dialogue Architecture and its Applications. Ph.D. Thesis, University of Tampere, 2004.
- [22] Turunen M., Salonen, E.-P., Hartikainen, M. et al. AthosMail - a Multilingual Adaptive Spoken Dialogue System for E-mail Domain. Proceedings of Workshop on Robust and Adaptive Information Processing for Mobile Speech Interfaces: 77-86, 2004.
- [23] Turunen, M., Salonen, E.-P., Hartikainen, M., Hakulinen, J. Robust and Adaptive Architecture for Multilingual Spoken Dialogue Systems. Proceedings of ICSLP 2004: 3081-3084, 2004.
- [24] Turunen, M., Hakulinen, J., Rähkä, K.-J., Salonen, E.-P., Kainulainen, A., Prusi, P. An architecture and applications for speech-based accessibility systems. IBM Systems Journal, Vol. 44, No 3: 485-504, 2005.
- [25] Turunen, M., Hakulinen, J., Salonen, E.-P., Kainulainen, A., Helin, L. Spoken and Multimodal Bus Timetable Systems: Design, Development and Evaluation. Proceedings of 10th International Conference on Speech and Computer (SPECOM 2005): 389-392, 2005.
- [26] Turunen, M., Salonen, E.-P., Hakulinen, J., Kanner, J., Kainulainen, A. Mobile Architecture for Distributed Multimodal Dialogues. Proceedings of ASIDE 2005.
- [27] VoiceXML Forum:  
<http://www.voicexml.org>.