

## Auteursrechterlijke overeenkomst

Opdat de Universiteit Hasselt uw eindverhandeling wereldwijd kan reproduceren, vertalen en distribueren is uw akkoord voor deze overeenkomst noodzakelijk. Gelieve de tijd te nemen om deze overeenkomst door te nemen, de gevraagde informatie in te vullen (en de overeenkomst te ondertekenen en af te geven).

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling met

Titel: Ubiquitous computing & haar toepassingen binnen transport en mobiliteit

Richting: 2de masterjaar in de toegepaste economische wetenschappen: handelsingenieur in de beleidsinformatica

Jaar: 2009

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Ik ga akkoord,

RAYMAEKERS, Beppe

Datum: 14.12.2009

# ***Ubiquitous computing & haar toepassingen binnen transport en mobiliteit***

**Beppe Raymaekers**

promotor :

Prof. dr. Tom BRIJS

# *Woord vooraf*

Deze scriptie vormt het sluitstuk van mijn opleiding tot Handelsingenieur in de Beleidsinformatica aan de Universiteit Hasselt. De totstandkoming ervan zou onmogelijk geweest zijn zonder de steun en assistentie van een aantal personen die ik via deze weg graag mijn dank wil betuigen.

Eerst en vooral zou ik mijn ouders willen bedanken voor deze kans en de morele en financiële steun tijdens mijn opleiding.

Verder heb ik deze masterpoef tot een goed einde kunnen brengen door de professionele kennis en inzichten van mijn promotor, Prof. dr. Tom Brijs.

Tenslotte wil ik ook mijn vriendin, vrienden, familie en kotgenoten bedanken, niet enkel voor hun onvoorwaardelijke steun, maar evenzeer voor de vele fijne tijden die we hebben beleefd tussen de studiemomenten door.

*Mei 2009,*

*Beppe Raymaekers*

# Samenvatting

*Ubiquitous computing* is een term verwijzend naar de alom aanwezigheid van computers in onze maatschappij, mede mogelijk gemaakt door de steeds verdergaande miniaturisatie van computertechnologieën. De mens wordt als het ware omsingeld door ontelbare, kleine en inter-verbonden computers, onzichtbaar ingebouwd in allerlei soorten van dagdagelijkse objecten en apparaten, die mensen assisteren in hun alledaagse leven. Zodoende worden deze objecten en apparaten voorzien van allerhande *processing*- en communicatiecapaciteiten, zodat deze kunnen handelen op een omgevingsgevoelige manier. *Ubiquitous computing* zal bovendien leiden tot een naadloze verbinding van de fysieke wereld met haar belichaming in informatie systemen. Verschillende nieuwe intelligente applicaties kunnen bovendien worden geïmplementeerd, steunend op het voorgaande. Echter, beperkingen omtrent deze toepassingen zullen eerder komen vanuit een economische, sociale en ethische hoek, dan vanuit de technologie zelf.

Naast haar impact op het bedrijfsleven stelt de *ubiquitous computing* visie expliciet voor de maatschappij grondig te veranderen door ze volledig te informatiseren. Ten gevolge hiervan zullen we sterk afhankelijk zijn van het betrouwbaar functioneren van de *ubiquitous* systemen rondom ons, wat zal resulteren in een aantal fundamentele sociale en ethische uitdagingen. Hiervan is privacy onbetwistbaar de meest prominente. Velen maken zich namelijk zorgen voor een potentieel misbruik ervan en een mogelijk gebrek aan controle erop. Hierdoor is er nood aan zowel wettelijke overwegingen, als nieuwe technologische benaderingen en sociale inspanningen om deze uitdagingen te trotseren. Andere sociale en ethische uitdagingen hebben onder andere betrekking op de sociale comptabiliteit en aanvaarding van *ubiquitous computing*.

*Ubiquitous computing* biedt ons bovendien evenzeer talrijke mogelijkheden op vlak van transport en mobiliteit, maar ter realisatie hiervan moeten we evenzeer het hoofd kunnen bieden aan allerhande uitdagingen en risico's, welke vooral betrekking hebben op het leren omgaan met bewegende objecten in het 4D universum van plaats en tijd. Voor de ontwikkeling van deze mobiliteitstoepassingen hebben we namelijk nood aan een

effectief *ubiquitous* systeem dat mobiliteitsdata aangaande bewegende objecten verzamelt, bewaart en verwerkt, teneinde de mobiele gebruiker te kunnen voorzien van allerhande relevante informatie en diensten, afhankelijk van diens geografische positie.

Aangaande de verzameling van mobiliteitsdata zien we dat nieuwe ontwikkelingen binnen mobiele communicatietechnologieën en *ubiquitous computing* ervoor zorgen dat we op een eenvoudigere manier toegang hebben tot een omvangrijker volume hiervan. Onze dagelijkse activiteiten en de manier waarop mensen leven en bewegen, laten namelijk digitale sporen na. Binnen deze scriptie bekijken we twee prominente technologieklassen gebruikt om mobiele gebruikers te lokaliseren. Deze positioneringstechnologieën vormen als het ware de mobiliteitsdatabronnen. Eerst en vooral kijken we naar de *cellular-based* technologieën, welke voortbouwen op het mobiele telecommunicatienetwerk. Als alternatief hiervoor bekijken we eveneens de satellietplaatsbepaling, welke is gebaseerd op de reistijden van elektromagnetische impulsen tussen een zender en een ontvanger. Daar deze data worden verwezenlijkt op een *ubiquitous* manier, beschikken ze bovendien over additionele karakteristieken en aspecten welke hun effect hebben in de verdere verwerking ervan.

Naast de individuele posities van de mobiele gebruikers zijn we dikwijls eveneens geïnteresseerd in hun beweging of hun trajecten. Deze trajecten gaan we reconstrueren uit een verzameling steekproefwaarnemingen aan de hand van een proces dat we interpolatie noemen. De accuraatheid van deze trajecten is bovendien afhankelijk van de locatieonzekerheden voortvloeiend uit dit interpolatieproces, evenals de *sampling rate* en de accuraatheid van de gebruikte positioneringstechniek. Aan de bekomen trajecten is zodoende een zekere inherente onzekerheid gekoppeld, welke we moeten managen binnen de reconstructie, representatie, opslag of het verdere gebruik ervan.

Vervolgens moet deze grote hoeveelheid aan *spatiotemporal* data aangaande bewegende objecten, ofwel snel worden verwerkt voor *real-time* doeleinden, ofwel onderworpen worden aan allerlei *knowledge discovery* algoritmes. Eender welke verwerking wordt vereist, het bewaren van deze data is essentieel. Hiervoor focussen we ons op een speciale vorm van *spatiotemporal* databases, *Moving Objects Databases* (MOD) genaamd, en maken we bovendien gebruik van datamodellen die beschrijven hoe deze data zijn gestructureerd. Betreffende deze opslag is het vervolgens belangrijk rekening te houden met enkele managementkwesties hieromtrent. Eerst en vooral kijken we naar

*spatiotemporal* indexeringsmethoden, welke worden gebruikt door *spatiotemporal* databases aangaande *queries* te optimaliseren door objecten in een 3D ruimte zodanig te rangschikken zodat ze efficiënt teruggevonden kunnen worden. Daarnaast bekijken we eveneens hoe we kunnen omgaan met database *queries* wanneer er sprake is van locatieonzekerheden.

*Knowledge discovery* vervolgens vindt vandaag de dag plaats in een des te meer kennisrijke en rekenkrachtige omgeving. Mobiliteitskennis, verborgen in de talrijk verzamelde en kennisrijke mobiliteitsdatarecords, biedt dan ook talloze mogelijkheden ter ondersteuning van allerhande *ubiquitous* transport- en mobiliteitstoepassingen. Vooraleer we ons echter richten op de *spatiotemporal knowledge discovery* aspecten en methoden hebben we nood aan een gedegen begrip van de aspecten en technieken betreffende haar *spatial* variant. Binnen het *knowledge discovery* proces beschouwen we eerst en vooral de *feature extraction* stap, welke we aanwenden om *spatial* of *spatiotemporal* karakteristieken die impliciet gecodeerd zijn in de objecten te extraheren, daar deze waardevolle informatie kunnen aanbrengen. Vervolgens spitsen we ons toe op het *minen* van deze *spatial* en *spatiotemporal* data. Hierbij is het belangrijk op te merken dat de mobiliteitscontext waarop we ons toespitsen allerhande extra uitdagingen met zich meebrengt, waardoor het essentieel is dat de algoritmes over de mogelijkheid beschikken deze te managen en te incorporeren. Kennis geëxtraheerd uit trajectdata zou ons vervolgens kunnen bijstaan in diverse taken betreffende alledaagse transport- en mobiliteitsfenomenen.

Als laatste gaan we aan de hand van een praktijkbenadering dieper in op verschillende aspecten, behandeld doorheen deze scriptie. Deze praktijkbenadering wordt vorm gegeven door middel van mijn zelfontwikkelde UbiTracker applicatie, welke is bestemd voor mobiele telefoons en bestaat uit drie primaire doeleinden. Eerst en vooral willen we namelijk aan de hand van deze *ubiquitous* applicatie onderzoeken wat de mogelijkheden zijn van een alternatief voor satellietplaatsbepaling aangaande mobiele telefoons, waarna we gebaseerd op deze positioneringmethode via de applicatie *location-based services* en *track and trace* mogelijkheden willen aanbieden aan de mobiele gebruiker. Op een abstract niveau kunnen we vervolgens stellen dat de applicatie bestaat uit twee opeenvolgende stappen. Eerst en vooral bepalen we de positie van de mobiele gebruiker, waarna we deze locatie-informatie verzenden naar onze eigen UbiTracker database, waar

we de bezochte locaties loggen voor de *track and trace* mogelijkheid, en als basis voor het bieden van informatie en diensten aan de mobiele gebruiker.

# Inhoudsopgave

<b>Hoofdstuk 1</b> <i>Probleemstelling</i> .....	- 1 -
1.1 Praktijkprobleem .....	- 1 -
1.2 Onderzoeksprobleem.....	- 3 -
1.2.1 Centrale Onderzoeksvraag.....	- 3 -
1.2.2 Deelvragen .....	- 3 -
1.2.3 Onderzoeksopzet.....	- 4 -
<b>Hoofdstuk 2</b> <i>Ubiquitous Computing</i> .....	- 5 -
2.1 Op weg naar Smart en Inter-Verbonden Objecten.....	- 5 -
2.2 Technologische Trends en Ontwikkelingen.....	- 7 -
2.3 Economische Impact .....	- 9 -
2.4 Sociale en Ethische Impact .....	- 11 -
2.4.1 Privacy in Ubiquitous Computing .....	- 12 -
2.4.2 Andere Sociale en Ethische Uitdagingen .....	- 15 -
2.5 Ubiquitous Systemen.....	- 17 -
2.6 Transport en Mobiliteit.....	- 18 -
2.6.1 LKW-MAUT .....	- 20 -
2.6.2 IM@GINE-IT .....	- 21 -
2.6.3 Be-Mobile .....	- 22 -
2.6.4 Elementen van Ubiquitous Mobiliteitstoepassingen.....	- 24 -
<b>Hoofdstuk 3</b> <i>Mobiliteitsdata</i> .....	- 25 -
3.1 Aspecten, Karakteristieken en Concepten.....	- 25 -
3.1.1 Ubiquitous Data.....	- 26 -
3.1.2 De Semantiek van Beweging.....	- 27 -
3.1.3 Trajecten.....	- 29 -
3.1.4 Ruimte.....	- 31 -
3.1.5 Tijd.....	- 32 -
3.1.6 Bewegende Objecten .....	- 33 -
3.2 Positiebepaling - Ubiquitous Mobiliteitsdatabronnen.....	- 33 -
3.2.1 Cellular-Based Positioneringtechnologieën .....	- 34 -
3.2.1.1 Cell Identity.....	- 36 -



3.2.1.2	Cell Identity en Timing Advance .....	- 37 -
3.2.1.3	Time of Arrival .....	- 38 -
3.2.1.4	Enhanced Observed Time Differences.....	- 39 -
3.2.1.5	Network Assisted GPS .....	- 40 -
3.2.2	Satellite-Based Positioningtechnologieën .....	- 42 -
3.2.2.1	Global Positioning System .....	- 43 -
3.2.3	Tracking-Methodes in de Praktijk.....	- 45 -
3.2.4	Privacy.....	- 47 -
	<b>Hoofdstuk 4</b> <i>Het Modelleren en Bewaren van Bewegende Objecten</i> .....	- 48 -
4.1	Reconstructie en Representatie van Trajecten .....	- 48 -
4.1.1	Omgaan met Locatieonzekerheid .....	- 50 -
4.2	Trajectdata Modellen .....	- 52 -
4.2.1	Een Semantisch-Gebaseerd Trajectdata Model .....	- 54 -
4.3	Moving Objects Databases .....	- 55 -
4.3.1	Trajectindexering .....	- 55 -
4.3.2	Het Managen van Locatieonzekerheid.....	- 57 -
4.3.3	Traject-Data Warehouses .....	- 59 -
	<b>Hoofdstuk 5</b> <i>Mobiliteitskennis</i> .....	- 60 -
5.1	Spatial Knowledge Discovery .....	- 60 -
5.1.1	Geografische Informatie Systemen .....	- 61 -
5.1.2	Spatial Feature Extraction.....	- 62 -
5.1.3	Spatial Data Mining.....	- 62 -
5.1.3.1	Clusteren, Classificatie en Regressie .....	- 63 -
5.1.3.2	Associatieregels .....	- 64 -
5.2	Spatiotemporal Knowledge Discovery .....	- 65 -
5.2.1	Spatiotemporal Feature Extraction.....	- 65 -
5.2.2	Spatiotemporal Data Mining.....	- 66 -
5.2.2.1	Clusteren .....	- 67 -
5.2.2.2	Spatiotemporal Patronen .....	- 68 -
5.2.2.3	Predicties .....	- 71 -
5.2.2.4	Privacy-Preserving Data Mining .....	- 73 -
	<b>Hoofdstuk 6</b> <i>De UbiTracker Applicatie</i> .....	- 74 -
6.1	Hoofddoeleinden .....	- 74 -
6.2	De Positioneringmethode .....	- 75 -
6.2.1	Praktijktoetsing .....	- 76 -

6.3 Architectuur en Functionaliteit .....	- 80 -
6.3.1 Werking .....	- 82 -
6.3.2 De databases .....	- 84 -
6.3.3 Foutafhandeling .....	- 85 -
6.3.4 Reverse Engineering .....	- 86 -
6.3.4.1 Klassendiagram.....	- 86 -
6.3.4.2 Activiteitendiagram.....	- 88 -
6.4 Optimalisatiemogelijkheden .....	- 90 -
<b>Algemene Conclusie.....</b>	<b>- 91 -</b>
<b>Lijst van de Geraadpleegde Werken.....</b>	<b>- 95 -</b>
<b>Bijlagen .....</b>	<b>- 102 -</b>
Bijlage A Online module van de UbiTracker applicatie.....	- 102 -
Bijlage B Gebruikte codes.....	- 103 -
Bijlage B.1 UbiTracker Java ME code .....	- 103 -
Bijlage B.2 Db script.....	- 113 -
Bijlage B.3 Streetandinfo script.....	- 113 -
Bijlage B.4 UbiTracker script .....	- 113 -
Bijlage B.5 Station script .....	- 114 -
Bijlage B.6 Trackandtrace script.....	- 115 -
Bijlage B.7 XMLread script .....	- 116 -
Bijlage B.8 Home script .....	- 117 -
Bijlage B.9 Distance script .....	- 117 -
Bijlage B.10 Map script .....	- 119 -

# Lijst van Figuren

<b>Figuur 2-1</b> De drie golven van computing .....	- 6 -
<b>Figuur 2-2</b> Het voorkomen van media breaks.....	- 10 -
<b>Figuur 2-3</b> Ubiquitous transport- en mobiliteitstoepassingen.....	- 19 -
<b>Figuur 2-4</b> Het automatic log-on systeem.....	- 21 -
<b>Figuur 2-5 (L)</b> Traject- en snelheidsdeductie aan de hand van C-FVD, <b>(R)</b> G-FVD..	- 23 -
<b>Figuur 3-1 (L)</b> Ruimte-tijd pad, <b>(R)</b> een traject .....	- 28 -
<b>Figuur 3-2</b> Ruimte-tijd prisma.....	- 29 -
<b>Figuur 3-3 (L)</b> Cell Identity, <b>(R)</b> Cell Identity & Timing Advance .....	- 37 -
<b>Figuur 3-4</b> Time of Arrival .....	- 39 -
<b>Figuur 3-5 (L)</b> De E-OTD componenten, <b>(R)</b> hyperbolic positioning.....	- 40 -
<b>Figuur 3-6</b> Network Assisted GPS .....	- 41 -
<b>Figuur 3-7</b> GPS plaatsbepaling met van één, twee of drie satellieten.....	- 44 -
<b>Figuur 4-1</b> Een traject als een cilindrisch volume in 3D .....	- 51 -
<b>Figuur 4-2 (L)</b> Moving region, <b>(R)</b> moving point .....	- 52 -
<b>Figuur 4-3 (L)</b> MBRs, <b>(R)</b> de corresponderende R-tree .....	- 56 -
<b>Figuur 4-4 (1)</b> Possibly-Sometime-Inside, <b>(2)</b> Possibly-Always-Inside, <b>(3)</b> Definitely-Always-Inside.....	- 58 -
<b>Figuur 5-1</b> Voorbeeld van een afstandsgebaseerde clustermethode .....	- 68 -
<b>Figuur 5-2</b> De hoofdbenaderingen inzake privacy preserving data mining .....	- 73 -
<b>Figuur 6-1</b> Het Antwerpse testtraject samen met de geregistreerde BTSs.....	- 77 -
<b>Figuur 6-2</b> Het tweede testtraject samen met de geregistreerde BTSs .....	- 78 -
<b>Figuur 6-3</b> Gelogde posities op het tweede testtraject.....	- 79 -
<b>Figuur 6-4</b> Testlocatie in Diepenbeek .....	- 80 -
<b>Figuur 6-5</b> Stappen binnen de UbiTracker applicatie .....	- 81 -
<b>Figuur 6-6</b> Grafische weergave van de UbiTracker applicatie.....	- 83 -
<b>Figuur 6-7</b> Het UbiTracker klassendiagram .....	- 87 -
<b>Figuur 6-8</b> Het UbiTracker activiteitendiagram .....	- 89 -
<b>Figuur 6-9</b> Optimalisatiemogelijkheid aangaande de positionering van de gebruiker-	- 90 -

# *Hoofdstuk 1*

## *Probleemstelling*

### *1.1 Praktijkprobleem*

Het praktijkprobleem situeert zich op het raakvlak van twee belangrijke deeldomeinen. Aan de ene kant hebben we de nieuwe ontwikkelingen binnen de mobiele communicatietechnologieën en *ubiquitous computing*, welke hebben geleid tot een stijging in zowel de kwantiteit als de kwaliteit van data die continu en automatisch rondom ons worden verzameld, op een zeer gedistribueerde manier, voor een veelvoud aan doeleinden. Onze dagelijkse activiteiten en de manier waarop we leven en bewegen laten digitale sporen na. Voorbeelden hiervan zijn onder andere telecommunicatiedata van mobiele telefoons, plaatsgebonden informatie van *location-aware devices* of data verzameld via allerlei sensoren en *smart-tags*. Deze recente ontwikkelingen en hun plaats- en tijdsgebonden data bieden ons talrijke mogelijkheden, maar evenzeer uitdagingen en risico's (Gianotti & Pedreschi (Eds.), 2007).

Aan de andere kant daarentegen hebben we het prominente en omvangrijke transport- en mobiliteitsdomein en de vele toepassingen hierbinnen ter ondersteuning van deze mobiliteit en haar houdbaarheid.

Met deze thesis tracht ik vervolgens een blik te werpen op het onderzoeksdomein dat beide deeldomeinen integreert. Hierbij zullen we ons vooral focussen op de stappen binnen het proces van de verzameling van allerhande vormen van *spatiotemporal* mobiliteitsdata, tot de ontwikkeling van diverse *ubiquitous* mobiliteitstoepassingen in het alledaagse leven. Deze data moeten namelijk worden verzameld, gebaseerd op voorgaande technologieën en vervolgens verwerkt, ter ondersteuning van deze mobiliteitstoepassingen. De Metropolen van de 21e eeuw worden namelijk gekenmerkt door complexe sociale systemen, waardoor observaties van verplaatsingspatronen en

gedragsmodellen van mensen zeer belangrijke hulpmiddelen kunnen vormen voor onder andere verkeersdeskundigen en allerhande overheidsbeambten, ter ondersteuning van de mobiliteit (Gianotti & Pedreschi (Reds.), 2007). Dezelfde informatie kan ook relevant zijn voor allerlei andere toepassingsgebieden binnen transport en mobiliteit, zo denk ik aan *location-based services* of *-marketing*, om logistieke systemen op poten te zetten, of om de modale huisman te helpen bij het plannen van zijn verplaatsingen. Vervolgens zullen de uitdagingen enkel maar talrijker worden tijdens de *real-life* ontwikkeling van deze toepassingen. Zo moet onder andere worden gedacht aan de architectuur van zulke systemen en toepassingen, hoe dat de nodige *interfaces* worden ontworpen en op welke manieren de gebruiker interactie heeft met de toepassing.

Deze recente ontwikkelingen en haar *spatiotemporal* mobiliteitsdata bieden ons dus talrijke mogelijkheden op vlak van transport en mobiliteit, maar ter realisatie hiervan moeten we het hoofd kunnen bieden aan allerhande uitdagingen en risico's op zowel het technologische, als het sociale en ethische vlak. Op het technologische vlak heeft dit vooral betrekking op het leren omgaan met bewegende objecten in het 4D universum van plaats en tijd. Methodes moeten ondermeer ontwikkeld worden voor het representeren, bewaren en managen van deze objecten. Daarnaast kunnen deze data, samen met haar verdere verwerking, een belangrijke bedreiging vormen voor onze individuele privacy en veiligheid (May & Saitta (Reds.), 2007). Deze mobiliteitsdata bevatten immers persoonlijk gevoelige informatie, waardoor de ontwikkeling van betrouwbare technieken en technologieën een belangrijke succesfactor zal worden binnen de sociale acceptatie van *ubiquitous* transport- en mobiliteitstoepassingen (Gianotti & Pedreschi (Reds.), 2007). Het verkrijgen van potentiële voordelen door middel van betrouwbare en effectieve technologieën is een zeer uitdagend doel, maar eens vervuld zou dit een positieve sociale en economische impact moeten creëren en het publieke nut ten goede komen (Gianotti & Pedreschi (Reds.), 2007).

Dit praktijkprobleem speelt zich af zowel op het academische, als het praktische gebied. De uitdagingen gesteld door de technologische ontwikkelingen en haar data zullen enerzijds conceptueel grondig moeten worden uitgewerkt door academici en wetenschappers. Deze oplossingen zullen dan later specifiek worden aangepast aan de noden van de praktische situaties en toepassingen. De eerder besproken mogelijkheden, uitdagingen en risico's vereisen dus een gezamenlijke inspanning van onder andere academici, wetenschappers binnen talloze velden (geografie, statistiek, sociale

wetenschappen, rechten, telecommunicatie, verkeerskunde, informatica, enz.) technici, onderzoeksinstituten, overheden en de modale consument.

## 1.2 *Onderzoeksprobleem*

### 1.2.1 *Centrale Onderzoeksvraag*

Het uiteindelijke doel van dit werk is het formuleren van een uitgebreid en tevens duidelijk antwoord op de volgende centrale onderzoeksvraag:

*Wat zijn de mogelijkheden, uitdagingen en risico's die de technologische evoluties binnen ubiquitous computing bieden op vlak van transport en mobiliteit?*

Zoals eerder opgemerkt binnen de probleemstelling zijn er een heleboel mogelijkheden en voordelen die *ubiquitous computing* ons kan bieden op het vlak van transport en mobiliteit. Om hier echter ten volle gebruik van te kunnen maken, moeten we eerst een aantal uitdagingen en risico's overmeesteren in verband met mobiliteitsdata, haar verzameling, opslag en verwerking en de technologische evoluties binnen *ubiquitous computing*. In mijn onderzoek zou ik daarom graag aantonen wat juist die mogelijkheden zijn en hoe we hier dan ook geschikt gebruik van kunnen maken in allerhande *ubiquitous* mobiliteitstoepassingen.

### 1.2.2 *Deelvragen*

Om tot een gestructureerd antwoord op de centrale onderzoeksvraag te komen, is het nuttig een aantal deelvragen te bepalen waarop we in de loop van het onderzoek een antwoord zullen formuleren. Het zijn subdoelen die als het ware een leidraad doorheen het onderzoek vormen.

Deze deelvragen zouden het complete proces van mobiliteitsdata tot mobiliteitsinformatie en -kennis, en van deze informatie en kennis tot *real-life, ubiquitous* mobiliteits-toepassingen moeten behandelen. Hierbij denken we aan de volgende stappen: hoe worden deze mobiliteitsdata verzameld, bewaard, geanalyseerd en verwerkt, wat zijn de eigenschappen van deze data en wat zijn de nodige stappen en uitdagingen om tot een werkelijke applicatie te komen? Dit leidt ons tot de formulering van volgende deelvragen:

1. *Wat verstaan we onder ubiquitous computing, haar mogelijkheden, risico's en uitdagingen?*
2. *Welke technologische ontwikkelingen binnen mobiele communicatie en ubiquitous computing kunnen het verzamelen van mobiliteitsdata vergemakkelijken?*
3. *Wat zijn de kenmerken van deze mobiliteits- en transportdata?*
4. *Welke uitdagingen bieden deze data in verband met preprocessing, opslag en managing voor verder gebruik?*
5. *Hoe halen we relevante informatie en kennis uit deze rauwe mobiliteitsdata?*
6. *Wat zouden de toepassingsmogelijkheden van ubiquitous computing kunnen zijn binnen transport en mobiliteit, hoe zien deze er uit in de praktijk en hoe komen deze tot stand?*

### *1.2.3 Onderzoeksopzet*

Eerst en vooral zullen deze deelvragen gestaafd worden door een grondige literatuurstudie. Deze literatuurstudie dient vooral om verkennend met de reeds aangehaalde technologieën en technieken om te gaan. Dikwijls ben ik ook beperkt tot een literatuurstudie, aangezien veel van de technieken en technologieën (bvb. *spatiotemporal knowledge discovery*) zich nog maar in hun ontwikkelingsfase bevinden.

De laatste deelvraag die zich toespitst op de toepassingen in de praktijk biedt vervolgens de ruimte om deze technologieën en technieken in de mate van het mogelijke eens vanuit een praktische zin te benaderen. Eerst en vooral zullen we hier een aantal toepassingen binnen transport en mobiliteit bekijken. Daarnaast is het hier ook de bedoeling om zelf een *ubiquitous* mobiliteitstoepassing te ontwikkelen. Hierbij is het dan ook grotendeels mijn inbreng die deze deelvraag vorm zal geven.

## Hoofdstuk 2

# Ubiquitous Computing

### 2.1 Op weg naar Smart en Inter-Verbonden Objecten

*"The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it."* (Weiser, 1991)

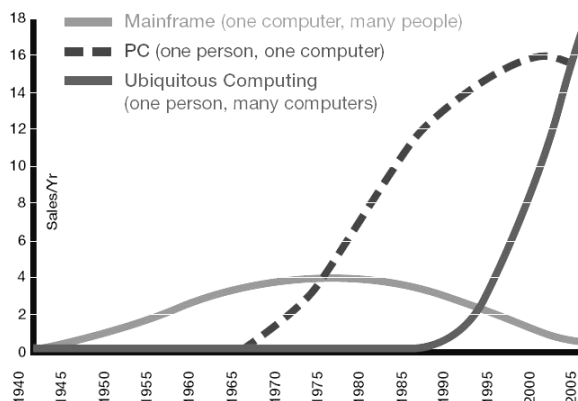
De alsmaar verdergaande miniaturisatie van computertechnologieën zal in de nabije toekomst leiden tot de integratie van diverse microprocessors en sensoren in meer en meer dagdagelijkse objecten. Dit zal uiteindelijk resulteren in het verdwijnen van de traditionele PC *input* en *output* media zoals we die nu kennen, voorbeelden hiervan zijn onder andere het toetsenbord, de muis en het beeldscherm. Daarentegen zullen we rechtstreeks gaan communiceren met onder andere onze kledij, horloges, verlichting en meubels. Deze objecten zullen ook met elkaar, evenals met objecten van anderen communiceren (Mattern, 2004).

Meer dan 15 jaar geleden reeds voorspelde Mark Weiser deze ontwikkelingen en beschreef ze in zijn invloedrijk artikel *'The Computer for the 21st Century'* (Weiser, 1991). Weiser, een onderzoeker aan het XEROX's Palo Alto onderzoekscentrum, sprak in zijn visie over *'ubiquitous computing'*, een term verwijzend naar de alom aanwezigheid van computers die mensen helpen in hun dagelijkse leven, thuis en op het werk. Deze zouden onzichtbaar moeten functioneren op de achtergrond en mensen grotendeels bevrijden van verschillende routineuze activiteiten. In 1999 gebruikte de Information Society Technologies Advisory Group (ISTAG) van de Europese Unie de term *ambient intelligence* om een visie te beschrijven gelijkaardig aan die van Weiser. Vandaag de dag wordt ook de term *pervasive computing* gebruikt in een gelijkaardige betekenis (Bohn, Coroama, Langheinrich, Mattern & Rohs, 2004). In het verdere verloop van de tekst zullen we echter consistent de term *ubiquitous computing* hanteren.



Vandaag de dag zien we dat het internet zo goed als alle computers ter wereld met elkaar verbindt. Hieruit vertrekkende zouden we vanuit een technologisch standpunt *ubiquitous computing* kunnen beschrijven als het fenomeen dat de resterende zaken verbindt met het net en met elkaar. Dit met de bedoeling informatie te kunnen leveren aan alles, eender waar of wanneer (Mattern, 2001). De mens wordt als het ware omsingeld door ontelbare, kleine en inter-verbonden computers, onzichtbaar ingebouwd in zowat alle soorten van dagdagelijkse objecten en apparaten. In combinatie met allerlei sensoren ter waarneming van hun omgeving, zouden microprocessors deze objecten kunnen voorzien van allerhande *processing*- en communicatiecapaciteiten. Deze extra dimensie maakt bovendien dat deze objecten en apparaten kunnen handelen en reageren op een omgevingsgevoelige manier, zodat deze als 'smart' kunnen worden beschouwd (Mattern, 2004). Ze zouden dan niet alleen kunnen communiceren met mensen en andere objecten, maar onder andere ook ontdekken waar ze zich bevinden, welke objecten zich in hun nabijheid vertoeven en wat er met hen gebeurd is in het verleden (Mattern, 2001).

Volgens Weiser en Brown (1996) vormt *ubiquitous computing* de derde grote golf binnen *computing* (Figuur 2-1). Belangrijke golven van technologische veranderingen, beweren ze, veranderen fundamenteel de plaats van technologie in ons leven. Het is niet de technologie zelf, maar haar relatie met ons die belangrijk is. De twee voorgaande grote golven zijn *mainframe computing*, waarbij de computer nog een zeldzaam gegeven was, en *personal computing*, gekenmerkt door de *personal* computer. Het is het internet en haar resulterende *distributed computing* die zorgen voor de overgang naar *ubiquitous computing*, waarbij de technologie verdwijnt in de achtergrond.



**Figuur 2-1** De drie golven van computing (Ley, 2007)

## 2.2 Technologische Trends en Ontwikkelingen

Weiser (1991) beweerde dat de fysieke wereld naadloos zou worden verbonden met haar belichaming in informatie systemen. Zijn visie was niet zomaar uit de lucht gegrepen, maar gestaafd door allerlei drijvende krachten achter *ubiquitous computing*, welke we kunnen groeperen in vijf technologische trends en ontwikkelingen (Strassner & Schoch, 2002). Deze tonen tevens ook aan dat *ubiquitous computing* geen verre droom is, maar zich meer en meer begint te profileren in onze hedendaagse maatschappij.

De eerste twee trends die mede deze technologische evolutie mogelijk maken, kunnen we vinden binnen de micro-elektronica en haar lange termijn ontwikkelingen. De wet van Moore (Moore, 1965), ontwikkeld in de late jaren 60 door Gordon Moore, geldt nog steeds met een verbazingwekkende accuraatheid en consistentie. Deze wet beweert dat het aantal transistoren op dezelfde chip ongeveer verdubbeld om de 18 maanden, of anders gezegd, dat de kracht van microprocessors verdubbeld om de 18 maanden (Bohn, Coroama, Langheinrich, Mattern & Rohs, 2004). Deze groei in efficiëntie, welke zich nog een drietal decennia zou voorzetten, resulteert in een neerwaartse spiraal van zowel de afmetingen, als de prijs van computerchips met een equivalente *computing* kracht, waardoor deze kunnen worden geïntegreerd in allerhande dagdagelijkse objecten (Mattern, 2004). Ook andere ontwikkelingen binnen microsystemen en nanotechnologie zijn belangrijk. Zo worden de evoluties binnen de sensortechnologie beschouwd als de tweede trend. Net zoals computerchips worden sensoren steeds kleiner en goedkoper en kunnen we ze terugvinden voor meer en meer doeleinden (Strassner & Schoch, 2002).

Ontwikkelingen binnen de materialenwetenschap vormen de derde trend. Computers van de toekomst zullen een compleet andere vorm hebben en misschien niet eens meer herkenbaar zijn wanneer ze verdwijnen in hun omgeving (Mattern, 2004). Beeldschermen worden bijvoorbeeld alsmaar van een dunner en flexibeler materiaal geproduceerd, waardoor ze in de toekomst zonder problemen hun intrede zullen doen in onder andere onze kleding.

Evoluties op vlak van draadloze communicatie kunnen worden beschouwd als de vierde significante trend. Mobiele communicatietechnologie bijvoorbeeld, valt nimmer weg te denken uit onze hedendaagse maatschappij en systemen van de volgende generatie

zullen beschikken over een nog grotere bandbreedte en betere mogelijkheden voor data communicatie. *Short-range* communicatietechnologieën, die een zo min mogelijk energieverbruik hebben, zijn ook zeer relevant voor *ubiquitous computing*. Wetenschappers zijn bovendien intensief bezig met de verbetering van technologieën voor de lokalisatie van bewegende objecten (Mattern, 2004; Strassner & Schoch, 2002).

Als laatste trend bekijken we de evoluties binnen de computerwetenschap. De capaciteit van opslagtechnologieën wordt steeds groter, terwijl het medium zelf alsmaar kleiner wordt. Een ander voorbeeld is de software. Deze wordt steeds krachtiger maar kent ook een evolutie naar alsmaar kleinere objecten toe, zoals de mobiele telefoon of mp3-speler. *Interfaces* en interactiemethoden worden ook voortdurend aangepast afhankelijk van de toepassing.

Op het vlak van *ubiquitous computing* hebben deze vijf trends en ontwikkelingen geleid tot twee enorme technologische evoluties met de potentie onze relatie met onze omgeving grondig te veranderen (Gama & Cornuéjols, 2007):

- *Smart objecten*. De hiervoor besproken trends hebben tot gevolg dat de simplistische objecten in onze omgeving worden voorzien van onder andere *computing* kracht en allerlei sensoren en actuatoren, en veranderen zodoende van statische, levensloze objecten in adaptieve, reactieve systemen met de mogelijkheid om steeds nuttiger en efficiënter te worden.
- *Netwerken*. Deze trends zullen evenzeer leiden tot een explosie van allerhande soorten netwerken, van langeafstandsnetwerken met een grote bandbreedte, tot lokale *wifi* systemen. Deze extra communicatiedimensie biedt de mogelijkheid tot een werkelijke intelligente inter-verbondenheid van de *smart* objecten. Dit is wat de International Telecommunication Union (ITU) 'the internet of things' noemt.

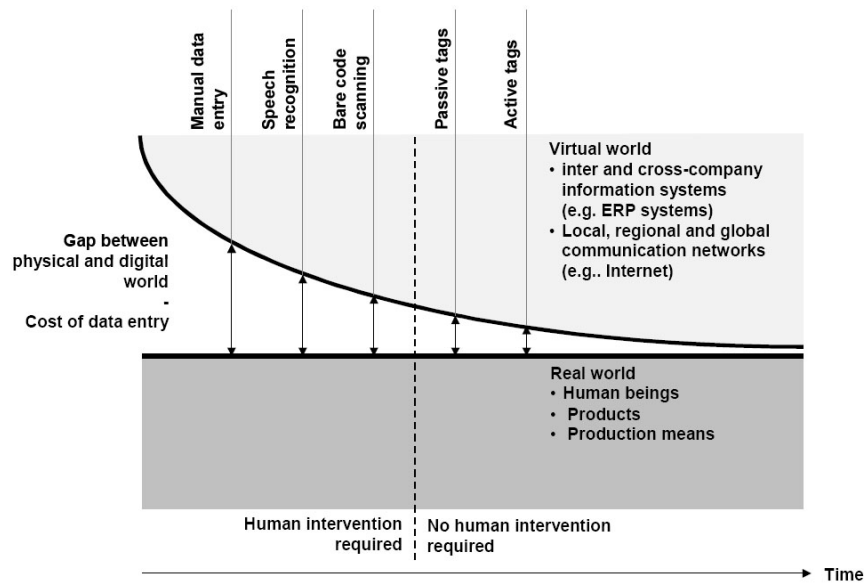
Verschillende nieuwe intelligente applicaties kunnen worden geïmplementeerd, steunend op het voorgaande. Beperkingen omtrent deze toepassingen zullen eerder komen vanuit een economische, sociale en ethische hoek, dan vanuit de technologie zelf. Het is namelijk de technologie die bepaald wat de toekomst kan brengen, maar het sociale proces dat stipuleert wat het mag brengen (Mattern, 2001). Hierop ligt dan ook de nadruk in de volgende secties. De mogelijkheden van de evoluties op vlak van informatie- en communicatietechnologie (ICT) in het algemeen, en de ontwikkelingen

binnen *ubiquitous computing* in het bijzonder zullen namelijk een grondige economische, sociale en ethische impact met zich mee brengen.

## 2.3 *Economische Impact*

Voor het merendeel van bedrijven is continue innovatie een kritische succesfactor om te overleven. Hieromtrent kunnen nieuwe technologieën worden gebruikt ter creatie van een competitief voordeel, nieuwe producten of diensten. Er bestaan echter nog steeds talrijke problemen die niet voldoende kunnen worden verholpen door traditionele IT systemen, zoals *enterprise resource planning* (ERP) of *e-Business* systemen. Vooraleer bedrijven echter plannen een nieuwe technologie te adopteren, willen ze hiervan de *business* impact ruimschoots op voorhand kunnen schatten. Deze impact vertaalt zich binnen het bedrijfsleven namelijk in allerlei voordelen en kosten (Strassner & Schoch, 2002).

Zoals reeds hiervoor vermeld trachten de nieuwe technologieën van *ubiquitous computing* de werkelijke wereld, met inbegrip van haar dagdagelijkse objecten, producten en productiemiddelen, automatisch te koppelen aan de virtuele wereld van het internet, de *e-commerce* en *supply chain management* (SCM) systemen binnen de organisatie. Dit zal enkel mogelijk zijn als technologie de menselijke tussenkomst op verschillende manieren probeert te elimineren (Fleisch, 2001). In de *'real-time'* economie of de *'now'* economie (Siegele, 2002) die op deze manier ontstaat, kan *ubiquitous computing* met inbegrip van haar technologieën drie grote *business* voordelen opleveren. Namelijk het voorkomen van *media breaks* tussen de werkelijke en de digitale wereld (Figuur 2-2), de bewustwording van *smart* objecten en de ondersteuning van mobiliteit (Strassner & Schoch, 2002).



**Figuur 2-2** Het voorkomen van media breaks (Fleisch, 2001)

*Media breaks* vormen zich wanneer *business* processen menselijke tussenkomsten vereisen om data van een bepaald medium naar een ander te transfereren. Dit proberen te elimineren is een sleutelaspect binnen automatisering en biedt voldoende potentieel om de efficiëntie van *business* processen te verhogen. *Ubiquitous computing* kan hierbij gezien worden als de volgende stap in de integratie van businessrelevante objecten met hun belichaming in de virtuele wereld (Fleisch, 2001). Met bewustzijn bedoelen we vervolgens dat objecten de mogelijkheid bezitten om data betreffende hun huidige en historische context te verlenen. Beslissingen die een invloed uitoefenen op het object kunnen tevens door zichzelf worden genomen (Strassner & Schoch, 2002). De ondersteuning van mobiliteit als laatste, kan verklaard worden doordat dataverzameling niet langer gebonden is aan statische technologieën, zoals barcodescanners, maar eender waar en wanneer kan gebeuren. Dit geldt evenzeer voor het gebruik van deze data. Fleisch (2001) beweert dat *ubiquitous computing* ervoor kan zorgen dat data worden gedistribueerd tot op het punt waar ze kunnen worden gebruikt of gegenereerd.

In de praktijk leiden deze *business* voordelen onder andere tot meer efficiënte en effectieve *inventory* managementacties. We beschikken namelijk over *real-time* informatie betreffende de locatie en condities van deze voorraad die accurater is dan ooit, een grotere transparantie binnen de toeleveringsketen en kan het continu

monitoren van kritische productparameters bovendien leiden tot snellere en meer gepaste acties (Bohn, Coroama, Langheinrich, Mattern & Rohs, 2004).

Desondanks het grote aantal potentiële economische voordelen, bestaan er ook substantiële risico's binnen deze economie op autopiloot, welke grotendeels is gebaseerd op *ubiquitous computing* technologieën (Bohn, et al., 2004). Eerst en vooral kan onder andere de verhoogde mate van automatisering, samen met de exclusie van mensen als beslissingnemers leiden tot problemen onder abnormale omstandigheden, bijvoorbeeld diegene niet opgenomen in de softwarecode. Bovendien kunnen we stellen dat, hoewel de implementatie van deze technologieën meestal leidt tot een grotere efficiëntie binnen de economie, dit evenzeer kan resulteren in allerlei problemen. Het is net de *'lack of slack'* die bij onvoorziene onderbrekingen kan leiden tot ernstige consequenties. Een voorbeeld hiervan zou kunnen zijn wanneer elke partner binnen de toeleveringsketen slechts net voldoende voorraad gaat aanhouden. Het is zodoende belangrijk op te merken dat de betrouwbaarheid van deze *'real-time'* economie een belangrijke factor zal zijn binnen haar algemene adoptie.

## 2.4 Sociale en Ethische Impact

Hoewel ontwikkelingen op vlak van informatietechnologie (IT) nooit het expliciete doel hebben gehad om de maatschappij te transformeren, maar dit toch dikwijls deden als neveneffect, stelt de eerder besproken *ubiquitous computing* visie expliciet voor de maatschappij grondig te veranderen door ze volledig te informatiseren. Het is daarom vrij zeker dat deze evoluties een grondige en lange termijn impact zullen hebben op ons dagelijkse leven en onze ethische waarden, welke zelfs verder reikend zal zijn dan die van het internet met al haar perikelen. Doordat *ubiquitous computing* georiënteerd is naar zowel de privé, als de publieke en commerciële wereld, droomt het ervan een technologie te zijn die ons zal begeleiden doorheen onze levens, dag in en dag uit. En als Mark Weiser zijn visie betreffende *invisible computing* ooit werkelijkheid wordt, zullen we haar aanwezigheid niet eens merken (Bohn, Coroama, Langheinrich, Mattern & Rohs, 2004).

Net doordat ubiquitous computing zich op deze manier profileert, zien vele critici het als een vijandige poging om onze dagelijkse levens te doordringen met technologie (Araya, 1995), en groeit volgens Mutanen (2007) de internationale angst betreffende het *big brother* effect ervan. Meestal draait deze angst en kritiek ook rond onwetendheid en onzekerheid aangaande de gevolgen van *ubiquitous computing*, dit merkt ook Lucky op (1999): "Everything will be connected to everything else, but no one has any idea what all those connections will mean" (geciteerd in Bohn, et al., 2004, p. 14). Hoe moeten we deze *smart* apparaten gebruiken in onze dagelijkse routine? Wanneer zouden we ze aan en uit moeten zetten? Wat zouden ze mogen zien, voelen, of horen? En aan wie zouden ze dit verder mogen vertellen. Dit zijn maar enkele voorbeelden van vragen die mensen zich stellen (Langheinrich, 2002).

Het wordt vandaag de dag steeds moeilijker om ons een leven voor te stellen zonder computers. Meer en meer objecten en omgevingen worden namelijk uitgerust met diverse *ubiquitous computing* technologieën, waardoor onze afhankelijkheid van hun correct en betrouwbaar functioneren onvermijdelijk stijgt in dezelfde zin. In de grotendeels gecomputeriseerde toekomst die we tegemoet gaan, zal het steeds moeilijker worden om te ontsnappen aan deze technologisch geïnduceerde afhankelijkheid. Dit zal resulteren in een aantal fundamentele sociale en ethische uitdagingen voor toekomstige *ubiquitous computing* systemen. Privacy is slechts één van deze uitdagingen, hoewel waarschijnlijk de meest prominente (Bohn, et al., 2004). In de volgende subsecties bespreken we de belangrijkste.

### *2.4.1 Privacy in Ubiquitous Computing*

Terwijl veel mensen overtuigd zijn van de talrijke mogelijkheden en voordelen die *ubiquitous computing* voor ons in petto heeft, stellen ze zich desondanks toch vragen betreffende de rol van privacy hierin. Privacy is één van de meest genoemde kritieken op *ubiquitous computing* en zal bijgevolg ook een belangrijke variabele zijn binnen haar succes op lange termijn (Hong & Landay, 2004). Velen maken zich namelijk zorgen voor een potentieel misbruik ervan en een mogelijk gebrek aan controle erop. Kortom, er is nood aan *ubiquitous* systemen die deze privacy incorporeren.

Aangezien continue informatieverzameling, informatie-integratie vanuit verschillende bronnen en het zoeken binnen deze vergaarde datasets aan de basis liggen van alle *smart* systemen, is de impact van deze nieuwe technologieën op onze privacy niet te miskennen. Alledaagse *smart* objecten kunnen worden gelokaliseerd en opgespoord en verschillende objecten rapporteren hun status en sensorinformatie aan andere objecten of gebruikers. Gebruikers van mobiele telefoontechnologie bijvoorbeeld laten voortdurend digitale sporen na van hun geografische positie, welke wordt bewaard binnen de *provider* haar databases, en hebben bovendien de mogelijkheid op te sporen wie in hun omgeving vertoefd. Zelfs nu reeds laten creditcardgebruikers continu informatie achter betreffende de locatie en tijd van hun transacties. Mattern (2004) wijst er ook op dat het in de toekomst, door allerlei data te integreren en te doorzoeken, veel makkelijker zal worden een vrij compleet beeld van een ander individu te vormen. Dit maakt duidelijk dat zonder effectieve beschermingsmaatregelen de technologieën van *ubiquitous computing* ervoor zullen zorgen dat de grenzen van onze privacy op verschillende manieren zullen worden overschreden. Hierdoor is er nood aan zowel wettelijke overwegingen, als nieuwe technologische benaderingen en sociale inspanningen om deze uitdagingen te trotseren.

Op vlak van *ubiquitous computing* zijn er een aantal aspecten van privacy die onze aandacht verdienen en welke een belangrijke rol zullen bekleden binnen een *privacy-aware* implementatie van *ubiquitous* systemen (Giannotti & Saygin, 2007). Als we een sociale acceptatie ervan in de hand willen werken, zullen we binnen deze systemen steeds rekening moeten houden met één of meer van deze aspecten van privacy.

- *Anonymity*. Anonimiteit verzekert dat een individu gebruik kan maken van een middel of een dienst zonder het opgeven van zijn identiteit.
- *Pseudonymity*. *Pseudonymity* verzekert dat een gebruiker een middel of een dienst kan gebruiken zonder het opgeven van zijn identiteit, maar er wel toerekenbaar voor is. In dit geval worden werkelijke identiteiten vervangen met pseudoniemen, identiteiten die niemand buiten de bevoegde personen kan linken aan een specifiek individu. Het verschil met anonimiteit is dat de actoren hier wel distantieerbaar zijn, maar niemand de pseudoniemen kan linken aan de werkelijke identiteiten.
- *Unlinkability*. *Unlinkability* verzekert dat een gebruiker gebruik kan maken van meerdere middelen of diensten, zonder dat anderen deze aan elkaar kunnen linken.



- *Unobservability*. Dit verzekert dat een gebruiker gebruik kan maken van een middel of dienst zonder dat anderen, vooral derde partijen, dit gebruik kunnen observeren. De focus ligt hier eerder op de gebruikte items dan op de identiteit van de gebruiker. Door geen informatie te bieden aangaande de gebruikte middelen of dienst, wordt de privacy van de gebruiker beschermd.

Er bestaan bovendien talloze manieren waarop we het privacy concept vorm kunnen geven, met andere woorden welke inhoud we er aan toekennen en waar de grenzen worden getrokken. Binnen het domein van *ubiquitous computing* en *knowledge discovery* zijn er echter twee categorieën van privacy die zeer relevant zijn. Namelijk, informatie privacy en communicatie privacy (Pedreschi, et al., 2007; Beresford, 2005).

- *Informatie privacy*. Deze vorm van privacy heeft betrekking op de ontwikkeling van regels betreffende het verzamelen en verwerken van persoonlijke data, zoals financiële en medische records. Deze vorm van privacy valt ook dikwijls onder de noemer 'data protectie'. *Ubiquitous computing* technologieën zorgen immers voor een stijging in zowel de kwantiteit als de kwaliteit van data, die continu en automatisch rondom ons worden verzameld voor een veelvoud aan doeleinden. Binnen deze data schuilt belangrijke informatie en kennis, maar de manier waarop we deze data verwerken kan evenwel een bedreiging vormen voor onze individuele privacy.
- *Communicatie privacy*. Dit behandelt de veiligheid en privacy van post, telefoon, e-mail en andere vormen van communicatie. Allerhande vormen van communicatie liggen namelijk aan de grondslag van *ubiquitous computing* systemen waardoor deze vorm van privacy eveneens veel aandacht vereist.

De reden dat privacy zoveel belangstelling krijgt binnen de *ubiquitous computing community*, is dat privacy wordt aanzien als een fundamentele vereiste voor elke moderne democratie (Bohn, Coroama, Langheinrich, Mattern & Rohs, 2004) en dat het een zeer beduidend mensenrecht is, of zou moeten zijn (Pedreschi, et al., 2007). Van alle mensenrechten is privacy waarschijnlijk ook het moeilijkst te definiëren, aangezien het varieert naargelang de context en cultuur. Wat voor de één aanzien wordt als een schending van zijn privacy kan voor de ander overkomen als een normale zaak.

## 2.4.2 *Andere Sociale en Ethische Uitdagingen*

Net doordat *computing* 'ubiquitous' wordt, kan elke situatie plots veranderen in een *computing* situatie (Hartikainen, 2007). *Computing* is immers niet langer beperkt tot onze desktop. Onze huidige wereld wordt bijgevolg steeds grondiger geïnformatiseerd, waardoor het steeds moeilijker zal worden om te ontsnappen aan de invloed van de technologieën deelsluitmakend van *ubiquitous computing*. Zoals eerder vermeld zal *ubiquitous computing* zodoende een hele reeks sociale en ethische uitdagingen met zich meebrengen. Privacy hebben we al behandeld in de vorige sectie, maar dit is slechts het topje van de ijsberg. In dit deel behandelen we een aantal andere sociale en ethische vraagstukken aangaande *ubiquitous computing* systemen. We zullen het namelijk hebben over de betrouwbaarheid van, de controle op en de sociale comptabiliteit en aanvaarding van deze systemen (Bohn, Coroama, Langheinrich, Mattern & Rohs, 2004).

De visie van *ubiquitous computing* beschrijft systemen die volledig discreet en op een *low-profile* manier op de achtergrond, taken en opdrachten voor ons verrichten. Willen we de sociale acceptatie van deze systemen promoten, dan moeten we ten allen tijde op deze systemen kunnen rekenen, in alle mogelijke situaties. Met andere woorden, betrouwbaarheid is een sleutelvariabele binnen het lange termijn succes van *ubiquitous computing*. Bovendien, als het correct en betrouwbaar functioneren van de informatietechnologie infrastructuur van vitaal belang is voor zowel de gemeenschap als het individu, moeten deze systemen niet alleen beschikken over geschikte *back-up* mechanismen om een *denial of service* (DoS) te omzeilen, maar moeten ze eveneens worden ontworpen met deze sleutelvariabele in gedachte (Mattern, 2004). Een communicatieverbinding kan bijvoorbeeld als betrouwbaar worden aanzien indien het systeem diverse parallelle mogelijkheden biedt, zoals GSM, bluetooth en *Wireless LAN* (Bohn, et al., 2004).

Weiser (1991) wees in zijn artikel 'The Computer for the 21st Century' er reeds op dat veel van de sociale hindernissen binnen *ubiquitous computing* te maken zullen hebben met controle, of misschien eerder het gebrek eraan (Weiser, 1991). In de autonome complexe *ubiquitous computing* systemen is menselijke tussenkomst eerder een uitzondering, maar wie is er dan in controle en wie is verantwoordelijk voor de acties ondernomen door het systeem? Bohn et al. (2004) duiden erop dat we controle binnen

*ubiquitous computing* kunnen opsplitsen naar inhoudscontrole en systeemcontrole. Namelijk, als smart objecten informatie over henzelf zouden bieden, wie of wat garandeert dan de accuraatheid en objectiviteit van deze informatie. Betreffende systeemcontrole kunnen we ons de vraag stellen of het de eigenaar is of een andere entiteit, zoals een verzekeringsmaatschappij of de overheid, die de smart objecten kan controleren. Bijvoorbeeld, *smart* auto's zouden kunnen weigeren te starten als de bestuurder teveel alcohol in zijn bloed heeft. Naast controle is er ook nog de verantwoordelijkheid voor de ondernomen acties door het systeem. Bijvoorbeeld, hoe zit het met de verantwoordelijkheid in het geval dat *smart* objecten, zoals een pop, beslissingen op haar eigen begint te nemen en zomaar online kleren aankoopt voor zichzelf? Dit vergt zowel een technische als een wettelijke benadering.

Een andere fundamentele sociale uitdaging voor *ubiquitous computing* systemen is hun sociale comptabiliteit en aanvaarding (Bohn et al., 2004). Willen we voor deze systemen een zo groot mogelijke gebruikersbasis creëren dan zal de *ubiquitous computing* omgeving moeten voldoen aan de noden en eisen van een zo groot mogelijke fractie van de doelgroep, waardoor verschillende systeemp parameters dynamisch zullen moeten zijn. Onafhankelijk van de specifieke toepassing kunnen we stellen dat een sociale comptabiliteit vereist dat het systeem transparant, fair en universeel toegankelijk moet zijn. Mensen gaan veel dingen slechts vertrouwen als het gedrag grotendeels transparant is voor hen en fair overkomt. Net doordat *ubiquitous computing* zich profileert als '*invisible computing*' kan transparantie, in verband met bijvoorbeeld dataverzameling, een probleem vormen (Abowd & Mynatt, 2000). Alle lagen van de maatschappij moeten ook de mogelijkheid krijgen om deel te nemen in deze informatiegemeenschap, hierdoor moet het systeem flexibel en dynamisch zijn ten opzichte van de eigenschappen van de gebruikers. Als laatste bekijken we de aanvaarding van *ubiquitous computing* systemen. Hierbij is het hoofdzakelijk belangrijk dat mensen geloven in de mogelijkheden ervan. De invloed van de systemen op onze omgeving en gezondheid ten gevolge van bijvoorbeeld hun energieconsumptie en elektromagnetische straling, kan eveneens een belangrijke factor vormen in de sociale acceptatie ervan.

Al deze sociale en ethische aspecten van *ubiquitous computing* vereisen een grondige analyse en aanpak op verschillende domeinen vooraleer de *ubiquitous computing* visie werkelijkheid zou kunnen worden. Hoeveel mogelijkheden en voordelen deze

technologische evoluties ook bieden, het zal de gemeenschap zijn die bepaalt wat hier ook werkelijk van in huis zal komen.

## 2.5 *Ubiquitous Systemen*

Tanenbaum en Van Steen (2001) definiëren een *distributed* systeem als een verzameling van onafhankelijke computers die zich naar de gebruiker toe echter profileren als één enkel coherent systeem. Hoewel *ubiquitous* systemen enkele overeenkomsten delen met *distributed* systemen, betwisten ze deze definitie desalniettemin op verschillende manieren. Vooreerst zijn er *ubiquitous* systemen zoals *wireless sensor networks*, waarvan de elementen moeilijk te beschouwen zijn als onafhankelijke computers. Ten tweede is de gebruiker frequent het systeem zelf indien dit systeem gebaseerd is op zelfbeheer. Als laatste moeten we constateren dat *ubiquitous* systemen dikwijls zeer complex en dynamisch zijn waardoor coherentie, hoewel zeer gewenst, vaak moeilijk te bereiken valt (Schuster & Wolff, 2007).

Hier bovenop beschrijven Crowcroft (2003) en Gama en Cornuéjols (2007) in hun werken enkele uitdagingen en vereisten gesteld aan *ubiquitous* systemen. Deze systemen moeten eerst en vooral schaalbaar zijn, hiermee bedoelen we haar aanpassingsvermogen aan evoluerende omgevingscondities en gebruikersgewoontes en -noden. De *ubiquity* karakteristiek vraagt eveneens om toegankelijkheid in de breedste zin van het woord. Het systeem moet bovendien niet enkel *context-aware* zijn, maar evenzeer *resource-aware*, daar onder andere *power-*, *processing-* en *opslag-resources* beperkt zijn. Het systeem, met inbegrip van haar *devices*, moet ook over zekere zelfdiagnose beschikken, niet enkel voor het oplossen van problemen, maar eveneens om voorspellingen en advies te kunnen bieden, wat leidt tot zelfconfigurerende systemen. Hiervoor hebben we nood aan een ingebouwde *real-time* redenering, beslissingnamen, *machine learning* en planning. Er zal ook aandacht moeten worden besteed aan aspecten zoals betrouwbaarheid en privacy. Daar deze systemen omgaan met bewegende gebruikers is er bovendien ook nood aan nieuwe interactiemethoden en vormen van draadloze communicatie tussen beiden, maar eveneens tussen *devices* en over toepassingen heen.

De architectuur of technologie van een *ubiquitous* systeem kan volgens Schuster en Wolff (2007) worden geanalyseerd op basis van zes dimensies. Ze hebben het namelijk over:

- *Device*. Wat zijn de opslag- en processingmogelijkheden van elk betrokken apparaat?
- *Communicatie*. Hoe gaan de verschillende apparaten met elkaar communiceren?
- *Gebruikers*. Wie zijn de gebruikers en hoe verloopt de interactie met het systeem?
- *Controle*. Wie controleert elk van de voorgaande componenten?
- *Data*. Welke data worden er verzameld en bewaard?
- *Infrastructuur*. Welke zijn de aanwezige applicaties?

## 2.6 *Transport en Mobiliteit*

Vandaag de dag kunnen we vaststellen dat de exponentiële diffusie van *low-cost computing devices* en het concept van *embedded intelligence*, samen met de vrijwel moeiteloze verzameling en opslag van data, aanleiding geven tot de ontplooiing van allerlei *ubiquitous* toepassingen in een verscheidenheid aan toepassingsdomeinen (Vanhoof, 2007). Hoewel we *ubiquitous computing* in het voorgaande deel op een zeer algemene manier hebben behandeld, willen we ons in het verdere verloop van dit werk, gezien de huidige relevantie, omvangrijkheid en prominentie in het dagelijkse leven, toespitsen op transport en mobiliteit en de rol die *ubiquitous computing* hierin kan spelen.

Het transportnetwerk ondersteunt op een fundamentele wijze de economische, commerciële, sociale en culturele activiteiten, niet enkel binnen Europa, maar in de ganse wereld (Vanhoof, 2007). Zowel individuen als economische entiteiten zijn betrokken bij een verscheidenheid aan activiteiten waarbij het transport van mensen en/of vrachten onontbeerlijk is. Hierbij is het belangrijk op te merken dat het managen van deze activiteiten in tijd en ruimte sterk onderworpen is aan een veelvoud aan factoren (Vanhoof, 2007). Zo zorgen onder andere persoonlijke en sociale beperkingen ervoor dat activiteiten moeten worden voltooid in een beperkt tijdframe en dat sommige activiteiten moeten worden voltooid voordat andere kunnen starten. Daar zo goed als elke activiteit bovendien gebruik maakt van het publieke infrastructuurnetwerk, moeten

ook deze beperkingen mee in rekening worden genomen. Daarenboven zijn allerlei omgevings- en tijdscondities, mogelijke transportmodes en allerhande vereiste prioriteiten eveneens van belang. Elk van deze factoren zullen dan ook, gezien hun prominentie binnen transport en mobiliteit, hun weerslag vinden binnen de *ubiquitous* toepassingen hierin.

Steeds meer non-traditionele toepassingen, gebaseerd op *ubiquitous computing* en rekening houdend met de hiervoor vernoemde factoren, steken vandaag de dag de kop op binnen transport en mobiliteit, focussend op diverse gebruikers (toeristen, businessmensen, consumenten, enz.). Enkele voorbeelden hiervan zijn onder andere *location-* (LBS) en *movement-based services* (MBS), *track* en *trace* toepassingen, verkeersmanagement, *mobile yellow pages* en navigatie- en *traffic control* toepassingen (Renso, et al., 2007). Een toerist ontvangt bijvoorbeeld op zijn PDA interessante en *context-aware* informatie tijdens zijn tour door de stad, of politievoertuigen worden op een intelligente manier door het verkeer geloodst. Andere voorbeelden kunnen we eveneens terugvinden in Figuur 2-3.



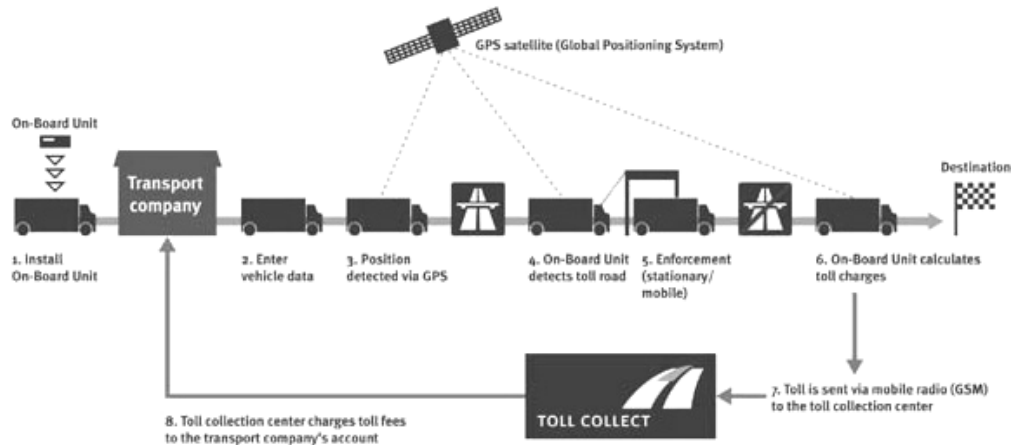
**Figuur 2-3** Ubiquitous transport- en mobiliteitstoepassingen (ETSI, 2008)

Aan de basis van elk van deze '*mobile services*' toepassingen vinden we vervolgens een effectief *ubiquitous* systeem dat mobiliteitsdata verzamelt en verwerkt, en dat afhankelijk van de geografische positie, de mobiele gebruiker voorziet van allerlei informatie en diensten. Echter, vooraleer we dieper ingaan op de implementatieaspecten van *ubiquitous* mobiliteitstoepassingen willen we even stilstaan bij enkele voorbeelden van zulke toepassingen. Dit maakt het mogelijk een beter beeld te vormen van de noodzakelijke generieke onderliggende technieken, elementen en aspecten ervan.

### 2.6.1 LKW-MAUT

Het Duitse LKW-MAUT (Roadtraffic Technology, z.d.) is een elektronisch tolsysteem voor vrachtwagens geïntroduceerd op de 12000 km lange Duitse *autobahn* in januari 2005. De op afstand gebaseerde taks is geldig voor elke vrachtwagen met een gewicht boven 12 ton en wordt berekend op basis van de gereden tolroute, de vervuilingklasse van de vrachtwagen, haar gewicht en het aantal assen. Het door Toll Collect beheerde systeem is gebaseerd op satellietplaatsbepaling (GPS) (*automatic log-on*), maar biedt evenzeer een alternatief hiervoor (*manual log-on*) bestaande uit een manuele betaling op voorhand, of een betaling via het internet. Gezien de relevantie, bekijken we hier echter enkel het eerste alternatief.

Het *automatic log-on* systeem (Figuur 2-4) is gebaseerd op een combinatie van GSM en GPS technologie. Om hiervan gebruik te kunnen maken moeten bestuurders echter beschikken over een *on-board unit* (OBU). De OBU gaat automatisch de afgelegde afstand bepalen, berekent de te betalen tol op basis van de klasse van de wagen en allerlei andere informatie en verstuurt vervolgens deze informatie via GSM communicatie (*cellular*) naar het Toll Collect centrum voor een verdere verwerking.



**Figuur 2-4** Het automatic log-on systeem (Roadtraffic Technology, z.d.)

## 2.6.2 IM@GINE-IT

Om over te schakelen van individueel- naar publiektransport hebben we nood aan allerlei informatie zoals onder andere *park and ride* faciliteiten, vertrekuren van bussen en treinen en connectieroutes van de auto naar het station of perron (Vanhoof, 2007). Een voorbeeld van een LBS toepassing die op deze vereisten inspeelt is het Europese IM@GINE-IT (*Intelligent Mobility Agents, Advanced Positioning and Mapping Technologies, Integrated Interoperable Multi-modal location based services*) project gelanceerd door IST (Information Society Technologies) in januari 2004 met een duur van 24 maanden.

Bestaande *infomobility* diensten zijn meestal lokaal, incompleet en gebruiken variërende interfaces en ondersteunende platformen (IST, 2004). IM@GINE-IT daarentegen, wil een uniek toegangspunt bieden waarlangs de eindgebruiker locatiegebaseerde, inter-modale transportinformatie samen met *mapping*, routing, navigatie en andere gerelateerde diensten kan ontvangen binnen Europa, gebaseerd op de persoonlijke voorkeur van de gebruiker. Voorbeelden van de aangeboden diensten op gebruikersniveau zijn onder andere: hotel en restaurant reservatie, verkeersinformatie, *point of interest* (POI) informatie, publiektransport informatie en multi-modale reisplanning (Bekiaris, Panou, Mizaras, en Petraki, 2004).



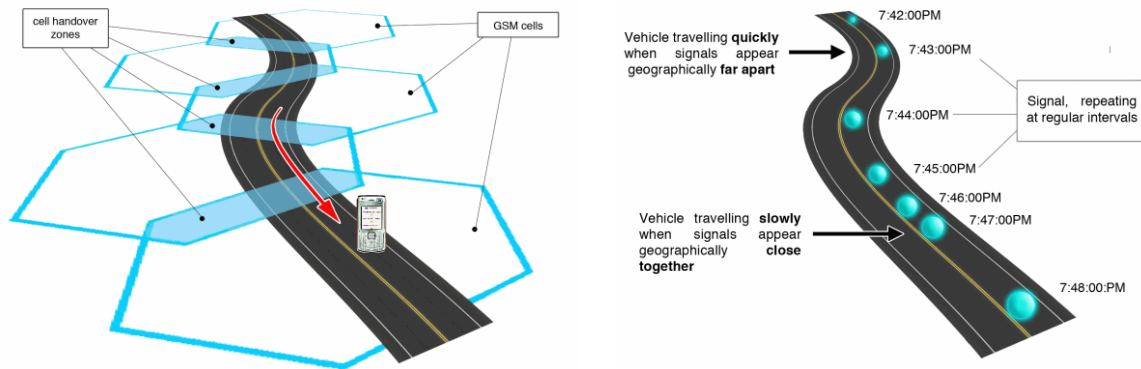
Zonder al te veel in detail te willen treden kunnen we stellen dat systemen ter ondersteuning van zulke toepassing gebruik moeten maken van allerhande communicatietechnologieën zowel binnenin de toepassing als met de gebruiker, *context-aware* moeten zijn, over verbindingen moeten beschikken met verscheidene data-bronnen, gebruik moeten maken van allerlei data-aggregatie en verwerkingsmethoden en moeten worden geïmplementeerd op allerhande mobiele *handhelds* (Vanhoof, 2007).

### 2.6.3 *Be-Mobile*

Be-mobile is een Belgische provider van dynamische verkeers- en mobiliteitsinformatie die hiervoor gebruik maakt van de door ITIS holdings ontwikkelde *Floating Vehicle Data* (FVD) technologie voor data-acquisitie doeleinden, samen met allerhande andere informatiebronnen. Be-mobile verwerkt deze data intern tot kwalitatieve verkeersinformatie en biedt verschillende innovatieve mobiliteitsdiensten aan voor diverse markten zoals onder andere generieke verkeersinformatie voor allerlei media, gepersonaliseerde consumentendiensten via het internet en andere communicatiekanalen en *real-time* mobiliteitsinformatie voor management- en planningsdoeleinden (Be-mobile, z.d.).

De FVD technologie (Be-mobile, z.d.) is een manier om 'flow' informatie te verzamelen, waarbij we de verkeersstroom meten door het anoniem monitoren van voertuigen uitgerust met mobiele telefonie technologie (C-FVD) of GPS-gebaseerde *tracking* apparaten (G-FVD). Deze anonieme GPS en GSM signaaldata worden daarna door de FVD technologie omgevormd tot accurate, *real-time* informatie omtrent de snelheid en richting van de voertuigen bewegend binnen het wegennetwerk en geven ons op deze manier een beeld van de verkeersdichtheid hiervan (Figuur 2-5). Deze accurate omvorming brengt echter wel een aantal uitdagingen met zich mee (ITIS, z.d.). Hierbij horen onder andere de privacyaspecten betreffende de dataverzameling, het omvormen van de miljoenen datapunten naar datastromen en het *map-matchen* van de resulterende patronen teneinde de snelheid en reistijd af te leiden. Hoewel positiebepaling aan de hand het GSM netwerk minder accuraat is dan via het GPS netwerk, wordt dit gecompenseerd door het groot aantal GSM gebruikers op elke weg, de

kennis van het onderliggende wegennetwerk en het gebruik van verschillende statistische technieken (ITIS, z.d.).



**Figuur 2-5 (L)** Traject- en snelheidsdeductie aan de hand van C-FVD, **(R)** G-FVD (ITIS, z.d.)

In België maakt Be-Mobile gebruik van de informatie van het Proximus netwerk en van duizenden professionele voertuigen uitgerust met *track and trace* apparatuur. Ten opzichte van andere complementaire verkeersinformatiebronnen zoals inductielussen en camera's (*fixed* sensorsystemen) biedt FVD onder andere de voordelen dat de dekkingsgraad veel hoger is en dat filepredictie mogelijk wordt aan de hand van een combinatie van historische patronen met weersvoorspelling- en ongevalinformatie.

De Be-mobile operaties worden gemanaged aan de hand van het Be-Mobile *Traffic Platform* (Be-mobile, z.d.). Dit platform vormt het centrale knooppunt tussen de verschillende bronnen van verkeersinformatie en de verscheidenheid aan applicaties, en bestaat uit één enkele database met zowel *real-time* als historische verkeers- en mobiliteitsinhoud.

## 2.6.4 Elementen van Ubiquitous Mobiliteitstoepassingen

Eerder werd reeds gesteld dat we aan de basis van elk van deze 'mobile services' toepassingen een effectief *ubiquitous* systeem terugvinden dat mobiliteitsdata verzamelt en verwerkt, en afhankelijk van de geografische positie de mobiele gebruiker voorziet van allerhande relevante informatie en diensten. Voorgaande voorbeelden van *ubiquitous* transport- en mobiliteitstoepassingen kunnen reeds een gedegen kijk bieden op de onderliggende elementen en aspecten van deze systemen en de gebruikte technologieën en technieken hierbinnen. Hoewel de architectuur en operationele aspecten van het *ubiquitous* systeem afhankelijk zullen zijn van de specifieke toepassing, zijn de te doorlopen cruciale deelstappen voor het tot stand brengen van de geleverde diensten ongeveer gelijkloend. Deze deelstappen omhelzen het complete proces van mobiliteitsdata tot mobiliteitsinformatie en -kennis, evenals het proces van deze informatie en kennis tot *real-life* mobiliteitstoepassingen.

Eerst en vooral zien we dat elk van deze toepassingen voortbouwt op mobiliteitsdata die op één of andere manier moeten worden verzameld. Zo steunen voorgaande toepassingen op de geografische locatie of context van de gebruiker voor het leveren van allerhande informatie en diensten. Vervolgens zal de grote hoeveelheid aan *spatiotemporal* data aangaande de mobiele gebruikers ofwel snel verwerkt moeten worden teneinde hieruit noodzakelijke informatie en kennis af te leiden voor *real-time* doeleinden, ofwel onderworpen worden aan allerlei *knowledge discovery* algoritmes. Eender welke verwerking wordt vereist, het bewaren van deze data zal evenzeer een belangrijk aspect vormen binnen deze toepassingen.

De hierboven aangehaalde aspecten zullen dan ook worden behandeld in het verdere verloop van deze scriptie, gaande van de verzameling en opslag van mobiliteitsdata, tot hun verdere verwerking. Als laatste zullen we eveneens zelf een *real-life ubiquitous* transport- en mobiliteitstoepassing ontwikkelen.

# *Hoofdstuk 3*

## *Mobiliteitsdata*

De moderne samenleving wordt gekenmerkt door een stijging in zowel de kwantiteit als de kwaliteit van data, die continu en automatisch rondom ons worden verzameld voor een veelvoud aan doeleinden.

Dit is eveneens van toepassingen binnen transport en mobiliteit. Het Europese Geographic Privacy-aware Knowledge Discovery and Delivery (GeoPKDD) project leert ons dat nieuwe ontwikkelingen binnen mobiele communicatietechnologieën en *ubiquitous computing* ervoor zorgen dat er een groot volume aan mobiliteitsdata wordt verzameld. Onze dagelijkse activiteiten en de manier waarop mensen leven en bewegen, laten digitale sporen na. Voorbeelden hiervan zijn o.a. telecommunicatie data van mobiele telefoons, plaatsgebonden data van *location-aware devices* of data verzameld via allerlei sensoren en *smart-tags* (Gianotti & Pedreschi (Eds.), 2007).

Vooraleer we deze recente ontwikkelingen binnen mobiele communicatietechnologieën en *ubiquitous computing* bespreken, lijkt het ons nuttig eerst even stil te staan bij deze plaats- en tijdsgebonden mobiliteitsdata. Als laatste bespreken we ook enkele *tracking* methodes uit de praktijk.

### *3.1 Aspecten, Karakteristieken en Concepten*

Aangezien we *ubiquitous* systemen, met mobiliteitstoepassingen in het bijzonder, moeten proberen te vatten vanuit zoveel mogelijk invalshoeken, bekijken we hier hun fundamenteën. In deze sectie zullen we het namelijk hebben over de basisaspecten, karakteristieken en concepten van beweging in het algemeen, en van mobiliteitsdata in het bijzonder. We kijken naar de karakteristieken van *ubiquitous* data, en aangezien

beweging plaatsvindt in zowel ruimte als tijd bespreken we bovendien ook hun eigenschappen. We zullen eveneens kort ingaan op zaken die beweging kunnen beïnvloeden en hierdoor evenzeer aandacht verdienen binnen deze analyse.

### 3.1.1 *Ubiquitous Data*

In veel van de huidige *ubiquitous systemen* gaan de data verwezenlijkt worden op een *ubiquitous* manier. Dit voegt extra facetten toe aan de verzamelde data. Hotho et al. (2007) definiëren *ubiquitous data* als data die gecreëerd worden op een asynchrone, gedecentraliseerde manier door verschillende zwakgekoppelde, gedeeltelijk overlappende en mogelijk contraderende bronnen. We zullen deze karakteristieken vervolgens verder proberen toe te lichten (Hotho, et al., 2007).

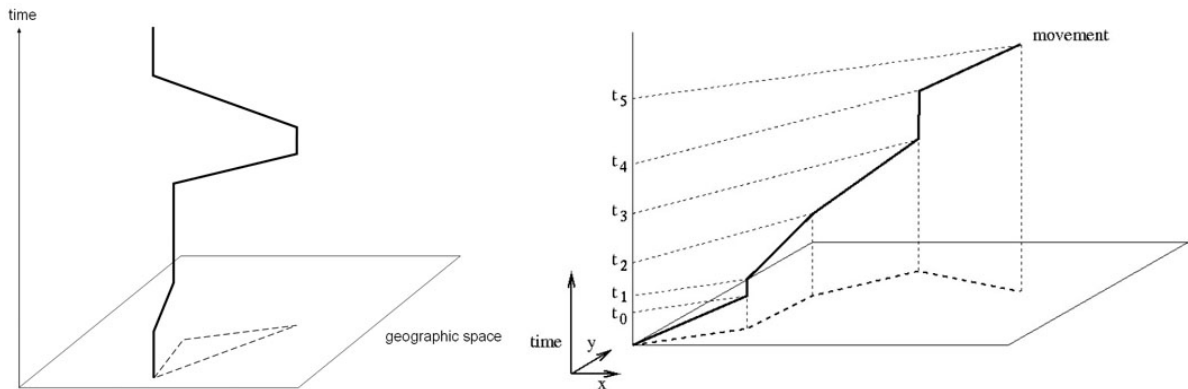
- *Decentralisatie*. Een eerste karakteristiek van dit *ubiquitous* scenario is dat de data asynchroon en op een zeer gedecentraliseerde manier worden geproduceerd. Dit maakt dat het bewaren van deze data in een centraal *datawarehouse* zeer kostelijk kan zijn en soms zelfs onmogelijk. Dit wordt zelfs nog moeilijker als de data op een zeer snelle manier worden gevormd, waardoor we eerder een *data stream* gaan krijgen. Om pragmatische redenen gaan deze data hierdoor dikwijls worden voorzien van tijd- en locatielabels.
- *Heterogene data*. Dikwijls zijn er ook verschillende datatypes bij betrokken. Om een compleet beeld te krijgen van een bepaalde situatie gaan verschillende sensoren bijvoorbeeld zowel afbeeldingdata verzamelen als geluids-, video-, temperatuurs-, licht- en versnellingsmetingen uitvoeren.
- *Meerde bronnen*. In bijna alle *ubiquitous* scenario's zien we dat data worden verzameld door een veelvoud aan gedeeltelijk overlappende en zwakgekoppelde bronnen. Mobiliteitsdata kunnen bijvoorbeeld worden verzameld aan de hand van een sensornetwerk. Hierin bevinden zich een groot aantal verspreide en gedeeltelijk gekoppelde sensors. Metingen van deze sensoren kunnen overlappen (*neighboring sensors*) of elkaar zelfs tegenspreken, indien een sensor bijvoorbeeld niet goed zou werken.

Zoals we verder nog zullen zien leiden deze karakteristieken tot diverse uitdagingen in verband met de verdere verwerking van deze *ubiquitous* data. Onafhankelijk echter van de technologieën en methodes waarmee mobiliteitsdata worden verzameld, gerepresenteerd, geëxploreerd of geanalyseerd, blijven de onderliggende karakteristieken en aspecten van deze data stabiel en behouden ze hun relevantie. Dit geldt evenzeer voor de basisconcepten gerelateerd aan beweging in een geografische ruimte (Andrienko, Andrienko, Peleksis & Spaccapietra, 2007).

### *3.1.2 De Semantiek van Beweging*

In Andrienko, Andrienko, Peleksis en Spaccapietra (2007) kunnen we lezen dat beweging wordt gedefinieerd als een verandering in de fysieke positie van een entiteit met betrekking tot een referentiesysteem, waarin we posities kunnen vaststellen. Doorgaans zal dit referentiesysteem de geografische ruimte zijn.

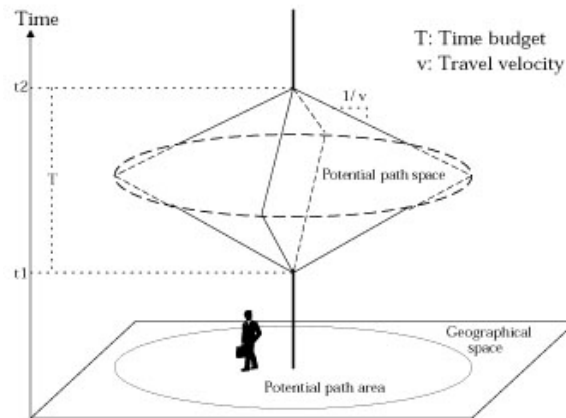
Een belangrijk werk omtrent beweging is dit van Hagerstrand (1970). Hoewel Hagerstrand zijn *time geography* theorie zich eerder bezighoudt met het zoeken naar trends binnen een gemeenschap, is zijn theorie evenwel gebaseerd op de studie van bewegingen van individuen. Hierdoor kunnen we uit deze theorie nuttige ideeën en concepten halen betreffende beweging in het algemeen. Centraal binnen zijn studie van bewegingen, schuilt de opvatting dat ruimte en tijd onafscheidelijk zijn. We kunnen beweging immers enkel waarnemen door een vergelijking tussen twee of meer verschillende tijdsinstanties (Macedo, et al., 2007). Hagerstrand beschouwt beweging dan ook als een ruimte-tijd pad in een driedimensionaal universum. Hierbij worden de horizontale assen (**X**- en **Y**-as) beschouwd als de geografische ruimte, en de verticale as als de tijd (Figuur 3-1). Deze representatie wordt ook wel de ruimte-tijd kubus genoemd (Andrienko, et al., 2007).



**Figuur 3-1 (L)** Ruimte-tijd pad (Keßler, 2006), **(R)** een traject (Brakatsoulas, Pfooser & Tryfona, 2004)

De lijnen in Figuur 3-1 vertegenwoordigen de beweging van een entiteit, waarbij de verticale delen van de lijnen een verblijf op een bepaalde locatie voorstellen. De hellende delen van de lijnen daarentegen vertegenwoordigen een beweging naar een andere locatie. Hoe sneller de beweging, des te vlakker het lijnsegment zal zijn. In Figuur 3-1 veronderstellen we een constante snelheid, dit kan u constateren aan de rechte lijnen. De stippellijn in de beide afbeeldingen stelt de 2D projectie voor van de beweging, beter gekend onder de naam *footprint* (Brakatsoulas, Pfooser & Tryfona, 2004).

Een ander belangrijk concept van *time geography* is de ruimte-tijd prisma (Figuur 3-2). In de 3D representatie is de *potential path space* gelijk aan het volume in ruimte en tijd, dat een persoon kan bereiken in een bepaald tijdsinterval (*time budget*), startend en eindigend op dezelfde locatie (bijvoorbeeld, waar een individu kan geraken tijdens zijn lunchpauze). Haar 2D projectie noemen we de *potential path area*, welke wordt weergegeven als een cirkel, waarbij men veronderstelt dat het individu elke locatie kan bereiken hierbinnen. In werkelijkheid zal het individu echter allerlei beperkingen ondervinden zoals het beschikbare vervoersmiddel of de toegankelijkheid (Andrienko, et al., 2007).



**Figuur 3-2** Ruimte-tijd prisma (Miller, 1991)

### 3.1.3 Trajecten

Een synoniem voor het reeds beschouwde ruimte-tijd pad is een traject. Een traject beschrijft zodoende de beweging van entiteiten (Macedo, et al., 2007). Op een abstract niveau kunnen we trajectdata dan ook beschouwen als bestaande uit drie elementaire componenten (Andrienko, Andrienko, Peleksis & Spaccapietra, 2007):

- *Tijd*. Een verzameling van momenten.
- *Populatie*. Een bewegende entiteit, of een verzameling hiervan.
- *Ruimte*. Een verzameling van locaties die kunnen worden bezet door de populatie.

Een traject vormt zich immers nooit instant, maar vereist eerder een zekere tijdspanne. Voor ieder moment binnen de tijdspanne van het traject bestaat er een locatie in de geografische ruimte, welke bezet kan worden door de populatie. We kunnen een traject dus beschouwen als een functie die de populatie en tijd (onafhankelijk variabelen) relateert aan bepaalde locaties in de ruimte (afhankelijke variabele). Indien we slechts één enkele identiteit beschouwen, kunnen we de populatie negeren, waardoor we een traject ook kunnen voorstellen als een opeenvolging van paren (tijd, locatie). Desondanks tijd, evenals ruimte, continue variabelen zijn, gaan we uit praktische



overwegingen trajecten echter voorstellen door een eindig aantal paren (Macedo, et al., 2007).

Deze tijd-locatie sequenties resulteren vervolgens uit diverse manieren, gebruikt om beweging te observeren en zodanig trajectdata te verzamelen:

- *Tijdsgebaseerde opnames*. De locaties van entiteiten worden met behulp van vaste tussentijden waargenomen, bijvoorbeeld om de 5 minuten.
- *Veranderingsgebaseerde opnames*. Een paar vormt zich wanneer de locatie afwijkt van de vorige.
- *Locatiegebaseerde opnames*. Hierbij wordt een paar gevormd wanneer een entiteit een bepaalde locatie bereikt, bijvoorbeeld aan de hand van een sensor.
- *Gebeurtenisgebaseerde opnames*. Een paar vormt zich op het tijdstip van een bepaalde gebeurtenis, deze wordt meestal opgewekt door de bewegende entiteit. Bijvoorbeeld, de entiteit gebruikt zijn mobiele telefoon.
- Of een *combinatie* van deze basismethodes.

Karakteristieken van trajecten kunnen worden gegroepeerd in twee groepen, gebaseerd op het feit dat ze gerelateerd zijn aan individuele momenten of eerder aan een tijdsinterval (Andrienko, et al., 2007; Brakatsoulas, Pfoser & Tryfona, 2004). Enkele voorbeelden hiervan zijn al eerder vernoemd. De momentgebaseerde karakteristieken bevatten onder andere:

- *Tijd*
- *Locatie*
- *Richting van de beweging*
- *Snelheid*
- *Verandering van richting*
- *Verandering van snelheid*
- *Totale reistijd en -afstand*

De karakteristieken van een traject in het algemeen, of een fragment hieruit gedurende een bepaald tijdsinterval, zijn onder andere:

- *Geometrische vorm, bewegingsvector*

- *Totale reistijd en –afstand*
- *Gemiddelde, mediaan en maximum snelheid*
- *Dynamisch gedrag van de snelheid en richting*

Naast individuele trajecten zijn we meestal ook geïnteresseerd in de vergelijking van twee of meer trajecten. Dit kunnen trajecten van verschillende entiteiten zijn, maar ook van één en dezelfde entiteit op verschillende tijdstippen, of een vergelijking tussen verschillende fragmenten binnen één traject. Hierbij kunnen we de eerder genoemde karakteristieken met elkaar vergelijken, bijvoorbeeld de snelheid en afstand. Maar soms zijn we ook eerder geïnteresseerd in *spatial* of *temporal* relaties. De trajecten hebben bijvoorbeeld locaties met elkaar gemeen (*spatial*), of de trajecten overlappen elkaar in de tijd (*temporal*) (Andrienko, et al., 2007; Brakatsoulas, et al., 2004).

Aangezien beweging steeds plaatsvindt binnen ruimte en tijd, lijkt het ons nuttig hun structuur en eigenschappen eveneens van naderbij te bekijken. Deze hebben dan ook een niet te miskennen invloed op het bewegingsgedrag in het algemeen.

### 3.1.4 Ruimte

We kunnen ruimte in deze context beschouwen als bestaande uit een verzameling van locaties of plaatsen. Een belangrijk aspect is dat we binnen de ruimte een afstand kunnen uitdrukken tussen haar elementen. Aangezien er echter geen natuurlijk begin- of eindpunt bestaat, hebben we nood aan een referentiesysteem zoals geografische coördinaten, om posities kunnen vast te stellen. Desondanks ruimte een continue variabele is, moeten we het uit praktische overwegingen echter beschouwen als discreet, of zelfs bestaande uit een slechts een eindig aantal locaties. Zulke discretisatie is soms ook onlosmakelijk verbonden met een specifieke mobiliteitstoepassing aangezien posities niet altijd specifiek kunnen worden vastgesteld waardoor men eerder zal werken met gebieden, zoals in een mobiel telefoonnetwerk. We moeten dus als het ware een structuur aanbrengen, de ruimte opdelen in gebieden. Dit kan hiërarchisch gebeuren of op een geometrische manier zonder rekening te houden met enige semantiek, bijvoorbeeld de ruimte opdelen in cellen van 1 km<sup>2</sup>. Deze ruimtestructurering biedt ook een alternatief referentiesysteem, door bijvoorbeeld te verwijzen naar straten en

huisnummers. Men kan ook steeds werken met relatieve in plaats van absolute posities, bijvoorbeeld 1 km verwijderd van individu B, of simpelweg met labels zoals 'thuis' en 'werk' (Andrienko, Andrienko, Pelekses & Spaccapietra, 2007).

Het is ook belangrijk te vermelden dat ruimte een heterogene variabele is, wiens eigenschappen kunnen variëren van plaats tot plaats, en zelfs in de tijd. Voorbeelden van relevante karakteristieken van individuele locaties zijn onder andere: aspecten van het terrein (hoogte, helling, enz.) toegankelijkheid, ondergrond, aanwezige objecten, manier van gebruik, activiteitsgebaseerde semantiek en capaciteit (Andrienko, et al., 2007).

### *3.1.5 Tijd*

Het merendeel van de hiervoor besproken aspecten van ruimte zijn ook van toepassing op de tijd variabele. We zullen ze hier kort even bespreken (Andrienko, Andrienko, Pelekses & Spaccapietra, 2007).

Wiskundig bekeken is tijd een continue variabele met een lineaire ordening, waarbij er afstanden bestaan tussen de elementen, welke posities of momenten in de tijd voorstellen. Analoog met ruimte, hebben we ook hier nood aan een referentiesysteem om deze posities in de tijd te kunnen bepalen. Hiervoor wordt meestal de gregoriaanse kalender gebruikt en de standaard opdeling van dagen in uren, uren in minuten en zo verder. De tijd kan ook relatief worden weergegeven of als abstracte labels (vb.  $t_1$ ,  $t_2$ , enz.).

Fysieke tijd bezit ook een aantal inherente cycli. Zo hebben we een natuurlijke cyclus, een gevolg van de rotatie van de aarde, en er bestaan ook cycli gerelateerd aan de mens zijn activiteiten, zoals de werkweek. In verband met onze studie is het zeer relevant te weten welke cycli een invloed hebben op beweging, zodat we deze mee in rekening kunnen nemen.

Diverse tijdstippen hebben ook verschillende eigenschappen in verband met mobiliteit, zo zien we bijvoorbeeld een verschil in bewegingen in het weekend en door de week en bestaat er zelfs variatie binnen één dag.

### 3.1.6 *Bewegende Objecten*

Net zoals locaties in de ruimte en tijd, hebben de objecten die bewegen eveneens hun eigen karakteristieken (Andrienko, Andrienko, Peleklis & Spaccapietra, 2007). Aangezien deze een invloed uitoefenen op de beweging moeten we ze in rekening nemen in onze analyse. Binnen deze eigenschappen vinden we onder andere: beroep, leeftijd en hun gezondheid. Eveneens zal het vervoersmiddel een invloed uitoefenen op de karakteristieken van de beweging, net zoals de activiteiten die worden uitgevoerd tijdens de beweging.

De mens is een voorbeeld van entiteiten die doelgericht bewegen. Het doel beïnvloedt onder andere de route en de snelheid van de beweging. Objecten bewegen ook steeds in een bepaalde omgeving, waardoor de beweging wordt beïnvloed door gebeurtenissen en fenomenen binnen deze omgeving. Voorbeelden hiervan zijn onder andere: weersomstandigheden, sportmanifestaties, klimaat, ongevallen, cultuur enz.

## 3.2 *Positiebepaling - Ubiquitous Mobiliteitsdatabronnen*

*Context-aware* mobiliteitstoepassingen zijn in grote mate gebaseerd op de *real-time* positie evenals beweging van de mobiele gebruiker in het netwerk. Voorbeelden hiervan zijn onder andere *location-* en *movement-based services*, *tracking* en *tracing* en allerlei intelligente transporttoepassingen binnen zowel het goederen- als personenvervoer. De Hoog, Steenbruggen en Crombaghs (2003) vermelden in hun werk drie belangrijke functies van positiedata binnen deze toepassingen. Vooreerst kan de positie de rol van filter vervullen, de verleende diensten worden namelijk bepaald door deze positie. Daarnaast kan de positie fungeren als *pointer* op een kaart of dergelijke voor het representeren van de mobiele gebruiker. Als laatste functie beschouwen ze die van *definer/launcher*, waarbij de ondernomen actie afhankelijk is van het overschrijden van bepaalde grenzen. Gezien de belangrijke rol van de actuele positie of beweging van de gebruiker binnen deze mobiliteitstoepassingen, behandelen we in deze sectie allerlei *real-time ubiquitous* positioneringstechnologieën en -methodes.

Renso et al. (2007) definiëren positioneren als het ontdekken waar de gebruiker zich bevindt binnen het netwerk op een gegeven tijdsinstantie. In deze sectie bespreken we twee belangrijke technologieklassen gebruikt om mobiele gebruikers bewegende binnen een draadloos netwerk te lokaliseren. Deze positioneringstechnologieën vormen als het ware de bronnen van de eerder besproken mobiliteitsdata. Gezien hun relevantie binnen transport en mobiliteit, bovenop het feit dat volgens Ranchordas en Lenaghan (2003) de gebruikersbasis van mobiele communicatie en GPS technologie steeds aanzienlijker wordt, spitsen we ons vervolgens toe op de positioneringstechnologieën gebaseerd op met GPS-uitgeruste objecten (*satellite based*), en diegene gebaseerd op de reeds aanwezige infrastructuur van het draadloze telecommunicatienetwerk (*cellular based*).

Aangezien accuraatheid de belangrijkste kwaliteitsmaatstaf is van positioneringstechnologieën, heeft de industrie het *Quality of Position* (QoP) concept geïntroduceerd (Hjelm, 2002). De accuraatheid van de positionering wordt niet alleen beïnvloed door de kwaliteit van de gebruikte technologie, maar eveneens door de fysieke omgeving (natuurlijke obstakels, verstorende radiosignalen, enz.). Onafhankelijk van de gebruikte technologieën, wordt QoP omschreven door de volgende dimensies (Renso et al., 2007):

- *De accuraatheid van de locatie-informatie.* Bijvoorbeeld hoe dicht we de positie van een mobiel object kunnen bepalen in een *worstcase* scenario.
- *De leeftijd van de locatie-informatie.* Bijvoorbeeld hoeveel tijd er al verstreken is sinds de informatie werd verzameld.
- *De betrouwbaarheid van de accuraatheid.* Bijvoorbeeld de accuraatheid varieert van 0 tot 50 m met een significantieniveau van 90%.

Verschillende toepassingen vereisen welteverstaan verschillende QoP's.

### *3.2.1 Cellular-Based Positioningstechnologieën*

Het mobiele telecommunicatienetwerk staat vooral bekend voor haar spraak- en datacommunicatietoepassingen. Veel minder bekend zijn de mogelijkheden tot *real-time* positionering die het eveneens biedt (de Hoog, Steenbruggen en Crombaghs, 2003).

Om *location-* en *movement-based services* te verschaffen aan hun mobiele klanten, gebruiken mobiele telecommunicatiebedrijven vandaag voornamelijk de *Global System for Mobile Communications* (GSM) positioneringstechnologieën. Het GSM protocol is het meest gebruikte digitale telecommunicatiesysteem voor mobiele telefonie wiens ontwikkeling vooral wordt gestuurd door het European Telecommunications Standards Institute (ETSI) (Renso, et al., 2007). Het GSM systeem is gebaseerd op een *cellular* netwerkarchitectuur welke bestaat uit drie primaire delen, namelijk: het *mobile station* (MS), het *base station system* (BSS) en het *switching system* (Mattsson, 2001). We bespreken vervolgens beknopt deze delen gezien hun relevantie betreffende de hierna behandelde positioneringstechnieken. Het MS bestaat uit twee verschillende entiteiten: het *mobile equipment* (ME) en de *subscriber identity module* (SIM) kaart, welke diverse informatie bevat, nodig voor het ME om gebruik te kunnen maken van het netwerk. Het BSS zorgt vervolgens voor de connectie van het MS met het netwerk en haar belangrijkste componenten zijn de aanwezige antennes (*base transceiver stations*, BTS). Het *switching system* als laatste managet de resulterende communicatie tussen de verschillende mobiele gebruikers.

Als we vervolgens kijken naar de fysieke implementatie (Renso, et al., 2007) van het GSM systeem, zien we dat de geografische ruimte in de *cellular* architectuur wordt gedekt door een aantal BTS, welke een signaal uitzenden dat wordt opgevangen door de aanwezige MS. Deze apparaten bewerkstelligen een connectie met een BTS op basis van signaalsterkte en contractgegevens. Doordat elke antenne één ruimte dekt (cel), wordt de geografische ruimte opgedeeld in een aantal mogelijk overlappende cellen, uniek geïdentificeerd door een antenne of BTS. In stedelijke gebieden liggen deze cellen dicht bij elkaar en hebben ze een kleinere diameter (tussen 100 m en 1 km) dan in landelijke gebieden, waar de radius van een cel kan oplopen tot een maximum van 35 km.

Naar de toekomst toe zal het *Universal Mobile Telecommunications System* (UMTS) echter aan belang winnen (de Hoog, Steenbruggen en Crombaghs, 2003). Deze derde generatie (3G) mobiele communicatie wordt aanzien als de opvolger van GSM (2G). De voordelen van dit telecommunicatiesysteem zijn onder andere de grotere verbindingssnelheid, wat de deur opent voor allerlei nieuwe toepassingen, en de hogere precisie betreffende plaatsbepaling. Dit protocol vereist echter bijkomende investeringen in de infrastructuur van het bestaande GSM netwerk.

Teneinde locatiegebaseerde toepassingen aan te bieden beschikken de *providers* over legio methodes/technieken om de positie van een mobiele gebruiker binnen het netwerk te bepalen. Dit ruime aanbod vroeg echter om standaardisatie, waardoor in 2000 het European Telecommunications Standards Institute (ETSI) een rapport publiceerde met daarin de vermelding van drie verschillende *cellular-based* methodes als toevoeging aan de *Cell Identity* (CI) methode: *Timing Advance* (TA), *Time of Arrival* (TOA) en *Enhanced Observed Time Differences* (E-OTD) (ETSI, 2000). In wat volgt bespreken we uitvoerig deze *cellular* positioneringstechnieken.

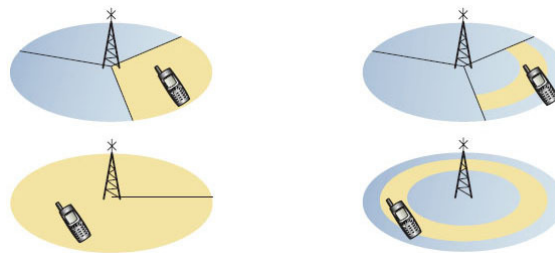
Positioneringmethodes kunnen zowel MS- als netwerkgebaseerd zijn of een hybride techniek vormen, namelijk netwerkgebaseerd, *MS-assisted* en MS-gebaseerd, *netwerk-assisted*. Het verschil is gebaseerd op het feit dat hoofdzakelijk het netwerk, het MS ofwel beiden gebruikt worden om de positie van de mobiele gebruiker te bepalen (Scourias, 1997). *Cellular-based* positionering wordt aanzien als een solide techniek aangezien het BTS signaal ongeveer overal in de wereld beschikbaar is, de penetratiegraad van GSM's vrij hoog is en het netwerk reeds aanwezig is binnen de maatschappij (Chu-feng, 2006). Dit laatste impliceert dan weer wel dat positionering niet het prioritaire doel is van het netwerk, waardoor we onder andere zullen moeten inboeten aan accuraatheid. Als laatste is het nog belangrijk op te merken dat hoewel de resulterende locatiedata afhankelijk zijn van de gebruikte methode, ze echter steeds een indicatie zouden moeten geven van:

- *Tijdstip en datum van detectie.*
- *Positie-informatie.* De positie met betrekking tot een referentiesysteem.
- *Identificatie van de gebruiker.* Bijvoorbeeld het telefoonnummer of de *International Mobile Subscriber Identity* (IMSI) bewaard op de *Subscriber Identity Module* (SIM) kaart.

### 3.2.1.1 *Cell Identity*

De *Cell Identity* (CI) positioneringmethode is de meest simplistische en kan bovendien zowel MS- als netwerkgebaseerd zijn. De methode is gebaseerd op het feit dat het netwerk of het MS de positie van het MS bij benadering kunnen bepalen, daar ze op ieder

moment weten met welke cel het MS is verbonden. De locatie van het MS in deze techniek wordt weergegeven door het vormen van een cirkel rond de geografische positie van het BTS in de cel, dit wordt grafisch weergegeven in Figuur 3-3 (Zeimpekis, Giaglis & Lekakos, 2003). De cirkel stelt het maximale bereik voor van het BTS en haar grootte wordt berekend aan de hand van de signaalsterkte en de gekende signaal attenuatie (Mattsson, 2001). Het voorgaande impliceert dus dat de accuraatheid van deze techniek wordt bepaald door de radius van de cel en is meestal vrij laag, wat de toepassingsmogelijkheden ervan beperkt. De accuraatheid kan namelijk variëren van 100 m tot enkele kilometers (Swedberg, 1999). Dit maakt dat deze techniek iets nauwkeuriger is in stedelijke gebieden (ongeveer 150 m) waar de dichtheid van de GSM-masten aanzienlijk groter is, waardoor de cellen kleiner zullen zijn. Het is eveneens belangrijk op te merken dat gezien allerhande signaaleffecten het sterkste signaal niet steeds van het dichtstbijzijnde BTS hoeft te komen (Mattsson, 2001). De CI methode vereist bovendien geen enkele modificatie van het netwerk of de *handsets*, wat het direct beschikbaar maakt en eveneens goedkoop (de Hoog, Steenbruggen en Crombaghs, 2003). Deze standaardmethode kan worden verbeterd aan de hand van onder andere *Timing Advance* (TA) dat wordt besproken in de volgende paragraaf.



**Figuur 3-3 (L) Cell Identity, (R) Cell Identity & Timing Advance (Swedberg, 1999)**

### 3.2.1.2 Cell Identity en Timing Advance

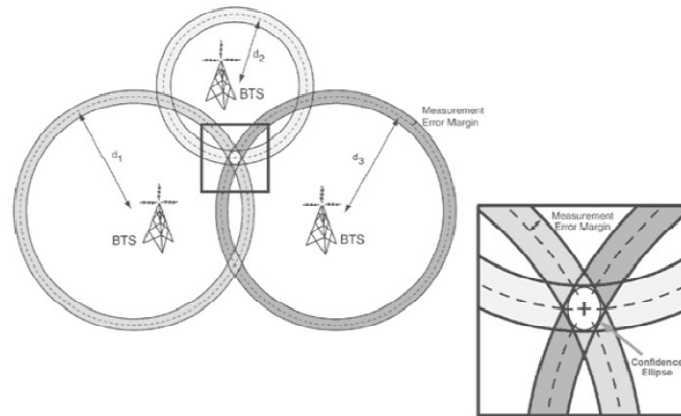
Deze methode, die zowel MS- als netwerkgebaseerd kan zijn, maakt gebruik van zowel de *Cell Identity* (CI) techniek als de *Timing Advance* (TA) parameter om de locatie van het MS te bepalen en de CI standaardmethode te verbeteren. Binnen het netwerk wordt de parameter reeds gebruikt voor synchronisatiedoeleinden, daar het signaal vertraging



oploopt indien de zender ver verwijderd is van de ontvanger (Mattsson, 2001). De TA parameter is een maat voor de afstand tussen het BTS en het MS (Swedberg, 1999), welke wordt bepaald aan de hand van de *round trip delay* van het signaal tussen het MS en het BTS, gebruikmakende van een simpel golfpropagatiemodel (Wang, Min & Byung, 2008). De TA parameter heeft een waarde tussen 0 en 63, wat met een maximum celradius van 35 km overeenkomt met een afstand van ongeveer 550m per eenheid. Dit impliceert dat we door het complementair gebruik van CI en TA de bandbreedte van de resulterende locatiecirkel kunnen verkleinen tot ongeveer 550 m (zie Figuur 3-3). De accuraatheid is bijgevolg iets beter dan bij de CI techniek, maar is sterk afhankelijk van de locatie van de antennes en andere omgevingscondities die de signaalvoortplanting (propagatie) beïnvloeden (Renso, et al., 2007). De Hoog, Steenbruggen en Crombaghs (2003) wijzen er wel op dat deze techniek een aanpassing in de netwerksoftware vereist, wat bijkomende kosten met zich meebrengt.

### 3.2.1.3 *Time of Arrival*

In de Time of Arrival (TOA) netwerkgebaseerde, *MS-assisted* methode wordt de afstand tussen het MS en het BTS bepaald aan de hand van de looptijd van het radiosignaal verzonden vanuit het MS naar het BTS. Deze looptijd wordt berekend door de *location measurement unit* (LMU), aanwezig bij elk BTS. De resulterende TOA waarde kan dan worden gebruikt om het MS op een cirkel rond het BTS te plaatsen, daar de looptijd van het signaal proportioneel is met haar afgelegde afstand. Indien we deze cirkel berekenen voor drie masten geeft de resulterende intersectie hiervan de benaderende locatie van het MS weer (zie *confidence ellipse* in Figuur 3-4). Daar deze methode echter de aanwezigheid vereist van een LMU voor elk BTS in het netwerk, is deze methode vrij duur (D'Roza en Bilchev, 2003). De accuraatheid van deze techniek is gepositioneerd tussen die van de *Cell Identity* en *Timing Advance* methode en die van de E-OTD methode, besproken in de volgende sectie.



**Figuur 3-4** Time of Arrival (Openwave Developer Network, 2002)

### 3.2.1.4 Enhanced Observed Time Differences

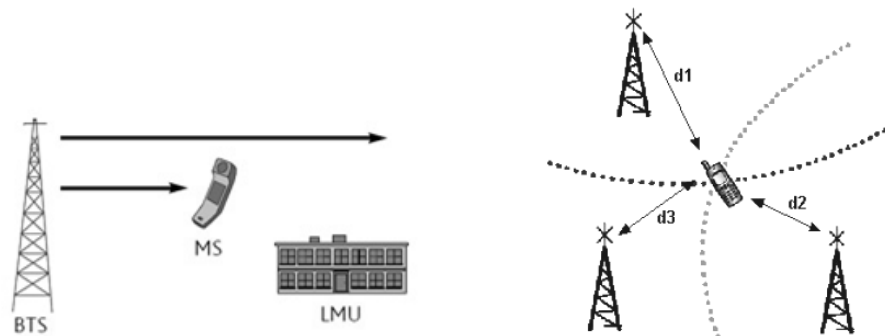
In de *Enhanced Observed Time Differences* (E-OTD) positioneringmethode maken we gebruik van de *multilateration* techniek, ook wel *hyperbolic positioning* genaamd. Hierbij bepaalt het MS haar positie op basis van relatieve metingen, dit in tegenstelling tot de TOA methode waar we met absolute metingen werken (*trilateration*) (Chu-feng, 2006).

De *Observed Time Difference* (OTD) parameter wordt berekend als het verschil in aankomsttijden<sup>1</sup> ( $t_1 - t_2$ ) van signalen verzonden vanuit twee verschillende BTS (Mattsson, 2001). Deze OTD parameter is echter niet voldoende om een accurate positiebepaling te verkrijgen. De verzonden signalen worden echter evenzeer opgevangen door een LMU wiens afstand tot de twee BTS is gekend (Figuur 3-5). Uit deze afstanden samen met de looptijden van de verzonden signalen kan de LMU de *Real Time Difference* (RTD) parameter berekenen, welke gelijk is aan het verschil tussen de verzendtijden van de signalen. Vervolgens kunnen we aan de hand van het verschil tussen de OTD en de RTD waarde, de *Geometric Time Difference* (GTD) parameter, een hyperbool opstellen tussen de twee BTS waarop deze GTD waarde constant is. Doen we dit voor twee verschillende paren BTS, dit wil zeggen minimum drie verschillende masten, dan kunnen we hieruit

---

<sup>1</sup> Bijvoorbeeld 20h22'03" – 20h21'04"

een intersectie bepalen welke de locatie van het MS bepaalt, dit proces kunnen we waarnemen in Figuur 3-5.



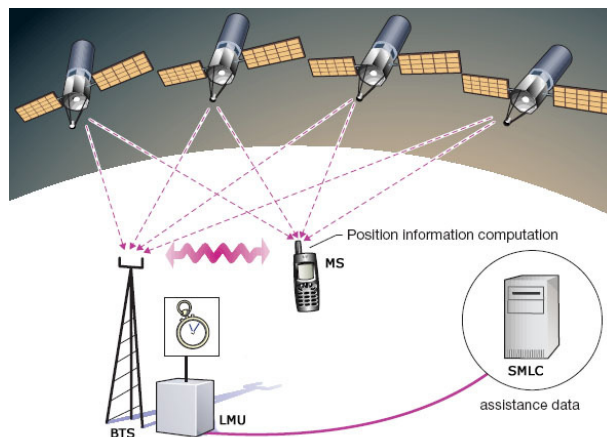
**Figuur 3-5 (L)** De E-OTD componenten, **(R)** hyperbolic positioning (Mattsson, 2001)

Van deze methode zijn er twee varianten beschikbaar, namelijk de E-OTD netwerkgebaseerde, *MS-assisted* methode en de E-OTD MS-gebaseerde, netwerk-*assisted* methode (Renso, et al., 2007; Wang, Min & Byung, 2008). In de *MS-assisted* methode gaat het MS de metingen communiceren naar het netwerk voor verdere berekeningen, dit in tegenstelling tot de netwerk-*assisted* methode waarbij het MS extra informatie verkrijgt vanuit het netwerk zodat het haar positiecalculatie zelf kan verrichten. De accuraatheid van deze methode varieert tussen de 50 en 100 m, afhankelijk van het aantal aanwezige LMU's binnen het netwerk (Renso, et al., 2007). Vergeleken met de TOA methode hebben we hier echter nood aan drie- tot vijfmaal minder LMU's. Aangezien deze methode evenwel additionele en specifieke hardware en software vereisten heeft voor zowel het netwerk als de *handset*, is deze methode eerder duur. Voor deze reden gaan LBS *providers* dan ook eerder goedkopere en simpelere oplossingen prefereren, zoals de *assisted* GPS (A-GPS) besproken in de volgende paragraaf (Renso, et al., 2007).

### 3.2.1.5 Network Assisted GPS

Één van de veel geciteerde nadelen van de GPS technologie (*satellite-based* methodes worden besproken in sectie 3.2.2) is haar benodigde tijd om de initiële locatie van de

mobiele gebruiker te bepalen (*time to first fix*, TTFF), ook wel eens 'cold start' genoemd (Chu-feng, 2006). Het is hier onder andere dat de *network assisted* (A-GPS) hybride techniek op inspeelt. In de A-GPS techniek gaat het MS de aankomsttijden van de signalen, uitgezonden door vier of meer satellieten, gebruiken om haar positie te bepalen. Om de benodigde tijd hiervoor echter te beperken en de accuraatheid te bevorderen, maakt de A-GPS methode hier bovenop eveneens gebruik van het GSM netwerk. Het GSM netwerk bekleedt binnen deze techniek namelijk de functie van referentienetwerk, bestaande uit ontvangers (BTS) met gekende locaties, een continue beschikbaarheid en een goede signaalontvangst (Mattsson, 2001). Binnen de A-GPS techniek gaat het referentienetwerk assisterende informatie bieden zoals onder andere satellietgegevens en GPS-foutencorrecties (Wang, Min & Byung, 2008). Het netwerk kan bijvoorbeeld voorspellen welke GPS signalen het best ontvangen worden door het MS en deze informatie vervolgens doorsturen naar het MS. Hierdoor beperken we de *search-space* en verkorten we de TTFF van minuten tot seconden, wat eveneens resulteert in een zekere energiebesparing (Djukenic en Richton, 2001). De werking van de A-GPS techniek is vereenvoudigd voorgesteld in Figuur 3-6.



**Figuur 3-6** Network Assisted GPS (Swedberg, 1999)

Afhankelijk van de vereiste accuraatheid heeft deze techniek nood aan een verschillend aantal LMU's binnen het netwerk, dit aantal zal echter steeds beperkter zijn dan dat noodzakelijk bij de TOA en de E-OTD methode. Zo hebben we bijvoorbeeld voor een accuraatheid van 10-20 m slechts nood aan één enkele LMU binnen het netwerk, elke 300 km, wat de implementatie substantieel goedkoper maakt (Swedberg, 1999). De

techniek presteert het beste in landelijke gebieden, aangezien in stedelijke gebieden gebouwen en andere obstakels de signalen van de satellieten verstoren. Hierdoor hebben we een accuraatheid die varieert van 2 m (landelijk) tot 20 m (stedelijk). In termen van de kwaliteit/kosten ratio is deze techniek bovendien ook zeer efficiënt en levert het betrouwbare informatie. Aan de andere kant vereist dit wel dat het MS GPS compatibel is (Renso, et al., 2007).

### 3.2.2 *Satellite-Based Positioningtechnologieën*

Als alternatief voor de zojuist behandelde *cellular-based* positioneringmethodes bekijken we hier de satellietplaatsbepaling. Moderne *satellite-based* positioneringtechnologieën zijn gebaseerd op de reistijden van elektromagnetische impulsen tussen een zender en een ontvanger en kunnen voornamelijk worden ingedeeld in twee grote klassen, namelijk *mobile-terminated* (MT) en *mobile-originated* (MO) technieken (Renso, et al., 2007).

De meeste *satellite-based* technieken gebruikt binnen LBS, met inbegrip van de GPS techniek die we hier gaan bespreken, behoren tot de MT familie. Hierbij zijn de zenders geïnstalleerd aan boord van de *satellite vehicles* (SV) en behoren de ontvangers toe tot de mobiele gebruikers. Deze ontvangers zijn dikwijls ook uitgerust met bijvoorbeeld GSM technologie, zodat deze informatie kunnen doorsturen naar allerlei *service centers*. In deze klasse is de locatie informatie ook effectief beschikbaar voor de gebruiker, noodzakelijk voor allerlei doeleinden.

In de *mobile-originated* familie is het de gebruikerapparatuur die uitgerust wordt met zenders. Deze vorm van positionering wordt meestal gebruikt binnen *search and rescue* diensten, SARSAT (*Search And Rescue Satellite-Aided Tracking*) satellite services genaamd. Hierbij gaat de gebruikersapparatuur signalen uitzenden naar de satellieten (SV), welke deze informatie doorsturen naar de bevoegde diensten.

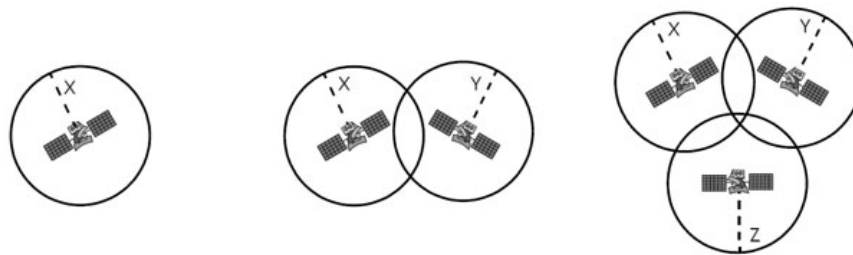
### 3.2.2.1 *Global Positioning System*

Het *Global Positioning System* (GPS) is het wereldwijde *satellite-based* radio-navigatiesysteem ontwikkeld door het Amerikaanse Ministerie van Defensie en momenteel ook het enige *Global Navigation Satellite System* (GNSS) beschikbaar. Het systeem is gebaseerd op 24 satellieten (SV) die, gelijk verdeeld in zes *orbital planes* op een hoogte van 20200 km, rond de aarde cirkelen (Zeimpekis, Giaglis & Lekakos, 2003). Deze banen zijn zodanig ontwikkeld dat we van op elke plaats ter aarde steeds vier satellieten kunnen waarnemen. Zoals reeds vermeld is het GPS project een *mobile-terminated* implementatie waarbij de zenders aanwezig zijn op de SV's. Hierdoor zal de informatie (positionering, snelheid, enz.) enkel beschikbaar zijn op het gebruikersniveau (Renso, et al., 2007).

Een satelliet zendt continu navigatiesignalen uit welke steeds een zeer precieze tijds melding bevatten, daar elke satelliet uitgerust is met meerdere atoomklokken. Een GPS ontvanger bepaalt zijn positie aan de hand van de afgeleide reistijden van de radiogolven verzonden door de constellatie van satellieten en de posities van deze satellieten. Om deze satellietpositie te kunnen berekenen verstuurd de satelliet twee types van informatie (Mattsson, 2001). Het eerste type is '*almanac*' genaamd en bevat informatie betreffende de baan van de satelliet en haar benaderende locatie. '*Ephemeris*' is het tweede type van informatie. Daar elke satelliet kleine afwijkingen ondervindt van haar geplande baan, worden deze steeds in de gaten gehouden van op de aarde. Deze monitors zenden vervolgens correcties naar de SV's betreffende hun baan, snelheid, hoogte en locatie. Beide types van informatie laten de ontvanger vervolgens toe de exacte positie van de SV's te bepalen.

Gebaseerd op het TOA principe (*trilateration*), kan de ontvanger de afstanden tussen de satellieten en zichzelf afleiden, waardoor ze met behulp van een mathematisch model haar positie kan berekenen. De ontvanger gebruikt steeds metingen met betrekking tot vier satellieten of meer aangezien we met vier onbekenden te maken hebben, namelijk positie ( $\mathbf{X}$ ,  $\mathbf{Y}$  en  $\mathbf{Z}$ ) en de klokfout. Deze fout is een gevolg van een foutieve synchronisatie tussen de satellietklok en die van de ontvanger. Essentieel komt de positiebepaling, zonder rekening te houden met de foutenbronnen, neer op het volgende: aangezien we weten op welk tijdstip het signaal werd verzonden ( $t_i$ ) en ontvangen ( $tr_i$ ),

kunnen we de reistijd hiervan berekenen ( $tr_i - t_i$ ), dit proces is *timing* genaamd. De index  $i$  staat voor de waarde van de gerelateerde satelliet, gaande van 1 tot het gebruikte aantal. Veronderstellend dat het signaal wordt verzonden aan de snelheid van het licht ( $c$ ), kan de afgelegde afstand van het signaal ( $p_i$ ) berekend worden als  $(tr_i - t_i) c$  (Misra en Enge, 2006). Op basis van deze afstanden kan de GPS ontvanger zichzelf positioneren. In essentie komt het er echter op neer dat we aan de hand van de afstand tot een satelliet, wetende dat de lichtsnelheid constant is en bovendien in alle richtingen even groot, de ontvanger op een denkbeeldige bol kunnen plaatsen, waarbij de satelliet het middelpunt vormt. Aan de hand van de afstand tot een tweede satelliet kunnen we een tweede dergelijke bol creëren. Dit beperkt de mogelijke positie van de ontvanger tot een cirkel, de snijlijn van de twee bollen. Als we hierbij een derde bol voegen, wordt de positie nog verder beperkt tot maar twee mogelijke punten (Figuur 3-7). Slechts één van deze twee punten kan zich op een positie in de buurt van het aardoppervlak bevinden, hierbij zorgt de vierde satelliet voor de oplossing (Bulusu, Heidemann & Estrin, 2000).



**Figuur 3-7** GPS plaatsbepaling met van één, twee of drie satellieten (D’Roza en Bilchev, 2003)

Elke GPS satelliet zendt signalen uit op twee frequenties:  $L_1 = 1,575.42$  MHz en  $L_2 = 1,227.60$  MHz. Hierbij is de  $L_1$  frequentie bedoeld voor zowel de consument als voor militaire- en andere speciale toepassingen, terwijl de  $L_2$  frequentie enkel bedoeld is voor dit laatste. In de nabije toekomst zal over de  $L_2$  frequentie een nieuw publiek signaal worden uitgezonden (L2CS), wat zou moeten leiden tot een hogere precisie en betere beschikbaarheid. Naast de positie kunnen GPS ontvangers nog twee andere belangrijke waarden meten, namelijk snelheid en tijd, waardoor de ontvanger ook gebruikt kan worden als een snelheidsmeter of een tijdsreferentie-instrument (Renso, et al., 2007).

Voor *low-profile* consumententoepassingen biedt het GPS systeem de volgende prestaties betreffende accuraatheid:

- *Positionering.* 32 m
- *Snelheid.* 0.1 m/s
- *Timing.* 1  $\mu$ s
- *Time to first fix (TTFF).* 1 min

De TTFF parameter is de tijd verstreken tussen het opzetten van de ontvanger en de beschikbaarheid van de initiële positiebepaling. Indien positie informatie onmiddellijk beschikbaar moet zijn, zoals bij LBS via mobiele telefoons, vereist dit de assistentie van andere technologieën (zie *Network Assisted GPS*, sectie 3.2.1.5) (Renso, et al., 2007). Deze prestaties worden beïnvloedt door verschillende foutenbronnen (Misra en Enge, 2006), zoals onder andere:

- *Atmosferische effecten.* Inconsistenties van de atmosferische condities beïnvloeden de snelheid waarmee de GPS signalen worden verstuurd.
- *Meerwegontvangst.* Wanneer het signaal reflecteert tegen bijvoorbeeld gebouwen, zal het de ontvanger via meerdere wegen bereiken. Het effect is minder kritiek bij bewegende objecten, de fout wordt dan immers onmiddellijk hersteld.
- *Klokfout.* Gezien de foutieve synchronisatie tussen de satellietklok en die van de ontvanger, vereist een nauwkeurige *timing* van het signaal een vierde satelliet.

Hoewel deze problemen individueel zeer miniem lijken, oefenen ze samen toch een niet te miskennen invloed uit op de kwaliteitsparameters. Als laatste is het nog belangrijk te vermelden dat er momenteel diverse gelijkaardige *satellite-based* systemen in ontwikkeling zijn (Glonass, Galilio, enz.). Deze worden hier echter niet besproken.

### 3.2.3 *Tracking-Methodes in de Praktijk*

De net besproken positioneringstechnieken bieden ons *real-time* posities van mobiele gebruikers binnen het netwerk op bepaalde tijdstippen. Positionering, zoals eerder vermeld, kan gebeuren aan de hand van vaste tijdsintervallen of gebaseerd op bepaalde gebeurtenissen. In sommige gevallen echter, zoals *movement-based services*, bieden deze posities niet voldoende informatie en zijn we eerder geïnteresseerd in trajecten. Wat we willen is dus eigenlijk een opeenvolging in de tijd van individuele posities van de



gebruikers, *tracking* genaamd, en de accuraatheid van deze trajecten is afhankelijk van zowel de frequentie van positionering (*sampling rate*) als de accuraatheid van de gebruikte techniek. In deze paragraaf bespreken we kort enkele *tracking*-methodes opgesplitst naar de eerder besproken ondersteunende netwerkinfrastructuren (Renso, et al., 2007).

Gebaseerd op de *cellular-based* netwerken, zouden we bijvoorbeeld een gebruiker zijn traject kunnen afleiden uit alle communicatiesignalen tussen de gebruiker en de infrastructuur. Hoewel deze de meest accurate *cellular-based* techniek beschikbaar is, heeft ze toch twee grote nadelen. Ten eerste zou dit het netwerk sterk overbelasten en ten tweede kunnen we ons enkel beperken naar het gebruikersaantal en niet in de geografische ruimte. Dit laatste is niet wenselijk indien we bijvoorbeeld voor een bepaalde *movement-aware* applicatie enkel gegevens van een bepaalde regio willen analyseren. Een andere techniek is gebaseerd op de facturatie informatie. Namelijk, binnen het netwerk zijn data beschikbaar met betrekking tot wanneer een bepaalde gebruiker een telefoon verrichtte, de datum, tijd en de duur ervan, evenals de cel waar het begon en waar het eindigde. Hoewel we ons hier echter wel kunnen beperken naar regio, moeten we onder andere inboeten aan accuraatheid van de afgeleide trajecten. Deze kan echter wel verbeterd worden door het in rekening nemen van allerlei geografische achtergrondinformatie, zoals de netwerktopologie.

Andere vormen van data die eveneens verzameld worden door de *provider*, en nuttig kunnen zijn voor verder gebruik, zijn onder andere statistische data betreffende de 'celdensiteit', bijvoorbeeld het aantal gebruikers actief in een bepaalde cel op een gegeven tijdstip.

Accuraatheid in *satellite-based* positioneringmethodes is zeer hoog vergeleken met de meeste *cellular-based* methodes. Hoewel dit zou resulteren in meer accurate trajecten, gaan we in de MT implementaties te maken krijgen met het nadeel dat de data worden verzameld op het MS niveau. Dit maakt het noodzakelijk een verdere connectie te voorzien naar een centrale server, die deze informatie zou aggregeren, alvorens deze informatie beschikbaar en bruikbaar is.

Op het vlak van de huidige draadloze *ubiquitous* toepassingen spannen LBS de kroon. Als gevolg hiervan kunnen positioneringstechnologieën beschikken over steeds meer

technologische ondersteuning en verbeteringen binnen het draadloze netwerk. *Tracking* technologieën daarentegen genieten slechts weinig ondersteuning, alhoewel dit zou kunnen veranderen door de mogelijkheden van de nieuwe generatie *movement-aware* toepassingen.

### 3.2.4 Privacy

Een belangrijke kwestie die onze aandacht verdient wanneer we kijken naar allerhande positioneringstechnieken van mobiele gebruikers is de individuele privacy. Haar belang binnen de sociale acceptatie van *ubiquitous computing* technologieën werd reeds uitvoerig besproken, maar kan echter niet genoeg benadrukt worden. Wanneer we over de mogelijkheid beschikken om mobiele *devices* te positioneren, betekent dit dan het ontstaan van een '*big brother is watching you*' maatschappij? Hoe ver kunnen we gaan vooraleer we belangrijke integriteitgrenzen overschrijden?

Deze bedenkingen ontstaan doordat deze positioneringstechnologieën over de mogelijkheid beschikken enorme hoeveelheden aan persoonlijke locatiedata te verzamelen, welke kunnen worden bewaard en gedeeld met externe partijen zonder de toelating van de gebruiker. Hierdoor is het belangrijk dat de gebruiker zelf regels kan bepalen betreffende waar en wanneer zijn apparaat kan worden gepositioneerd, en wat er met deze data naderhand gebeurt. Hierbij is het belangrijk op te merken dat er niet zoiets bestaat als 'goede' of 'slechte' data. Het is eerder van belang hoe deze data worden verwerkt en gebruikt (Pedreschi, et al., 2007).

Anonimisatie van data alvorens ze te verspreiden zou een manier kunnen zijn om privacybedreigingen te voorkomen. Echter, dit is onvoldoende daar *spatiotemporal* trajecten moeiteloos kunnen worden verbonden met individuen door gebruik te maken van publiek beschikbare informatie zoals het thuis- en werkadres (Bonchi, et al., 2007). Er bestaan vervolgens legio methoden en wetgevingen, zoals de *European Union Data Protection Directives*, om dataprivacy op een effectievere manier te garanderen. Deze zullen we hier echter niet verder behandelen, voor meer informatie kan u evenwel terecht in het werk van Pedreschi et al. (2007).

# *Hoofdstuk 4*

## *Het Modelleren en Bewaren van Bewegende Objecten*

De grote hoeveelheid aan *spatiotemporal* data aangaande bewegende objecten die moderne *location-* en *movement-aware* toepassingen ons biedt, moet ofwel snel verwerkt worden voor *real-time* doeleinden zoals *traffic control management* of *location-based services*, ofwel onderworpen worden aan allerlei *knowledge discovery* algoritmes (Orlando, Orsini, Raffaetà en Roncanto, 2007). Eender welke verwerking wordt vereist, het bewaren van deze data is essentieel.

*Spatiotemporal* databases zijn het resultaat van de aggregatie van tijd en ruimte in een gezamenlijk onderzoekskader (Macedo, et al., 2007). In dit hoofdstuk beschouwen we binnen dit kader managementkwesties betreffende databases ter ondersteuning van *spatial objects*, welke hun positie continu veranderen in de tijd, *Moving Objects Databases* (MOD) genaamd. Door bewegende objecten en hun trajecten te belichamen in computersystemen aan de hand van datamodellen, willen we niet enkel de huidige evenals historische locaties bewaren, maar evenzeer *location-aware queries*, evenals allerhande *knowledge discovery* algoritmes (zie Hoofdstuk 5) kunnen ondersteunen.

### *4.1 Reconstructie en Representatie van Trajecten*

Vooraleer we echter bewegende objecten en hun resulterende trajecten fysiek kunnen vertegenwoordigen in databases, hebben we nood aan trajectdata modellen die beschrijven hoe deze data zijn gestructureerd. Hiervoor is het echter noodzakelijk dat we eerst de basisconcepten behandelen betreffende de reconstructie- en representatie van de trajecten van bewegende objecten.

De eerder besproken positioneringstechnologieën (zie sectie 3.2) bieden ons *real-time* posities van mobiele gebruikers binnen het netwerk op bepaalde tijdstippen. In verschillende toepassingsdomeinen (bijvoorbeeld *movement-based services*, transportmanagement, stadsplanning en toerisme) bieden deze individuele posities echter niet voldoende informatie en zijn we eerder geïnteresseerd in beweging, of met andere woorden in trajecten van de gebruiker (Alvares, et al., 2007). Wat we willen is dus eigenlijk een opeenvolging in de tijd van individuele posities van de gebruikers (Renso, et al., 2007).

Hoewel een traject vrij simpel kan worden gedefinieerd als een opeenvolging van geografische posities gelinkt aan de tijd, is haar reconstructie, representatie en manipulatie eerder complex (Macedo, et al., 2007). Een belangrijk aspect van continue beweging is immers dat deze onmogelijk kan worden gevat in haar volledigheid. Dit zou enkel mogelijk zijn moesten we de gebruiker zijn positie kennen voor elke instantie in de tijd (Pfoser en Theodoridis, 2000). Dit maakt dat vooraleer we een traject kunnen representeren aan de hand van een geschikte curve, de beweging namelijk moet worden gereconstrueerd uit een verzameling van steekproefwaarnemingen (individuele, gekende posities). Dit proces noemen we interpolatie. Formeel kunnen we een traject  $T$  beschouwen als de grafiek resulterende uit de continue afbeelding van de tijd in het 2D  $XY$ -vlak van de ruimte. Dit kunnen we voorstellen als volgt (Macedo, et al., 2007):

$$I \subseteq IR \rightarrow IR^2 : t \mapsto \alpha(t) = (\alpha_x(t), \alpha_y(t))$$
$$\text{Nu } T = \{(\alpha_x(t), \alpha_y(t), t) \mid t \in I\} \subset IR^2 \times IR$$

Hierbij is het belangrijk op te merken dat een traject nooit uniek is, er kunnen namelijk steeds een oneindig aantal mogelijke trajecten worden verbonden met de verzameling steekproefwaarnemingen. Hoewel we aan de hand van interpolatie echter de meest geschikte curve proberen te vinden die deze punten met elkaar verbindt, is deze slechts een benadering van de werkelijk waargenomen beweging. De hieruit resulterende onzekerheid, inherent verbonden met de bekomen trajecten, wordt interpolatie-onzekerheid genoemd (Macedo, et al., 2007).

Als illustratie bekijken we lineaire interpolatie, de snelste maar tevens ook eenvoudigste in zijn soort. Bij deze methode vormen de steekproefwaarnemingen de eindpunten van

de lijnsegmenten van een polylijn, waarbij de punten worden verbonden met rechte lijnen (Pfooser en Theodoridis, 2000). De lineariteit komt hierbij tot uiting doordat gelijke sprongen in de tijd (tussen de steekproefwaarnemingen) leiden tot gelijke sprongen in de ruimte (Macedo, et al., 2007; Gülsen, T., 2005). Het segment tussen de punten  $(\mathbf{x}_i, \mathbf{y}_i, t_i)$  en  $(\mathbf{x}_{i+1}, \mathbf{y}_{i+1}, t_{i+1})$  wordt dan gegeven door  $(\mathbf{x}_i, \mathbf{y}_i, t_i) + ((t - t_i)/(t_{i+1} - t_i))(\mathbf{x}_{i+1} - \mathbf{x}_i, \mathbf{y}_{i+1} - \mathbf{y}_i, t_{i+1} - t_i)$ . Het traject bestaat dan vervolgens uit een concatenatie van al deze segmenten. Hoewel deze methode snel en eenvoudig is, bezit ze toch enkele nadelen. Zo veronderstellen we een constante snelheid en richting binnen één enkel segment en zijn de veranderingen in snelheid en richting na elke steekproefwaarnemingen dikwijls abrupt en discontinu (Macedo, et al., 2007). Desondanks zullen we later merken dat deze methode evenwel frequent voorkomt binnen trajectdata modellering (sectie 4.2).

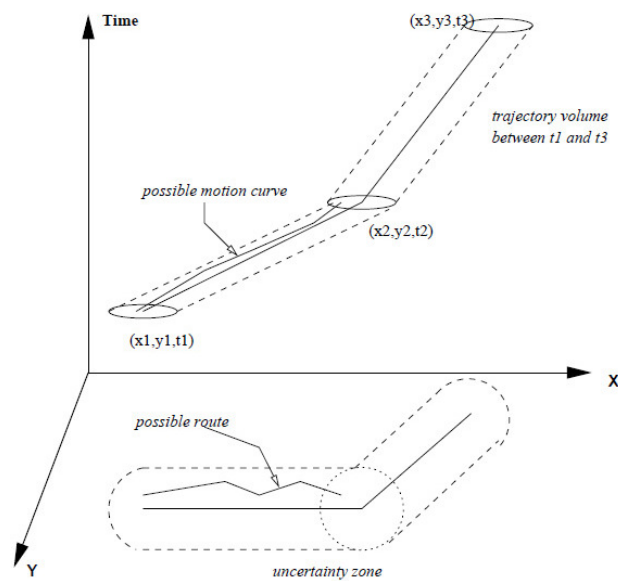
Door het bekomen traject vervolgens grafisch voor te stellen (zie Figuur 4-1 ter illustratie), creëren we een soort van conceptuele datarepresentatie. Deze trajectrepresentatie vormt een abstractie van de realiteit en reflecteert de hoofdeigenschappen van het bewegende object (Rinzivillo, et al., 2007). Betreffende trajecten stelt deze representatie een verzameling van metingen of observaties voor van bewegende objecten met zowel een tijds- als plaatsdimensie. Deze conceptuele trajectrepresentatie staat bovendien los van de fysieke implementatie van het traject in databases.

### 4.1.1 Omgaan met Locatieonzekerheid

Naast de eerder vernoemde interpolatieonzekerheid (sectie 4.1) is de accuraatheid van de bekomen trajecten bovendien eveneens afhankelijk van zowel de frequentie van positionering (*sampling rate*), als de accuraatheid van de gebruikte positioneringstechniek (Renso, et al., 2007). Aan de bekomen trajecten is dus een zekere inherente onzekerheid gekoppeld, welke afhankelijk van de bron meting- of interpolatieonzekerheid wordt genoemd (Macedo, et al., 2007). In de praktijk bestaan er vervolgens legio methodes om deze inherente onzekerheid verbonden met trajecten te managen binnen de reconstructie en/of representatie ervan, we bespreken er hier twee van.

In het model van Trajcevski, Wolfson, Zhang en Chamberlain (2004) wordt er gesproken over een *uncertainty threshold*  $r$ , welke wordt geassocieerd met het traject en beide

vormen van onzekerheid combineert. Deze *threshold* drukt de maximale afstand uit dat een object kan verwijderd zijn van de aangenomen positie binnen het traject, rekeninghoudend met beide vormen van onzekerheid. Het model kent aan elk punt van het geïnterpoleerde traject een *disk* toe (*uncertainty area*), parallel met het **XY**-vlak, met een radius  $r$  en middelpunt  $(x, y, t)$ , welke de verwachte positie voorstelt op tijdstip  $t$ . Het model gaat dus de onzekerheid expliciteren door het traject te representeren als een cilindrisch volume in 3D (Figuur 4-1).

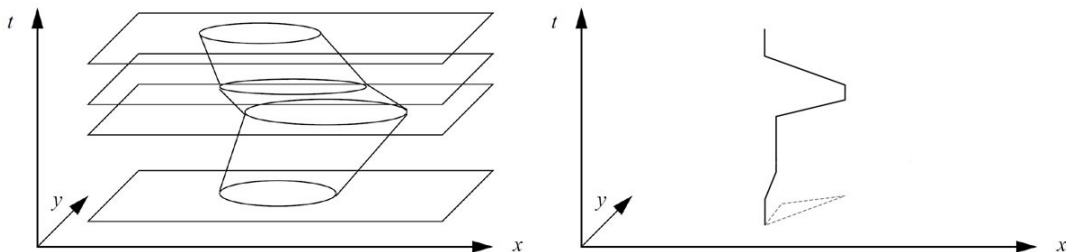


**Figuur 4-1** Een traject als een cilindrisch volume in 3D (Trajcevski, et al., 2004)

Het is eveneens interessant op te merken dat in het geval van netwerkbeperkte beweging (bijvoorbeeld autowegen of treinsporen), de trajectonzekerheid kan worden ingeperkt door rekening te houden met de onderliggende netwerktopologie (Güting en Schneider, 2005). Dit geldt zowel voor de reconstructie als de representatie ervan.

## 4.2 Trajectdata Modellen

Trajectdata betreffende bewegende objecten veranderen continu aangezien er voortdurend updates van hun locaties plaatsvinden. Deze karakteristiek brengt dan ook uitdagingen met zich mee in verband met de modellering van de trajecten van bewegende objecten, aangezien de meeste conventionele datamodellen immers statisch van nature zijn (Chakka, Everspaugh en Patel, 2003). In het werk van Macedo, et al. (2007) kunnen we lezen dat er drie categorieën bestaan van mogelijke benaderingen om trajecten van bewegende objecten te modelleren: *spatiotemporal data models*, *constraint data models* en *moving objects data models*. Vertrekkende van de vaststelling dat deze eerste twee niet bestemd, evenals volkomen geschikt zijn voor het modelleren van trajecten, hoewel hier evenwel gebruikt voor kunnen worden, focussen we ons in deze sectie op de ideeën achter de *moving objects data models* waarbij trajecten worden gemodelleerd als bewegende objecten. Indien we vervolgens enkel geïnteresseerd zijn in tijdsafhankelijke locaties (mensen, vliegtuigen, enz.) dan wordt *moving points* de gebruikte basisabstractie, wanneer we echter ook belang hechten aan de tijdsafhankelijke vorm en uitgestrektheid (orkanen, bossen, enz.) spreken we over *moving regions* (Figuur 4-2) (Güting, et al., 2000).



**Figuur 4-2 (L) Moving region, (R) moving point** (Güting en Schneider, 2005)

Een eenvoudige methode die frequent wordt gebruikt om *moving points* te modelleren is *point-location* management (Wolfson, 2002). Hierbij gaan we periodiek een locatie-tijd punt genereren van de vorm  $(l, t)$ , waarbij  $l$  een coördinatenpaar  $(x, y)$  kan voorstellen, en  $t$  de tijd. Deze punten worden bewaard in een database waarbij een querytaal wordt gebruikt om de locatie-informatie terug te halen. Deze methode heeft echter enkele

geciteerde nadelen zoals onder andere het gebrek aan het gebruik van inter- of extrapolatiemethoden (Wolfson, 2002).

Vanuit de *Moving Objects Databases* (MOD) literatuur zijn er hoofdzakelijk twee benaderingen gegroeid om trajectdata te modelleren. Één aanpak voor het modelleren van huidige en toekomstige posities en één voor historische trajecten (Güting en Schneider, 2005; Macedo, et al., 2007). De aanpak betreffende het modelleren en queryen van huidige en toekomstige posities moet rekening houden met het locatie-management probleem, waarbij we de continu veranderende posities van verschillende entiteiten (*moving points*) moeten managen in databases (Güting en Schneider, 2005). Dit vereist namelijk het maken van een zekere afweging. Frequente updates zouden enerzijds leiden tot zeer accurate locatie-informatie, maar dit zou echter resulteren in een te grote *update load*, wat het onmogelijk maakt voor een groot aantal objecten. Vanuit dit probleem is het *Moving Object Spatio-Temporal* (MOST) datamodel ontstaan (gecreëerd door Wolfson en collega's). Het basis idee binnen dit model is dat we een bewegend object niet opslaan op een directe manier aan de hand van zijn posities, maar eerder via zijn bewegingsvector. De locatie van het object wordt dus bepaald aan de hand van een lineaire functie in de tijd, welke wordt gespecificeerd aan de hand van twee parameters, namelijk zijn initiële positie  $\mathbf{y}_0$  en snelheidsvector  $\mathbf{v}$  (Macedo, et al., 2007). Hierdoor kunnen we zonder evenwel expliciete updates uit te moeten voeren, zijn positie continu berekenen aan de hand van lineaire interpolatie:  $\mathbf{y}(t) = \mathbf{y}_0 + \mathbf{v}(t - t_0)$  met  $t > t_0$ . In het datamodel worden de bewegingsvectoren weergegeven als een dynamisch attribuut, welke zijn waarde automatisch verandert met de tijd (Güting en Schneider, 2005). Een update is nu enkel nog nodig indien de snelheidsvector parameter  $\mathbf{v}$  van het bewegende object verandert, *dead-reckoning* genaamd. Dit dynamisch attribuut resulteert eveneens in *queries* wiens resultaten veranderen met de tijd, *continuous queries* genaamd en maakt het mogelijk uitspraken te doen over de toekomst. Hiervoor werd de *Future Temporal Logic* (FTL) querytaal ontwikkeld. Daar we een lineaire functie gebruiken gaat dit model echter wel minder goed presteren in geval van willekeurige mobiliteitspatronen.

Naast huidige en toekomstige beweging, bestaan er ook mobiliteitstoepassingen gebaseerd op informatie afgeleid uit historische trajecten van bewegende objecten, denk bijvoorbeeld maar aan stadsplanning. Voor deze toepassingen moeten we bijvoorbeeld evoluties onderzoeken of nagaan wanneer bepaalde relaties worden vervuld (Macedo, et



al., 2007). De Hiervoor ontwikkelde benadering betreffende het modelleren en *queryen* van historische trajecten wordt hier echter niet besproken, voor meer informatie kan u evenwel terecht in het werk van Macedo et al. (2007).

Aangezien een bespreking van alle mogelijke *moving objects data models* ons te ver zou leiden, zouden we ons hier echter willen beperken tot een vermelding van hun belangrijkste overeenkomsten. Als we veronderstellen dat een traject is gemodelleerd tot op de fijnste *spatial* granulariteit (exacte locatie), gaan deze datamodellen namelijk ondersteuning bieden op twee niveaus (Macedo, et al., 2007). Vooreerst bieden ze mechanismes waarmee je trajecten kan opsplitsen in subpaden aan de hand van allerlei variabelen, en ten tweede gaan de MOD modellen meestal gebruik maken van lineaire interpolatie tussen de steekproefwaarnemingen, welke de exact gekende locaties vormen.

#### *4.2.1 Een Semantisch-Gebaseerd Trajectdata Model*

Hoewel voorgaande modellen ons toe laten de posities van het *moving object* binnen zijn traject te modelleren en te representeren, is het systeem zich echter niet bewust van de onderliggende semantiek van het traject, terwijl deze achtergrondsemantiek net van fundamenteel belang is in de analyse van trajectdata binnen werkelijke toepassingen (Alvares, Bogorny, Macedo, Moelans en Spaccapietra, 2007).

In de ontwikkeling van een semantisch-gebaseerd conceptueel datamodel voor trajecten moeten we ons dan ook eerder focussen op relevante generische trajectkarakteristieken, onafhankelijk van een specifiek toepassingsdomein, zonder al te veel aandacht te besteden aan overwegingen in verband met de fysieke representatie van de trajecten in computersystemen (Macedo, et al., 2007). Conceptueel gezien heeft een traject namelijk twee facetten: een *spatiotemporal* facet betreffende de posities van het object in de tijd, maar eveneens een semantisch facet. Hiermee bedoelen we namelijk applicatierelevante karakteristieken van het traject, of subdelen hiervan (zie sectie 3.1.3). Bij het conceptueel modelleren van trajecten moet er bovendien ook rekening worden gehouden met allerlei trajectbeperkingen in verband met de relatie tussen haar subdelen evenals

met haar omgeving. We hebben het hier dan over allerlei semantische, topologische- en tijdgerelateerde beperkingen.

## 4.3 *Moving Objects Databases*

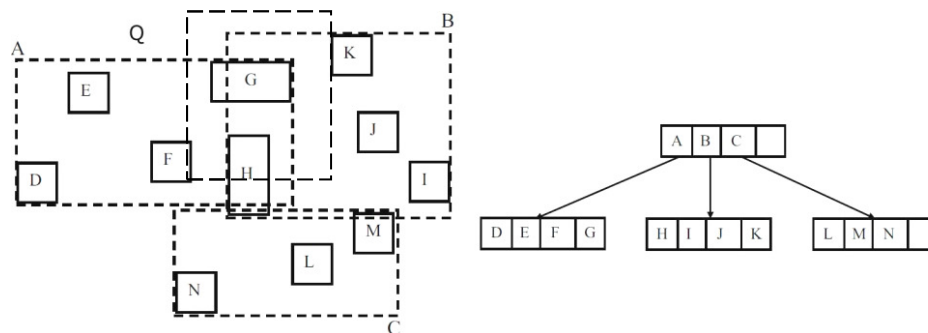
In deze sectie beschouwen we twee prominente managementkwesties betreffende *Moving Object databases* (MODs), namelijk trajectindexering en het omgaan met traject-onzekerheid, en bespreken we ook kort even de voordelen van Traject-Data Warehouses (TDWs) aangaande mobiliteitstoepassingen.

### 4.3.1 *Trajectindexering*

*Spatiotemporal* indexering wordt gebruikt door *spatiotemporal* databases aangaande *queries* te optimaliseren door objecten in een 3D ruimte zodanig te rangschikken dat ze efficiënt teruggevonden kunnen worden. Trajectindexeringmethodes kunnen worden opgedeeld in diegene welke netwerkbeperkte beweging, of beweging in een onbeperkte ruimte organiseren (Frentzos, Peleksis, Ntoutsis en Theodoridis, 2007). Daar de meeste hiervan gebaseerd zijn op, of een uitbreiding vormen van de *R-tree* methode, een *spatial* indexeringsmethode, beperken we onze bespreking tot deze *spatial* methode.

Een *spatial R-tree* is een hiërarchische datastructuur welke wordt gebruikt voor het dynamisch en efficiënt managen van een grote verzameling 2D geometrische data-objecten. De *R-tree* methode (Manolopoulos, Nanopoulos, Papadopoulos en Theodoridis, 2005) deelt de beschouwde 2D geografische ruimte, met inbegrip van haar geometrische objecten (vormen, lijnen en punten), op in hiërarchische, mogelijk overlappende, *Minimum Bounding Regions* (MBRs) en bewaart vervolgens de index *records* in de bladeren van de resulterende boom. Elk knooppunt van een *R-tree* bevat een variabel aantal *entries*. Een knooppunt-*entry* (niet-blad) is van de vorm **(*p*, *mbr*)**, waarbij ***p*** een verwijzing vormt naar een kind hiervan en ***mbr*** verwijst naar de *bounding rectangle* die al haar kinderen omvat. Een blad-*entry* is van de vorm **(*oid*, *mbr*)**, waarbij ***oid*** een

identificatie biedt voor het werkelijke database object en *mbr* verwijst naar de MBR die het object representeert. In een *R-tree* van de orde  $(m, M)$  heeft elk knooppunt (uitgezonderd enkele excepties) een maximale capaciteit van  $M$  entries, waarbij bovendien aan de beperking  $m \leq M/2$  moet worden voldaan. Hierbij is  $m$  de minimale knooppuntbezetting, meestal gelijk gesteld aan  $M/2$ , wat garandeert dat het ruimtegebruik minstens 50% is. Een knooppunt van de boom correspondeert bovendien met een *physical disk page*, dit is het fundamentele element waarop de fysieke schijfopslag is georganiseerd (Frentzos, et al., 2007). Figuur 4-3 illustreert een verzameling *spatial*-objecten gerepresenteerd aan de hand van hun MBRs ( $D \dots N$ ), de MBRs welke voorgaande objecten groeperen in knooppunt-entries ( $A$ ,  $B$  en  $C$ ) en de corresponderende *R-tree* veronderstellend dat  $m=2$  en  $M=4$ .



**Figuur 4-3 (L)** MBRs, **(R)** de corresponderende *R-tree* (Manolopoulos, et al., 2005)

De invoeging- en verwijdering algoritmes gebruiken deze MBRs om zo te verzekeren dat naburige objecten in dezelfde knooppunt-entries worden geplaatst. Objecten worden namelijk bij die MBR gevoegd die hierdoor de kleinste stijging van zijn grootte zal ondervinden. Zoekalgoritmes daarenboven gebruiken eveneens deze MBRs om te beslissen of ze al dan niet verder zoeken in een bepaald kind. Hierdoor worden de meeste knooppunten nooit aangeraakt binnen een *query*, wat de zoeksnelheid ingrijpend verhoogt (Manolopoulos, et al., 2005). Als voorbeeld van de verschillende soorten traject-zoekalgoritmes (*queries*) bespreek ik hier het (*spatial*) *R-tree range search* algoritme gebaseerd op Figuur 4-3 (Frentzos, et al., 2007; Manolopoulos, et al., 2005). Het *range*-zoekalgoritme start met een *search rectangle*  $Q$  en vertrekt bij de stam, checkend dat de verzameling MBRs (*entries*) van het knooppunt overlapt met  $Q$ . Wanneer zo een *entry*-MBR overlapt met  $Q$  ( $A$  en  $B$  in dit geval), volgt het algoritme de

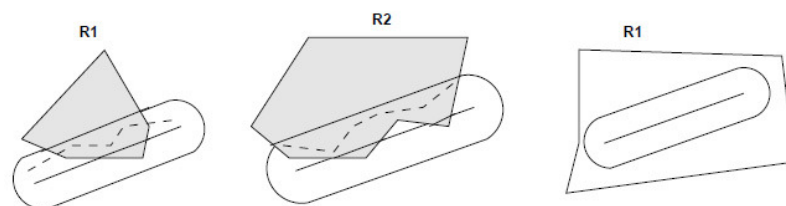
verwijzing van deze *entry* naar het corresponderende kind. Dit gebeurt op een recursieve manier totdat we een blad bereiken. Binnen dit blad worden de *entries* getest en hun *oid* vervolgens in de resultatenset geplaatst indien de *rectangles* overlappen (objecten **F** en **G** voor knooppunt **A** en **H** voor **B**). Een werkelijke zoekopdracht zou bijvoorbeeld kunnen zijn: "Zoek alle cafés in een omtrek van 2 km rondom mij". De uitbreiding van dit algoritme naar het *spatiotemporal* domein is een vrij eenvoudige taak, waarbij elke 2D MBR van de werkelijke objecten, knooppunten en *queries* wordt vervangen door haar 3D alternatief.

### 4.3.2 Het Managen van Locatieonzekerheid

In sectie 4.1 hebben we reeds gesproken over allerlei onzekerheden gekoppeld aan trajecten, welke maken dat de geregistreerde locaties van een bewegend object niet altijd precies haar werkelijke locaties weergeeft (Frentzos, et al., 2007). Deze locatieonzekerheid beïnvloedt verschillende MOD aspecten. In sectie 4.1.1 bespreken we reeds kort hoe deze onzekerheid wordt opgenomen in de reconstructie en representatie van bewegende objecten, in deze paragraaf bespreken we vervolgens hoe om te gaan met zoekopdrachten (*queries*) onder onzekerheid. Aangezien er echter legio methodes bestaan hieromtrent beperken we ons tot het model van Trajcevski et al. (2004), reeds gedeeltelijk aangehaald in sectie 4.1.1.

In dit werk (Trajcevski, et al., 2004) worden twee categorieën vermeld, met elk hun operatoren, aangaande het queryen van bewegende objecten onder onzekerheid, namelijk: *point queries* en *spatiotemporal range queries*, beiden verbonden met één enkel traject. *Point queries* verwijzen naar de locatie van het bewegende object op een specifieke instantie in de tijd, of naar de instantie(s) in de tijd op welke het object wordt verwacht zich op een specifieke locatie te bevinden. *Spatiotemporal range queries* zijn een uitbreiding van hun traditionele vorm rekening houdend met de aanwezige onzekerheid in de databaselocaties. Locatieonzekerheid beïnvloedt zowel de *temporal* als *spatial* aspecten van deze queries. Vervolgens kijken we naar de *spatiotemporal range queries* onder onzekerheid.

Traditioneel queryen *spatiotemporal range queries* naar de objecten binnen een bepaalde regio gedurende een bepaald interval. Aangaande het *temporal* effect betreffende locatieonzekerheid wijzigt dit naar de objecten wiens trajecten enige tijd (*sometime*), ofwel de hele tijd (*always*), aanwezig zijn in de *query*-regio  $R$  gedurende een bepaald tijdsinterval. Hierbij verwijst de 'always' situatie naar gevallen waarbij het traject binnen de *query*-regio is gelegen gedurende de volledige *query*-tijd  $[t_1, t_2]$ . Betreffende het *spatial* effect kunnen we 'queryen' naar de objecten die mogelijk (*possibly*), of met zekerheid (*definitely*) binnen de *query*-regio zijn gelegen. Waarbij 'mogelijk binnen' overeenstemt met situaties waarin slechts een gedeelte van de *uncertainty area* van het traject zich bevindt binnen de *query*-regio. Een combinatie van beide effecten leidt tot de volgende *spatiotemporal range query*-types onder onzekerheid: *Possibly-Sometime-Inside*, *Possibly-Always-Inside*, *Always-Possibly-Inside*, *Definitely-Always-Inside*, *Definitely-Sometime-Inside* en *Sometime-Definitely-Inside*. Ter illustratie bekijken we in Figuur 4-4 de 2D projecties van *Possibly-Sometime-Inside*, *Possibly-Always-Inside* en *Definitely-Always-Inside*. Het verschil bevindt zich erin dat in het *Definitely-Always-Inside* geval **(3)** de volledige *uncertainty zone* in  $R$  is gelegen gedurende het tijdsinterval, ten opzichte van slechts één mogelijke route in het *Possibly-Always-Inside* geval **(2)**. De opsplitsing naar 'always' en 'sometime' en is te wijten aan het feit dat de mogelijke route in het *Possibly-Sometime-Inside* geval **(1)** de *query*-regio  $R$  slechts kruist tijdens het volledige tijdsinterval en er niet volledig in is gelegen zoals in het *Possibly-Always-Inside* geval **(2)**.



**Figuur 4-4** **(1)** *Possibly-Sometime-Inside*, **(2)** *Possibly-Always-Inside*, **(3)** *Definitely-Always-Inside* (Trajcevski, et al., 2004)

Trajcevski (2003) biedt ook een methodologie om *Probabilistic Range Queries* (PRQ) te beantwoorden onder onzekerheid. De behandelde *queries* zijn van de vorm: "Wat is de kans dat een gegeven bewegend object binnen een gegeven regio enige tijd/altijd

was/zal zijn gedurende een gegeven tijdsinterval?”. Hierbij wordt de kans gegeven door de fractie gelijk aan het intersectievolume tussen het trajectvolume en de *query*-regio, op het trajectvolume (Frentzos, et al., 2007).

### 4.3.3 Traject-Data Warehouses

In Inmon (1996) worden *data warehouses* omschreven als een onderwerpgeoriënteerde, geïntegreerde, tijdsvariante en niet-volatiele aggregatie van data ter ondersteuning van het management van het beslissingsproces. In een *data warehouse* worden data georganiseerd en gemanipuleerd aan de hand van een multidimensionaal datamodel dat data bekijkt in de vorm van een data kubus. Hierbij stelt elke dimensie een ander perspectief voor. Gezien de grote beloftes willen we in deze sectie kort ingaan op een speciale vorm van *spatiotemporal data warehouses*, Traject-Data Warehouses (TDWs) genaamd (Peleksis, et al., 2007).

TDWs worden gevoed door tijdsafhankelijke locatiedata in de vorm van trajecten. De motivatie erachter is het transformeren van rauwe trajectdata naar waardevolle informatie die kan worden gebruikt voor beslissingsdoeleinden in allerlei *ubiquitous* toepassingen. Immers, het grote volume aan trajectdata, haar complexe aard, samen met haar gespecialiseerde *query processing* eisen, maken het namelijk moeilijk hieruit waardevolle informatie te extraheren. Hierdoor willen we de traditionele aggregatietechnieken uitbreiden en de mogelijkheid bieden voor *Online Analytical Processing* (OLAP) operaties. Een veelvoud aan mobiliteitstoepassingen zouden voordelen kunnen halen uit deze technologie, ter illustratie bekijken we het wegverkeerssupervisie toepassingsdomein, waarbij we gespecialiseerde *location-aware services* bieden aan geregistreerde gebruikers binnen een stad. *Queries* die waardevol kunnen zijn voor analisten, beslissingsnemers evenals eindgebruikers zijn onder andere: “Hoeveel mobiele gebruikers bevinden zich in een bepaald district tijdens een gegeven tijdsinterval?” of “Wat is de snelste weg naar het dichtstbijzijnde hospitaal rekening houdende met de huidige verkeerssituatie?”. Hoewel deze *queries* eveneens zouden kunnen worden beantwoord aan de hand van *legacy* systemen, zou dit ten koste gaan van de *processing* vereisten en de nodige responstijd.

## *Hoofdstuk 5*

### *Mobiliteitskennis*

*Knowledge discovery* vindt vandaag de dag plaats in een des te meer kennisrijke en rekenkrachtige omgeving. Mobiliteitskennis, verborgen in de talrijk verzamelde en kennisrijke mobiliteitsdatarecords, biedt dan ook talloze mogelijkheden ter ondersteuning van allerhande *ubiquitous* transport- en mobiliteitstoepassingen. Vooraleer we ons echter kunnen richten op de *spatiotemporal knowledge discovery* aspecten en methoden hebben we nood aan een gedegen begrip van de aspecten en technieken betreffende haar *spatial* variant.

#### *5.1 Spatial Knowledge Discovery*

In deze sectie, die als fundament dient voor haar *spatiotemporal* variant, behandelen we eerst en vooral Geografische Informatie Systemen (GIS) voor het managen van geografische data, waarna we het hebben over *spatial feature extraction* teneinde *spatial* karakteristieken te extraheren, impliciet gecodeerd in deze data. We eindigen vervolgens met de aspecten en technieken betreffende *spatial data mining*. In verband met deze sectie is het belangrijk op te merken dat 'geografisch' als een speciaal geval van '*spatial*' moet worden bekeken, daar het laatste betrekking kan hebben op elke ruimte met impliciete relaties tussen de aanwezige dataobjecten en niet enkel specifiek betrekking heeft op het aardoppervlak, zoals in het geografische geval (Miller, 1991).

### 5.1.1 Geografische Informatie Systemen

Indien we werken met *spatial* data betreffende transport en mobiliteit, zijn we onlosmakelijk verbonden met Geografische Informatie Systemen (GIS) teneinde het bewaren, manipuleren, managen, analyseren, modelleren en representeren van deze data (Rinzivillo, et al., 2007). Hoewel de laatste jaren GIS vrijwel geëvolueerd zijn van systemen tot een 'wetenschap', een complexe interactie van theorie, technologie en systemen, blijft het omgaan met *spatial* data betreffende locaties op het aardoppervlak (geografische data) en het extraheren van kennis hieruit het hoofddoeleinde ervan. Aan de technologiezijde zijn GIS onder andere afhankelijk van GPS technologie, reeds besproken in sectie 3.2.2.1.

De data bewaard binnen GIS zijn bovendien typisch verdeeld over diverse thematische lagen, welke corresponderen met de verschillende thema's binnen de applicatie (Rigaux, Scholl en Voisard, 2001). Zo kunnen we een laag hebben met wegennetwerkinformatie, een tweede met informatie betreffende rivieren of allerhande andere topologische aspecten, enzovoort. Deze lagen worden vervolgens geïntegreerd aan de hand van een expliciete locatie op het aardoppervlak. De locatie- en attribuutdata waaruit deze lagen bestaan, kunnen conceptueel worden gemodelleerd als een continu veld, waarbij elk punt slechts één attribuutwaarde bezit (bvb. de hoeveelheid regenval) of gebaseerd op discrete objecten (bvb. een huis). Beide modellen kunnen vervolgens worden gerepresenteerd in het GIS via het tessellatie- of het vectormodel, zodoende dat de continue geografische ruimte wordt gediscretiseerd. In het tessellatiemodel wordt de ruimte verdeeld in cellen, die elk een waarde van het beschouwde attribuut bevatten, hierbij gaat alle variatie binnen een cel verloren. In het vectormodel daarentegen worden de geografische attributen gerepresenteerd aan de hand van een geometrische structuur, namelijk: een punt, polylijn of polygoon, welke respectievelijk een boom, straat of stad kunnen vertegenwoordigen.

De grote hoeveelheid aan multidimensionale geografische data vereist vervolgens dat de GIS functionaliteit beschikt over geïntegreerde *data mining* technieken evenals allerhande *spatial feature extraction* mogelijkheden, teneinde relevante kennis hieruit te kunnen extraheren. Deze aspecten worden vervolgens behandeld in de volgende secties.



### 5.1.2 *Spatial Feature Extraction*

In deze subsectie gaan we het kort hebben over *spatial feature extraction* en haar rol binnen het *spatial knowledge discovery* proces (Rinzivillo, et al., 2007). Vooreerst is het belangrijk op te merken dat een *spatial feature* kan worden beschouwd als een *spatial* karakteristiek van een geografisch object, impliciet gecodeerd hierin. Een geografisch object wordt daarenboven gekenmerkt door een *spatial* component, bvb. een geometrische figuur, samen met een verzameling attributen die zijn *non-spatial* dimensie beschrijven, bvb. het bouwjaar. Terwijl deze *non-spatial* informatie kan worden gequeryed op de traditionele manier, moeten de *spatial features* worden geëxtraheerd gedurende de *data preprocessing* of de *data mining* stap, daar deze zeer waardevolle informatie kunnen aanbrengen. Ter illustratie beschouwen we hier kort relatie- en aggregatiegebaseerde *spatial features*. Onder relatiegebaseerde *features* verstaan we informatie betreffende de relatie tussen twee of meer objecten, zoals hun onderlinge afstand, de richting ten opzichte van elkaar gebruikmakende van een referentiesysteem, of allerhande andere topologische relaties. Aggregatiegebaseerde *features* echter zijn het gevolg van data-aggregatie teneinde *features* te ontdekken die niet op één plaats kunnen worden gemeten, bvb. het aantal gezinnen met jonge kinderen in de regio van een bepaalde locatie, zodoende de geschikte locatie van een kinderdagverblijf te kunnen bepalen.

Daar bovendien veel *spatial features* natuurlijke geografische afhankelijkheden voorstellen, waarbij sommige objecten steeds gerelateerd zijn aan andere, kan deze stap efficiënter worden gemaakt door het opnemen van diverse soorten achtergrondkennis, vaak aanwezig in allerhande geografische databases. Hiermee wordt er onder andere beduidend minder tijd verspild betreffende het zoeken naar allerhande, reeds welgekende relaties.

### 5.1.3 *Spatial Data Mining*

In dit deel behandelen we de ideeën achter *spatial data mining*, samen met enkele technieken hierbinnen, toegepast op geografische data als basis voor haar *spatio-*

*temporal* variant (zie sectie 5.2). Vooraleer we ons echter toespitsen op de technieken is het belangrijk aandacht te besteden aan de uitdagingen die de geografische data ons stellen betreffende *data mining*, daar de karakteristieken van deze data enkele essentiële assumpties schenden (Rinzivillo, et al., 2007). De eerste assumptie hieromtrent is die van onafhankelijk en identiek verdeelde observaties (May en Saitta (Reds.), 2007). Daar de attribuutwaarden van *spatial* objecten namelijk sterker gecorreleerd zijn des te dichter deze objecten zich in elkaars nabijheid bevinden, beter gekend als *Tobler's Law*, wordt deze geschonden. Dit heeft daarenboven als effect dat algoritmes slechter zullen presteren indien zij deze autocorrelatie tussen de objecten negeren. Een tweede karakteristiek van geografische data is het feit dat de gevonden afhankelijkheden op kleine schaal niet noodzakelijk gelden op grotere schaal. Dit maakt dat de gevonden patronen afhankelijk zijn van de gekozen resolutie en onder invloed zijn van willekeurige variatie. Een derde uitdaging hieromtrent zijn de reeds besproken *spatial features*, oftewel de impliciet gecodeerde karakteristieken en relaties van de beschouwde objecten. Zoals eerder gesteld kunnen deze voor of tijdens de *data mining* stap worden geëxtraheerd.

In het algemeen zijn er twee benadering betreffende de manier waarop algoritmes geografische data behandelen (Rinzivillo, et al., 2007). Bij de eerste aanpak maken we gebruik van traditionele algoritmes samen met *spatial* attributen, opgenomen als variabelen of door gebruik te maken van *feature extraction* tijdens de *preprocessing* stap. De tweede methode maakt gebruik van speciale algoritmes die de *feature extraction* stap incorporeren of zelfs in staat zijn deze geografische karakteristieken en relaties rechtstreeks te behandelen. Vervolgens bespreken we enkele *spatial data mining* technieken.

### 5.1.3.1 *Clusteren, Classificatie en Regressie*

Clusteren is een techniek die we gebruiken om een verzameling van objecten op een bepaalde manier te verdelen in (niet)-overlappende groepen van gelijkaardige objecten zodat objecten uit een andere groep een zekere heterogeniteit vertonen (Witten en Frank, 2000). Aangezien clusteren in essentie gebaseerd is op afstanden tussen objecten

is deze inherent bruikbaar als *spatial* techniek (Rinzivillo, et al., 2007). Van deze techniek zijn er talloze varianten beschikbaar.

Vertrekkende vanuit een training set gaan vervolgens zowel de classificatie- als regressietechnieken de ongekende targetwaarde van een object voorspellen. Indien deze targetwaarde een discrete variabele is, spreken we over classificatie, regressie daarentegen heeft betrekking op continue variabelen (Witten en Frank, 2000). Ter illustratie bekijken we de *k-Nearest Neighbour* (kNN), instantiegebaseerde methode, waarbij ongekende instanties worden geclassificeerd afhankelijk van de targetwaarde (klasse) van de *k* meest gelijkaardige training instanties (Witten en Frank, 2000). In het geval van classificatie nemen we hiervoor de meest frequente klasse, bij regressie echter maken we gebruik van een (gewogen) gemiddelde hiervan. Om de gelijkaardigheid te expliciteren maakt de kNN methode vervolgens gebruik van een afstandsmaat. In het geval van geografische data kunnen we hiervoor gebruik maken van geografische coördinaten. Als gevolg hiervan gaat kNN gebruik maken van de *k* meest dichtstbijzijnde objecten en maken we dus impliciet gebruik van de eerder besproken positieve autocorrelatie tussen de *spatial* objecten en hun targetwaardes (Rinzivillo, et al., 2007).

### 5.1.3.2 *Associatieregels*

Associatieregels zijn van de vorm  $\mathbf{X} \rightarrow \mathbf{Y}$ , waarbij  $\mathbf{X}$  en  $\mathbf{Y}$  een verzameling voorstellen van attributen, samen voorkomend in een bepaalde *tuple* of rij van een dataset (Witten en Frank, 2000). De *support*  $\mathbf{s}$  van de associatieregule  $\mathbf{X} \rightarrow \mathbf{Y}$ ,  $\mathbf{s}(\mathbf{X} \cup \mathbf{Y})$ , komt overeen met het aantal instanties in de dataset waarbij beide attributensets  $\mathbf{X}$  en  $\mathbf{Y}$  voorkomen en de *confidence*  $\mathbf{c}$  is de *support*  $\mathbf{s}(\mathbf{X} \cup \mathbf{Y})$  uitgedrukt als een percentage van alle instanties welke voldoen aan  $\mathbf{X}$ , namelijk  $\mathbf{s}(\mathbf{X} \cup \mathbf{Y})/\mathbf{s}(\mathbf{X})$ .

*Spatial* associatieregels (SARs) zijn eveneens van de vorm  $\mathbf{X} \rightarrow \mathbf{Y}$ , echter hierbij stellen  $\mathbf{X}$  en  $\mathbf{Y}$  verzamelingen voor van predicaten, waarbij minsten één element een *spatial* predicaat is (een relatiegebaseerde *spatial feature*, zie sectie 5.1.2) (Koperski en Han, 1995). Een voorbeeld van een SAR met  $\mathbf{Y}$  een *spatial* consequent en  $\mathbf{X}$  een *non-spatial/spatial* antecedent zou kunnen zijn: ***is\_a(x, gas\_station) → close\_to(x, highway)***. In de bestaande SAR *mining* algoritmes is de *computational cost* vooral

gelegen in de *spatial feature extraction*, wat dan ook dikwijls reeds wordt uitgevoerd tijdens de *preprocessing* stap (Rinzivillo et al., 2007). Het probleem met zowel de *spatial* als *non-spatial* variant van deze *data mining* techniek is bovendien de generatie van grote hoeveelheden aan regels wat vermijdbaar is door het ex-ante specificeren van *support* en *confidence* minima (Witten en Frank, 2000).

## 5.2 *Spatiotemporal Knowledge Discovery*

Nanni, Kuijpers, Körner, May en Pedreschi (2007) wijzen ons erop dat het *spatiotemporal* onderzoeksveld betreffende *knowledge discovery* nog maar net in haar ontwikkelingsfase zit, waardoor we nog niet beschikken over een theoretisch onderzoekskader hieromtrent. Hierdoor willen we ons in dit deel dan ook vooral toespitsen op de ideeën en uitdagingen achter deze vorm van *knowledge discovery*, steunend op haar *spatial* variant. We behandelen vervolgens *spatiotemporal feature extraction* en *spatiotemporal data mining*.

### 5.2.1 *Spatiotemporal Feature Extraction*

Zoals reeds vermeld in sectie 5.1.2 is het *feature extraction* proces hoofdzakelijk gebaseerd op het onderzoeken van diverse relaties tussen de objecten in de dataset. In dit deel bekijken we de invloed van de extra *temporal* dimensie op dit proces (Rinzivillo, et al., 2007). Vooreerst kunnen we onderzoeken wat de invloed is van deze extra dimensie op de bestaande *spatial* relaties. Er bestaan namelijk tijdsinvariante relaties waarop de tijd geen invloed uitoefent, bijvoorbeeld de *Arc de Triomphe* zal steeds op de *Champs-Élysées* zijn gelegen, maar dit geldt natuurlijk niet voor alle relaties. Dit is dan ook slechts een startpunt, aangezien we de rol van tijd hier louter beschouwen als een *enabler/disabler* van *spatial* relaties. We moeten deze tijdsdimensie namelijk grondiger benutten. Wanneer een object beweegt in de tijd verandert immers niet enkel zijn positie, maar kan de omgeving eveneens veranderen in de tijd, wat evenzeer geldt voor de status van het object, enz. Alsook het gebruik van allerhande soorten achtergrondkennis om het *extraction* proces efficiënter te maken wordt moeilijker, daar deze nu van

een dynamische aard hoeft te zijn. Dit onderwerp komt nog deels aan bod in het *data mining* gedeelte (zie sectie 5.2.2).

## 5.2.2 *Spatiotemporal Data Mining*

In dit deel willen we ons hoofdzakelijk toespitsen op de ideeën en uitdagingen betreffende *spatiotemporal data mining* met de nadruk op het 'minen' van trajecten van bewegende objecten. De uiteenzetting in deze sectie is hoofdzakelijk gebaseerd op het werk van Nanni, et al. (2007), daar dit kan worden beschouwd als het eerste bredere, verkennende werk hieromtrent. Vooraleer we verder gaan is het echter belangrijk op te merken dat de mobiliteitscontext waarop we ons toespitsen allerhande extra uitdagingen met zich meebrengt. Zo handelen we steeds in een onstabiele omgeving (bvb. wegenwerken) en gaat het mobiele gedrag van individuen veranderen in de tijd. Hiervoor is het essentieel dat algoritmes over de mogelijkheid beschikken deze structurele veranderingen en nieuwe gedragspatronen te incorporeren (Nanni, et al., 2007).

Kennis geëxtraheerd uit trajectdata zou ons vervolgens kunnen bijstaan in diverse taken betreffende alledaagse transport- en mobiliteitsfenomenen zoals het detecteren van files, het voorspellen ervan en het ontdekken van relaties hiertussen (Nanni, et al., 2007). Daar files gepaard gaan met groepen van voertuigen gekenmerkt door een lage snelheid kunnen we gebruikmaken van diverse clustertechnieken betreffende filedetectie doeleinden. We kunnen eveneens gebruikmaken van clustertechnieken gebaseerd op relatieve snelheden om bijvoorbeeld een groep van objecten onder te verdelen in auto-, fiets- en voetgangersclusters. Het valt ook niet moeilijk in te beelden dat diverse files gerelateerd kunnen zijn betreffende hun tijdsdimensie. Deze relaties kunnen we vervolgens vastleggen in diverse associatieregels, zoals: ***file(Leuven, 7h30') → file(Brussel, 7h45')***. Classificatietechnieken bieden evenzeer mogelijkheden om bijvoorbeeld files te onderscheiden van normale verkeerssituaties. Trajectdata bieden bovendien evenzeer de mogelijkheid om sequentiële patronen te ontdekken van de vorm ***A → B → C***, wat betekend dat de situaties ***A***, ***B*** en ***C*** in een *temporal* sequentie voorkomen. Het ontdekken van *spatiotemporal* trends kan evenzeer nuttig zijn betreffende inzicht te verwerven in diverse verkeerssituaties, zoals: "de snelheid van bewegende objecten op de E313 verhoogd eenmaal afrit Lummen gepasseerd". Als

laatste beschouwen we de mogelijkheden van predicties van trajectdata, welke vragen zoals "Hoeveel trajecten zullen het klaverblad morgen om 16h00' kruisen?" kunnen helpen beantwoorden. Bovenstaande categorieën van *mining* technieken en toepassingen zullen vervolgens verkennend besproken worden in de volgende subsecties.

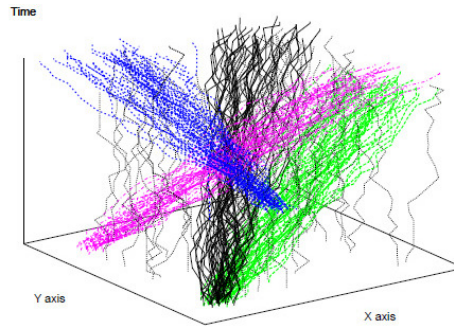
### 5.2.2.1 *Clusteren*

Clusteren in de context van trajecten van bewegende objecten komt in essentie neer op het groeperen van bewegende objecten met gelijkaardige gedragingen. Hiervoor hebben we twee benaderingen ter beschikking (D'Auria, Nanni en Pedreschi, 2006). In het eerste geval maken we gebruik van generische clusteralgoritmes door het hanteren van een bepaalde afstandsfunctie tussen trajecten. Dit impliceert dat de volledige trajectsemantiek wordt samengevat in deze afstandsfunctie. We kunnen daarentegen eveneens gebruikmaken van specifieke clusteralgoritmes betreffende trajectdata, waardoor de trajectsemantiek beter tot zijn recht kan komen. We zullen vervolgens beide benaderingen conceptueel toelichten.

Betreffende de afstandsgebaseerde clustermethoden voor trajecten gaan we afhankelijk van de soort clusters die we willen vormen een afstandsmaat tussen objecten ontwikkelen. Een zeer basis model betreffende het formuleren van een afstand zou kunnen zijn, die objecten als gelijkaardig te beschouwen welke ongeveer hetzelfde *spatiotemporal* traject afleggen, mogelijk gebaseerd op de Euclidische afstandsmaat. Dit met de bedoeling instanties van een populatie te identificeren die zich samen voortbewegen. Gebaseerd op ruimte-tijd coördinaten kunnen we dit grafisch illustreren aan de hand van Figuur 5-1, waarbij de verschillende kleuren vier diverse clusters voorstellen van samen voortbewegende instanties.

De zuiver afstandsgebaseerde methode bezit echter enkele inherente beperkingen betreffende zowel haar expressievermogen, een begrenzing naar de soort problemen die opgelost kunnen worden, als haar prestatieniveau. Dit is vooral te wijten aan het feit dat de afstandsfunctie vaak te beperkt de achterliggende semantiek, wat betreft de ongelijkheid, reflecteert (Nanni, et al., 2007). In tegenstelling tot het gebruik van een expliciete afstandsmaat tussen trajecten kan een trajectspecifiek clustermodel,

betreffende het voorgaande probleem, de gelijkaardigheid van trajecten bijvoorbeeld uitdrukken als het voorkomen van inter-traject 'ruis', waaraan een bepaald getolereerd frequentiequotiënt is verbonden (Nanni, et al., 2007). Hiermee zouden we dan evenzeer groepen kunnen identificeren die zich samen voortbewegen.



**Figuur 5-1** Voorbeeld van een afstandsgebaseerde clustermethode (D'Auria, et al., 2006)

### 5.2.2.2 *Spatiotemporal Patronen*

Data mining betreffende *spatiotemporal* patronen komt in essentie neer op het zoeken naar het herhaald voorkomen van interessante en precieze gedragingen van individuele of groepen van bewegende objecten (Nanni, et al., 2007). Hierbij zijn we voornamelijk geïnteresseerd in lokale patronen welke zich eerder focussen op subsecties van de *data-space*, zoals een subset van individuen of een beperkte regio in de ruimte. Dit is in contrast met de zogenaamde globale modellen, zoals cluster- of classificatiemodellen waarbij de nadruk hoofdzakelijk is gelegen op het karakteriseren van de complete *data-space*.

Betreffende *pattern mining* zijn er vervolgens twee benaderingen voorhanden, namelijk: *direct searches* en *inverse searches* (Nanni, et al., 2007). Hierbij is de nadruk omtrent het eerste vooral gelegen in het ontdekken van interessante patronen, terwijl we bij *inverse searches* eerder op zoek zijn naar het voorkomen van vooraf gespecificeerde patronen (wat, waar, wanneer en wie). De gebruikte algoritmes gaan bovendien evenzeer afhankelijk zijn van de gebruikte inputdata, welke in de mobiliteitscontext trajectdata kunnen zijn of hieruit geëxtraheerde informatie omtrent een individu, zoals

gebeurtenissen (bvb. de bezochte locaties) of allerhande dynamische aggregaten (bvb. de snelheid in functie van de tijd), of informatie betreffende een populatie, zoals filegegevens van een bepaalde dag (Nanni, et al., 2007). De reikwijdte van deze informatie (individu- of populatie-informatie) impliceert logischerwijze de potentiële reikwijdte van de hieruit geëxtraheerde patronen. Bovendien kunnen we indien we gebruikmaken van deze informatie als inputdata eveneens generische *pattern mining* algoritmes hanteren, dit wil zeggen niet specifiek bestemd voor *spatiotemporal* data. Ter illustratie bekijken we *frequent pattern mining*, een *direct search* methode, samen met een voorbeeld van *inverse searches*.

De basisbenadering betreffende *frequent pattern mining* die we hier bespreken is gebaseerd op *spatiotemporal feature extraction* (Nanni, et al., 2007). In essentie komt het erop neer dat we vooreerst een verzameling *features* extraheren uit de mobiliteitsdata. Dit levert ons onder andere gebeurtenissen, attribuutwaarden of, in het algemeen, *spatiotemporal* predicaten welke trajecten karakteriseren. Vervolgens gaan we generische *data mining* algoritmes toepassen op deze *featured data* teneinde onder andere frequente sequentiële patronen, trends of associatieregels te ontdekken. Op deze manier verzekeren we bovendien ex ante dat er rekening wordt gehouden met de onderliggende semantiek van de *spatiotemporal* data. In essentie is deze benadering bovendien gelijk aan die van de *spatiotemporal* associatie- en evolutieregels, waarvan deze laatste de *spatiotemporal* evoluties beschrijven van een object of groep van objecten.

Gezien voorgaande uiteenzetting zal de uitkomst van de *data mining* stap logischerwijze afhankelijk zijn van de *feature extraction* gedurende de *preprocessing* stap. Ter illustratie van *features* die mogelijk kunnen worden geëxtraheerd uit trajectdata betreffende bewegende objecten, bespreken we hier enkele individugebaseerde features (Nanni, et al., 2007):

- *Spatial en/of temporal aggregaten*. Bijvoorbeeld de lengte van het traject, de tijd gependend in de file of de gemiddelde snelheid van de dag.
- *Spatial gebeurtenissen*. Bijvoorbeeld de bezochte locaties.
- *Spatiotemporal gebeurtenissen*. Bijvoorbeeld uitgevoerde manoeuvres, plotse versnellingen of abrupte stops.



Als we hier vervolgens *data mining* algoritmes zouden op toepassen kan dit leiden tot de extractie van bijvoorbeeld het volgende patroon: ***bezoekt(x, stadsplein) → abrupte\_stop(x) → haakse\_bocht(x)***, waarbij ***x*** het bewegende object voorstelt. In de secties 4.1.1 en 4.3.2 hebben we echter reeds gewezen op de inherente *spatial* en *temporal* onzekerheden verbonden met trajectdata en de hieruit geëxtraheerde *features* door zowel fysieke als technische beperkingen verbonden met de verzameling en opslag van deze data. Indien we vervolgens herverwijzen naar de belangrijke werken van Trajcevski, et al. (2003; 2004) uit deze secties, zouden we deze inherente trajectonzekerheden kunnen managen door het 'minen' naar *spatiotemporal* predicaten van de vorm: ***Definitely\_Sometime\_Inside(x, stadsplein) → abrupte\_stop(x) → haakse\_bocht(x)***. Hierbij draagt ***Definitely\_Sometime\_Inside(x, stadsplein)*** de betekenis dat object ***x*** met zekerheid gedurende tenminste één tijdsinstantie op het stadsplein (referentieregio, ***R***) aanwezig was. Hierdoor worden de (on)zekerheden immers geëxpliciteerd.

In tegenstelling tot voorgaande *direct search* benadering bestaat de mogelijkheid dat de gebruiker wil zoeken naar het voorkomen van bepaalde patronen, *inverse searches* of *queries* genaamd (Nanni, et al., 2007). Hierbij onderscheiden we twee soorten, namelijk *elementary* en *synoptic inverse queries*. *Elementary queries* hebben betrekking op het bewegingsgedrag van een individueel object, terwijl haar *synoptic* variant betrekking heeft op collectief bewegingsgedrag. Het zoeken naar alle trajecten die locatie ***A*** voorbijrijden tussen tijd ***t<sub>1</sub>*** en ***t<sub>2</sub>*** zou een voorbeeld kunnen zijn van een *elementary range query*.

*Synoptic queries* daarentegen hebben betrekking op het simultane gedrag en de interactie tussen een aantal (***k***) objecten, *k*-groeps patronen genaamd. Het groepsconcept kunnen we bijvoorbeeld vormgeven door simpelweg een maximale afstandslimiet tussen objecten te specificeren gegeven een bepaalde tijd. Ter illustratie van een *synoptic inverse query* zouden we bijvoorbeeld kunnen zoeken naar een groep van bewegende objecten met een gemiddelde snelheid onder de 20 km/uur, teneinde files te detecteren. Merk op dat deze groeps patronen sterk gerelateerd zijn aan clusteren, besproken in sectie 5.2.2.1, hoewel we ons hier eerder focussen op subsecties van de *data-space*. Als laatste is het bovendien belangrijk te vermelden dat een groeps patroon zich kan voortzetten in de tijd, hoewel haar groepsbezetting wijzigt en dus niet stabiel is (Kalnis, Mamoulis, en Bakiras, 2005). Files illustreren dit principe. Deze soort van groeps patronen

zijn gebaseerd op het *moving clusters* fenomeen, wat refereert naar clusters die hun karakteristieken behouden, hoewel er verschillende objecten in deelnemen tijdens haar levensduur.

### 5.2.2.3 Predicties

Gezien de prominentie van onder andere *just-in-time* logistieke diensten, *real-time* verkeersmanagement en *location-based services*, wijzen Nanni, et al. (2007) ons op de belangrijke rol van betrouwbare predicties betreffende een bewegend object zijn toekomstige posities. Het anticiperen van de beweging van deze objecten laat ons toe bijvoorbeeld corrigerende acties uit te voeren, files te vermijden of nuttige informatie te verstrekken. Naast toekomstige posities bieden *spatiotemporal* data ons eveneens diverse alternatieve voorspellingsmogelijkheden betreffende *ubiquitous* mobiliteits-toepassingen, zoals de predictie van de objectdensiteit en de *reach*. Samen met een korte uiteenzetting betreffende de classificatie van trajecten of delen hiervan, bekijken we deze predictietoepassingen vervolgens van naderbij.

De meeste predictietechnieken betreffende de toekomstige posities van een bewegend object zijn gebaseerd op de assumptie dat deze objecten hun beweging voortzetten in de tijd met een constante snelheidsvector (Nanni, et al., 2007). *Forecasting queries* bijvoorbeeld zijn in essentie gebaseerd op de huidige locatie  $\mathbf{l}_h$  en de constante snelheidsvector  $\mathbf{v}_c$ . De toekomstige locatie  $\mathbf{l}_t$  na tijd  $\Delta t$  kan dan worden berekend als  $\mathbf{l}_t = \mathbf{l}_h + \mathbf{v}_c * \Delta t$ . Hoewel deze assumptie geldig is voor lineaire beweging in een obstructieloze ruimte (bvb. vliegtuigen), leidt deze tot minder goede resultaten in toepassingen waar objecten hun richting en snelheid continu veranderen (bvb. voertuigen in een straten-netwerk). Naast de toekomstige posities van bewegende objecten is het bovendien ook dikwijls van belang hun meest waarschijnlijke traject en eindbestemming te kunnen anticiperen (Nanni, et al., 2007). De algemene assumptie achter deze predictietoepassing is dat mensen routinegebaseerd leven. Hierbij kunnen we onder andere gebruik maken van de eerder geëxtraheerde patronen, zodoende efficiënter te handelen.

De objectdensiteit is een dynamische en globale karakteristiek die wordt gedefinieerd als het aantal objecten in een bepaalde regio, in relatie tot haar oppervlakte, op een bepaald

tijdstip (Nanni, et al., 2007). Deze karakteristiek draagt onder andere allerhande voordelen met zich mee ter verbetering van allerlei verkeerssituaties. In het werk van Hadjieleftheriou, Kollios, Gunopulos en Tsotras (2003) berekenen ze deze dichtheid over een bepaald tijdsinterval door het creëren van een *spatiotemporal* kubus in het **XYZ**-vlak. Voor de predictie gaan ze uit van een lineaire beweging van de objecten en maken ze gebruik van de extrapolatie techniek.

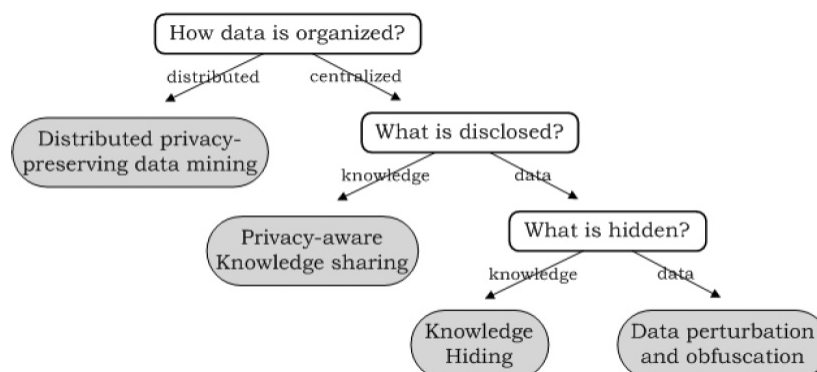
In de huidige context kan *reach* vervolgens worden gedefinieerd als de proportie objecten van een populatie die een bepaalde locatie kruisen gedurende een bepaalde tijdsspanne. *Reach* kan eveneens betrekking hebben op een netwerk van locaties, hierbij moeten de objecten dan minstens één van deze locaties kruisen (Nanni, et al., 2007). Betreffende deze predictietaak maken we gebruik van de extrapolatie van onvolledige steekproeftrajecten. Hierbij is het essentieel dat deze steekproef representatief is voor de volledige populatie. Zoniet moeten we gebruik maken van een stratificatieprocedure, zodoende te garanderen dat elke klasse van de populatie voldoende is gerepresenteerd in de steekproef (*training*- en testset) (Witten en Frank, 2000). De *reach*-karakteristiek kan allerhande waardevolle informatie verstrekken betreffende diverse *ubiquitous* mobiliteits-toepassingen waaronder *location-based services* (bvb. reclametoepassingen).

Als laatste bekijken we kort de classificatie van trajecten of delen hiervan (Nanni, et al., 2007). Classificatie leidt tot de verschaffing van diverse vormen van metadata betreffende bewegende objecten en hun trajecten, het kunnen onderscheiden van een toerist van een lokale inwoner op weg naar zijn werk kan hier een voorbeeld van zijn. Op basis van deze metadata kunnen bijvoorbeeld *location-based services* hun diensten gericht aanbieden. Hoewel Nanni, et al. (2007) er ons op wijzen dat onderzoek betreffende de mobiliteitscontext beperkt is, stellen ze echter dat we trajecten zouden kunnen classificeren aan de hand van allerhande *nearest-neighbour* algoritmes. Hierbij zal de specifieke classificatietaak aanleiding geven tot het specificeren van een bepaalde afstandsfunctie.

### 5.2.2.4 Privacy-Preserving Data Mining

Privacy is een essentiële vereiste in kennisgebaseerde mobiliteitstoepassingen. Vandaag de dag beschikken *service* providers immers over enorme hoeveelheden aan persoonlijke informatie welke ze kunnen gebruiken om hieruit patronen, modellen en trends te extraheren, teneinde hun diensten te optimaliseren. Hoewel de potentiële voordelen hiervan substantieel zijn, roept de verzameling en verwerking van deze persoonlijk gevoelige data vragen op betreffende de persoonlijke privacy (Bonchi, et al., 2007).

De bewustwording dat privacybescherming binnen *data mining* technieken onmisbaar is voor de sociale acceptatie van de hierop gebaseerde toepassingen en diensten, heeft geleid tot het ontstaan van de *privacy-preserving* tak binnen *data mining*. Hierbij kan het gaan over individuele privacy of zelfs *corporate* privacy, wanneer we handelen omtrent de bescherming van strategische informatie op het niveau van de organisatie (Lemmens, 2005). *Privacy preserving data mining* handelt rond de vraag hoe we trajecten van mobiele individuen kunnen analyseren zonder de privacy van deze individuen te bedreigen. In het werk van Bonchi et al. (2007) kunnen we lezen dat er in de literatuur hoofdzakelijk vier hoofdbenaderingen inzake *privacy preserving data mining* bestaan, gebaseerd op de volgende vragen: hoe zijn de data georganiseerd (gecentraliseerd of gedistribueerd), wat wordt er vrijgegeven (de data zelf of de kennis die eruit wordt geëxtraheerd) en wat wordt er verborgen (aspecten van de originele data, zoals de identiteit, of strategische kennis die uit deze data kan worden gehaald). In Figuur 5.2 kan u de resulterende classificatie bekijken. We gaan hier echter niet verder op in, voor meer informatie kan u evenwel terecht in het werk van Bonchi et al. (2007).



**Figuur 5-2** De hoofdbenaderingen inzake *privacy preserving data mining* (Bonchi, et al., 2007)

## *Hoofdstuk 6*

### *De UbiTracker Applicatie*

In dit hoofdstuk gaan we aan de hand van een praktijkbenadering dieper in op verschillende aspecten, behandeld doorheen deze scriptie. Deze praktijkbenadering wordt vorm gegeven door middel van mijn zelfontwikkelde UbiTracker applicatie, welke bestaat uit drie primaire doeleinden. Eerst en vooral willen we namelijk aan de hand van deze *ubiquitous* applicatie onderzoeken wat de mogelijkheden zijn van een alternatief voor satellietplaatsbepaling aangaande mobiele telefoons, waarna we gebaseerd op deze positioneringmethode via de applicatie *location-based services* en *track and trace* mogelijkheden willen aanbieden aan de mobiele gebruiker. Vooraleer we echter dieper ingaan op de architectuur en functionaliteit van de UbiTracker applicatie, bekijken we eerst deze drie hoofddoeleinden van naderbij.

#### *6.1 Hoofddoeleinden*

De UbiTracker applicatie is een *ubiquitous* mobiliteitstoepassing bestemd voor mobiele telefoons, een *location-aware technology* (LAT) die de mobiele gebruikers vergezellen. Voor het bepalen van de locatie van de gebruiker maken we gebruik van de *Cell Identity* methode (sectie 3.2.1.1), een positioneringmethode welke een alternatief vormt voor satellietplaatsbepaling en gebaseerd is op het *cellular* mobiele telefoonnetwerk. In sectie 6.2 gaan we hier dieper op in.

*Location-based services* (LBS) vervolgens, voorzien gebruikers van mobiele apparaten van informatie en diensten, specifiek afgestemd op de behoefte en locatie ervan (Mahmoud, 2004). Deze openen daarenboven legio nieuwe mogelijkheden voor zowel ontwikkelaars, als netwerk- en service providers betreffende het ontwikkelen en op de markt brengen van waardetoevoegende diensten, bijvoorbeeld: het adviseren omtrent

huidige verkeerssituaties, het bieden van route informatie of informatie aangaande nabijgelegen *points of interest* (POI). *Location-based services* beantwoorden namelijk drie primaire vragen (Mahmoud, 2004): Waar ben ik? Wat is er rond mij? Hoe geraak ik daar? Ze bepalen de locatie van de gebruiker aan de hand van een bepaalde positioneringstechniek (zie sectie 3.2) en gebruiken deze locatie, samen met andere informatie, voor het bieden van gepersonaliseerde diensten.

De *track and trace* mogelijkheid daarentegen heeft betrekking op het proces waarbij we continu de huidige positie van de mobiele gebruiker bepalen en bewaren (*track*) zodat deze posities later kunnen worden gereproduceerd op een kaart en we bijgevolg steeds de huidige positie van de mobiele gebruiker kennen (*trace*). Deze drie hoofddoeleinden zitten verweven in een enkele applicatie, de UbiTracker applicatie.

## 6.2 De Positioneringmethode

De *Cell Identity* methode waarop de UbiTracker applicatie voortbouwt is hier MS-gebaseerd en steunt op het feit dat het MS haar positie bij benadering kan bepalen, daar ze op ieder moment weet met welke cel ze is verbonden in het netwerk. Eens de verbonden cel is bepaald, kennen we eenvoudigweg de locatie van het *Base Transceiver Station* (BTS) van deze cel toe aan de mobiele gebruiker. Ten gevolge hiervan zal de accuraatheid van deze techniek afhankelijk zijn van de radius van de cellen in het netwerk, waardoor deze het meest optimaal zou moeten zijn in stedelijke gebieden. De dichtheid van GSM-masten is hier namelijk aanzienlijk groter, waardoor de cellen kleiner zullen zijn in oppervlakte. Ik heb bovendien voor deze techniek gekozen om de volgende redenen:

- De celinformatie bevindt zich in het MS zelf en kan op een eenvoudige manier worden benaderd.
- De methode is aanwezig binnen de functionaliteit van de *handset*, wat het direct beschikbaar maakt en eveneens goedkoop.
- Het één van de enige *cellular* methodes is die zelf valt te implementeren.
- Het bestaan van het OpenCellID project met haar *open source Cell ID* database (zie verder).

- De meeste mobiele telefoons nog niet beschikken over een geïntegreerde GPS-functionaliteit.
- Het de mogelijkheid biedt voor diverse toepassingen in stedelijke gebieden.

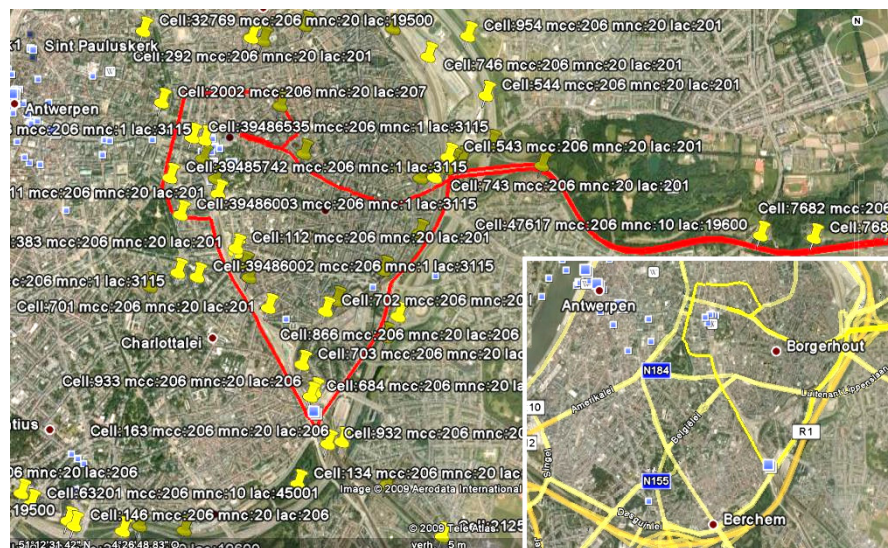
Het probleem met deze methode is evenwel dat de locatie van het BTS van de cellen geen publiek beschikbare informatie is. Providers houden deze informatie voor zichzelf, voornamelijk om de reden dat ze deze zelf als een extra inkomensbron willen aanwenden, door zelf LBS toepassingen aan te bieden. Het is dan ook hier dat het OpenCellID project ten tonele verschijnt. Dit project wordt vorm gegeven door een *open source* database bestaande uit celinformatie en heeft twee doeleinden. Iedereen kan de locatie van masten bepalen en deze uploaden naar de OpenCellID *server*, waardoor de reikwijdte van de database wordt geoptimaliseerd. De database kan vervolgens eveneens door iedereen worden aangewend om de benaderende locatie van hun mobiele telefoon te bepalen aan de hand van de *cell/get call* van de OpenCellID *Application Programming Interface* (API). Voor deze aanroep hebben we nood aan vier paramaters, welke beschikbaar zijn via het MS en worden meegegeven in de URL: *mcc* (*mobile country code*), *mnc* (*mobile network code*, de provider), *cellid* (*CellID*) en *lac* (*location area code*). Een voorbeeld van deze aanroep zou kunnen zijn: <http://www.opencellid.org/cell/get?mcc=206&mnc=20&cellid=133&lac=206>. De API geeft vervolgens twee retourparameters weer, namelijk *lat* (*lattice*) en *lon* (*longitude*) in decimale graden.

### 6.2.1 Praktijkttoetsing

Hoewel in theorie het OpenCellID project zich vooropstelt als een eenvoudig maar waardevol hulpmiddel aangaande de positionering van mobiele gebruikers, is het evenwel belangrijk onze hierop gebaseerde *Cell Identity* positioneringmethode te toetsen in de praktijk. In theorie luidde het eerder vermelde positioneringprincipe als volgt: ontdek de cel waarmee de mobiele gebruiker is verbonden en beschouw de locatie van het BTS van deze cel als die van hem. Hoewel dit principe eenvoudig is, biedt het interessante mogelijkheden als startpunt voor mijn applicatie. Het probleem met deze methode is evenwel dat we de mobiele gebruiker niet kunnen lokaliseren wanneer de verbonden cel niet geregistreerd is binnen de OpenCellID database, dit probleem zal echter geleidelijk aan worden verholpen naarmate meer en meer celinformatie wordt

opgenomen hierin. Even ter vergelijking: in de OpenCellID database zijn op dit moment onder andere 1592 Belgische masten geregistreerd, opgesplitst naar provider, ten opzichte van 12693 Nederlandse, 10047 Franse en 43976 uit het Verenigd Koninkrijk. Dit maakt de functionaliteit van het project voorlopig nog sterk land- en provider-gebonden. Daarnaast kunnen we momenteel ook nog op geen enkele manier weten hoe groot de radius is van de verbonden cel, waardoor we over geen informatie beschikken aangaande de accuraatheid van onze benaderende locatie. Een praktijktoetsing is dan ook de enige mogelijkheid teneinde de functionaliteit en accuraatheid van deze positioneringmethode te onderzoeken.

Voor het testen van mijn positioneringmethode maakte ik vervolgens gebruik van mijn mobiele telefoon met daarop de UbiTracker applicatie die om de tien seconden cel-informatie opvraagt en hiermee aan de hand van het OpenCellID project de locatie van de gebruiker opvraagt en bewaart. Hier bovenop maakte ik ook gebruik van een *GPS logger* zodoende mijn werkelijk traject te kunnen registreren en te vergelijken met datgene gebaseerd op de locaties bepaald via de UbiTracker positioneringmethode. Het eerste testtraject vond plaats in en rond de stad Antwerpen en werd gekozen op basis van de dichtheid van geregistreerde GSM masten in de regio, die in dit geval deel uit maken van het Base netwerk (mnc=20). Dit testtraject kan u waarnemen in Figuur 6-1.



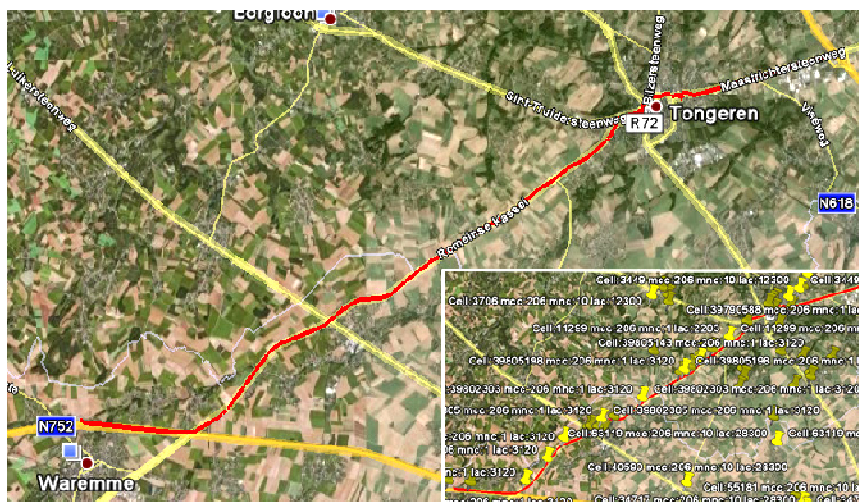
**Figuur 6-1** Het Antwerpse testtraject samen met de geregistreerde BTSs



Echter, de positioneringmethode is er geen enkele keer in geslaagd mijn benaderende locatie te bepalen gedurende het traject en de reden hiervoor is simpel: de mobiele telefoon heeft op geen enkel moment verbinding gemaakt met een cel die in de OpenCellID database is geregistreerd, desondanks de route over verschillende geregistreerde masten beschikte. Hoewel het MS steeds verbinding maakt met het BTS waarvan hij het sterkste signaal ontvangt, hoeft dit immers zoals eerder vermeld door signaleffecten en redenen van netwerkefficiëntie niet steeds de meest dichtstbijzijnde te zijn.

De stad Antwerpen zou normaliter een vrij accurate positionering moeten opleveren aangezien de radius van de cellen in het netwerk vrij klein is. Dit geldt echter enkel indien het merendeel van de aanwezige cellen geregistreerd zou zijn in de OpenCellID database, wat natuurlijk niet het geval is. Daar de stad over zeer veel masten beschikt, is de kans immers groot dat het MS verbinding maakt met een mast die niet in de database aanwezig is, wat ons vervolgens in de onmogelijkheid stelt de mobiele gebruiker te positioneren. Op aanraden van mijn promotor heb ik dan ook een nieuw testtraject gekozen in een iets meer landelijke regio, daar de kans hier groter is dat de geregistreerde masten ook quasi de enige aanwezigen zullen zijn in de regio.

Het meer landelijke tweede testtraject, dat u kan waarnemen in Figuur 6-2, vond plaats rondom Tongeren. De hier aanwezige geregistreerde masten maken bovendien deel uit van het Proximus netwerk (mnc=1).



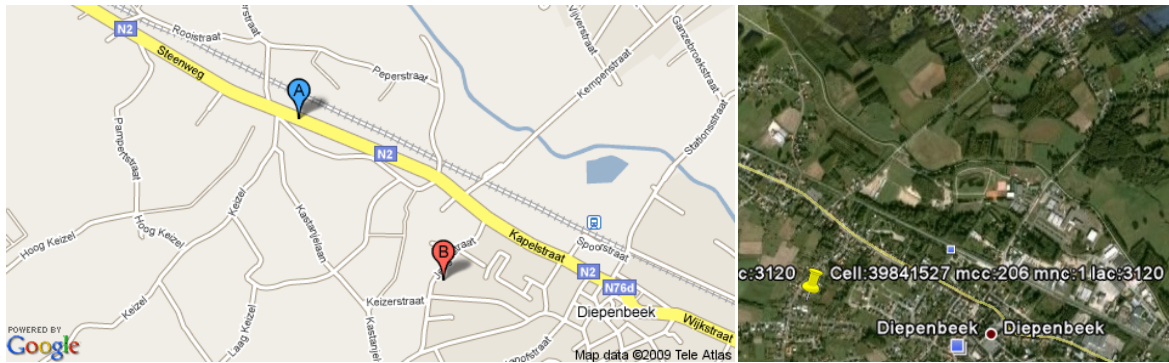
**Figuur 6-2** Het tweede testtraject samen met de geregistreerde BTSs

Dit testtraject leverde echter meer hoopvolle resultaten op. De positioneringmethode is er namelijk in geslaagd mij verschillende keren vrij accuraat te positioneren zoals u kan zien in Figuur 6-3. Ter illustratie van de accuraatheid van de positioneringmethode heb ik, gebaseerd op de testdata, een rode rechthoek getekend op de kaart. Dit is bij benadering het gedeelte van het traject waar de applicatie mij positioneerde als zijnde op positie twee, wat overeenkomt met een accuraatheid van plusminus 0-300 m. Hoewel echter deze accuraatheid gedurende het traject telkens varieert, is deze vrij gelijkaardig hieraan. Tussen de gelogde posities in op de kaart ben ik ook verschillende keren verbonden geweest met een cel die niet in de OpenCellID database aanwezig was, waardoor de methode dan ook niet in de mogelijkheid was om mij te positioneren.



**Figuur 6-3** Gelogde posities op het tweede testtraject

Naast de twee testtrajecten heb ik de methode daarenboven ook frequent uitprobeerde op *random* locaties doorheen België, dit echter meestal zonder resultaat tot ik het uitprobeerde op een locatie in Diepenbeek aan de hand van een Proximus SIM kaart. Het resultaat hiervan kan u waarnemen in Figuur 6-4, waarbij A mijn werkelijke positie voorstelt en B de locatie is die werd bepaald via de UbiTracker applicatie (cellid=39841527, mcc=206, mnc=1 & lac=3120). De positioneringfout van mijn benaderende locatie komt neer op ongeveer 1 km, wat weliswaar minder accuraat is dan in het voorgaande geval.



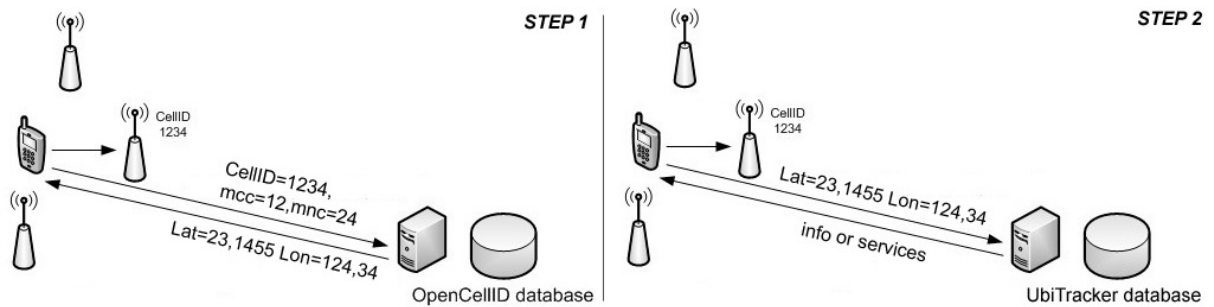
**Figuur 6-4** Testlocatie in Diepenbeek

Hieruit kunnen we besluiten dat hoewel het project nog in zijn ontwikkelingsfase zit, het reeds goede mogelijkheden biedt qua startpunt van mijn UbiTracker applicatie. Hoewel de accuraatheid van de positioneringmethode varieert van zowel plaats tot plaats als netwerk tot netwerk heb ik via deze praktijktoetsing evenwel kunnen aantonen dat een 'semi-exacte' positionering, zoals bij het tweede testtraject ( $\pm 0-300$  m), niet onmogelijk is. Voor deze reden heb ik dan ook in de ontwikkeling van de applicatie gekozen om de positie van de mobiele gebruiker bepaald via de UbiTracker positioneringmethode als 'semi-exact' te beschouwen. Bovenop voorgaande reden biedt deze 'semi-exacte' locatie eveneens veel meer mogelijkheden aangaande de functionaliteit van de UbiTracker applicatie en beschikken we bovendien over geen enkele informatie aangaande de radius van de verbonden cel, tengevolge waarvan we momenteel niet over de mogelijkheid beschikken een uitspraak te doen betreffende de werkelijke accuraatheid. Dit vormt echter wel een geplande update van het OpenCellID project.

### 6.3 Architectuur en Functionaliteit

Op een abstract niveau kunnen we vervolgens stellen dat de UbiTracker applicatie bestaat uit twee opeenvolgende stappen, dit kan u ook waarnemen in Figuur 6-5. Eerst en vooral bepalen we aan de hand van het MS de vier parameters, nodig voor de *cell/get call* van de API, waarmee we een draadloze connectie leggen met de *OpenCellID* database (*step 1*). De locatie-informatie die de database ons biedt verzenden we

vervolgens via het *Wireless Application Protocol* (WAP) naar onze eigen UbiTracker database, waar we de bezochte locaties loggen voor de *track and trace* mogelijkheid, en als basis voor het bieden van informatie en diensten aan de mobiele gebruiker (*step 2*). Deze stappen worden verder nog uitgebreider uit de doeken gedaan.



**Figuur 6-5** Stappen binnen de UbiTracker applicatie

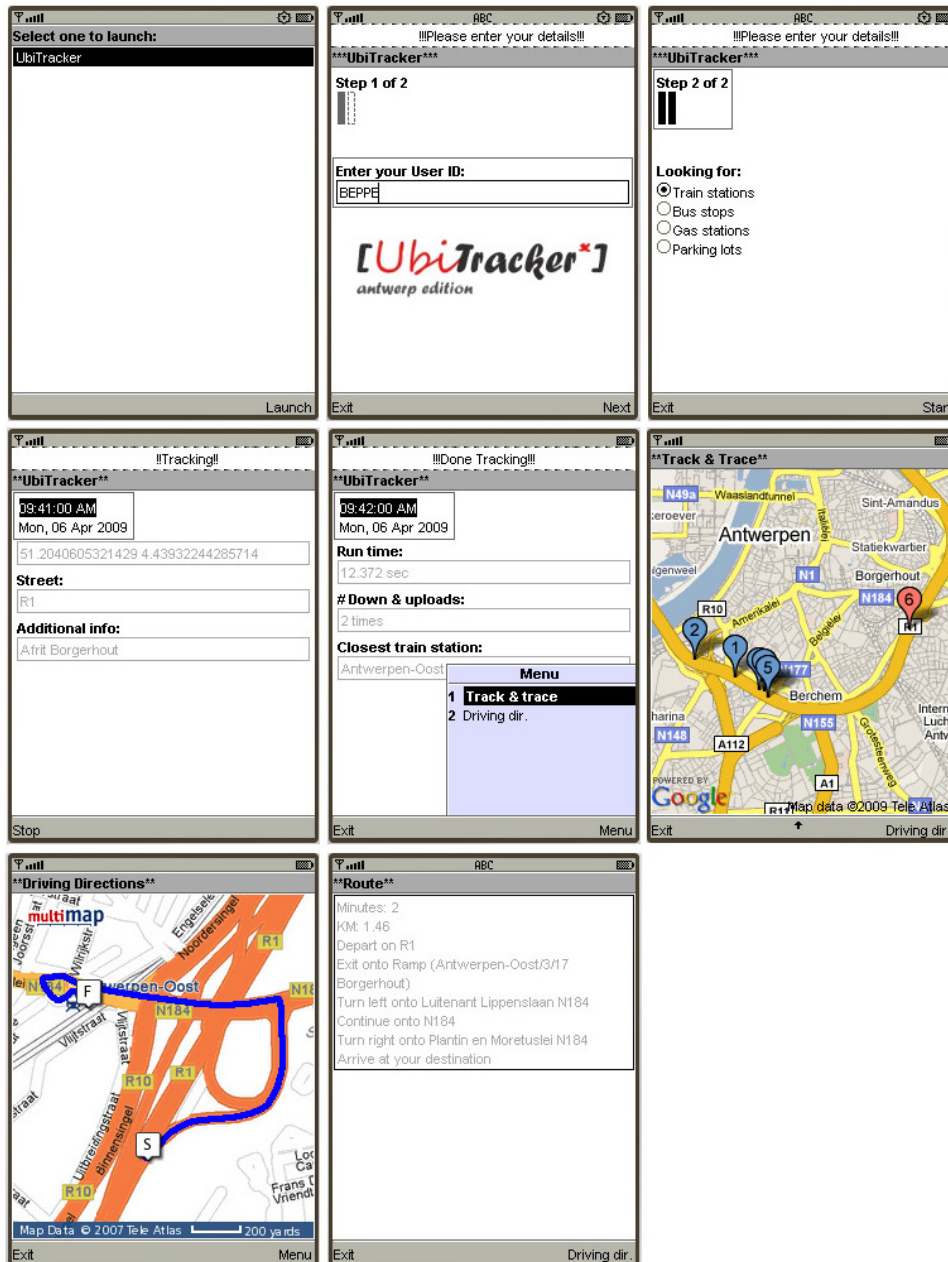
De interface voor de mobiele telefoons is vervolgens geschreven in Java *Micro Editions* (ME), speciaal ontworpen voor het programmeren van apparaten met beperkte capaciteiten. Dit maakt bovendien dat we de UbiTracker applicatie een *MIDlet* noemen. Deze javacode (Bijlage B.1) zorgt bovendien voor de locatiebepaling en handelt de retourparameters van de databases af. Connecties met de databases gebeuren op basis van het *Wireless Application Protocol*, beter gekend onder de naam WAP. De UbiTracker database is bovendien een MySQL database, welke zich bevindt op een *webserver*. Voor het bieden van informatie en diensten aan de mobiele gebruiker maak ik daarnaast gebruik van PHP scripts, welke zorgen voor de communicatie tussen de *handset* en de databases en zich eveneens op de *server* bevinden. Zoals u verder nog kan lezen maak ik bovendien ook gebruik van de Open APIs van Google Maps en multiMap voor de kaarten en de routebepaling.

Voor de *track and trace* mogelijkheid van de applicatie maak ik vervolgens gebruik van JavaScript, PHP en HTML. Deze functionaliteit bestaat namelijk uit zowel een *handset* als een PC module, daar de capaciteiten van een mobiele telefoon soms te beperkt zijn. Deze PC-zijde zit geïntegreerd in de online module van de UbiTracker applicatie (zie sectie 6.3.1).

### 6.3.1 Werking

In de ontwikkeling van de applicatie hebben we ons vervolgens beperkt tot de stad Antwerpen. Hoewel de stad tegenvallende resultaten opleverde tijdens de praktijktoetsing, biedt deze ons allerhande mogelijkheden wat betreft *location-based services* en beschikt de stad bovendien over een zeer hoge dichtheid qua GSM masten. Dit zou vervolgens moeten leiden tot een vrij accurate positionering, eens het merendeel van deze masten is geregistreerd in de OpenCellID database.

In essentie doet de UbiTracker applicatie vervolgens het volgende: eerst en vooral geeft de gebruiker zijn *User ID* in en duidt hij aan of hij op zoek is naar treinstations, bushaltes, tankstations of parkings. Beiden worden opgeslagen in variabelen, welke de functionaliteit beïnvloeden doorheen het programma (in het vervolg van de werking veronderstellen we dat de gebruiker geïnteresseerd is in treinstations). Vervolgens gaat de applicatie om de tien seconden celinformatie opvragen via het MS, welke via WAP wordt doorgegeven naar de OpenCellID server. Deze geeft dan de huidige locatie weer van de gebruiker, welke we loggen in onze UbiTracker database. Op basis van deze locatie-informatie proberen we ook continu straatgegevens en andere beschikbare additionele informatie weer te geven op de mobiele telefoon (zie verder). Drukt de gebruiker vervolgens op de stopknop dan bepaalt de server het dichtstbijzijnde treinstation. Deze afstand is niet die van de kortste route hiernaartoe, maar deze in vogelvlucht, daar we het aantal WAP connecties willen beperken en dit doorgaans toch het correcte antwoord geeft. Vertrekkende vanuit de twee coördinaatparen in decimale graden **(lat1, lon1), (lat2, lon2)** is deze afstand namelijk gebaseerd op de formule  **$1,609344 * \sqrt{((69,1*(lat1 - lat2))^2 + (53*(lon1 - lon2))^2}$** . Hierna kan de gebruiker vervolgens al zijn bezochte locaties opvragen (*handset* module van de *track and trace* mogelijkheid), of onmiddellijk de route bekijken naar dit dichtstbijzijnde treinstation, gebaseerd op zijn huidige locatie. Deze route wordt weergegeven op een multiMap kaart, maar kan na een druk op de routeknop evenzeer weergegeven worden in woorden. Dit is mogelijk doordat we de route via de multiMap Open API eveneens kunnen opvragen in een XML vorm, waarna we hieruit de benodigde informatie kunnen filteren aan de hand van een PHP script. U kan deze werking grafisch bekijken in Figuur 6-6.



**Figuur 6-6** Grafische weergave van de UbiTracker applicatie

Zoals reeds eerder vermeld is de *handset* module van de UbiTracker applicatie opgebouwd uit de UbiTracker Java ME code welke draait op de mobiele telefoon, evenals vijf PHP scripts welke zich bevinden op de *webserver* waarmee de applicatie verbinding maakt. Het eerste PHP script is het db script. Dit zorgt in alle andere scripts voor de connectie met de UbiTracker database en kan worden teruggevonden in Bijlage B.2. Het

streetandinfo script (Bijlage B.3) zorgt vervolgens voor het beschikbaar stellen van de straat- en andere additionele informatie in de applicatie. Het UbiTracker script (Bijlage B.4) daarentegen zorgt voor het loggen van de bezochte locaties van de gebruiker in onze database terwijl het station script (Bijlage B.5) het dichtstbijzijnde station in vogelvucht bepaalt en het trackandtrace script (Bijlage B.6) zorgt voor het mappen van zowel de voorgaande als de huidige positie van de mobiele gebruiker. Het XMLread script (Bijlage B.7) als laatste zorgt voor het weergeven van de geschreven routebeschrijving in de applicatie.

De internetzijde van de applicatie daarentegen bestaat uit een online module welke, op basis van de huidige locatie van de mobiele gebruiker, de afstand tot alle treinstations in zijn omgeving weergeeft. In tegenstelling tot de *handset* module is deze de werkelijke afstand, rekeninghoudend met de kortste route. Deze afstand is namelijk gebaseerd op een JavaScript code welke niet wordt ondersteund op het mobiele telefoon platform. Als de gebruiker vervolgens kiest voor één van deze mogelijkheden, geeft de module de kortste route hiernaartoe weer op basis van een kaart evenals geschreven richtingaanwijzingen, beide gebaseerd op de Google Maps Open API. Hiervoor legt deze online module eveneens connecties met de UbiTracker database. U kan deze module grafisch bekijken in Bijlage A. De online module bestaat bovendien uit drie scripts, namelijk het home, distance en map script welke u respectievelijk kan terugvinden in Bijlage B.8, B.9 en B.10. De home HTML code vormt het *inlog* scherm, waarna het distance script op basis van PHP en JavaScript de afstand tot alle mogelijke treinstations bepaalt. Het map script als laatste geeft de route weer van de huidige locatie van de gebruiker tot het gekozen treinstation, zowel in woorden als op een Google Maps kaart.

### 6.3.2 *De databases*

De *OpenCellID* database bevat informatie omtrent de locatie van het BTS, uniek verbonden met een cel. Deze locatie kunnen we opvragen op basis van de vier parameters beschreven in sectie 6.2.

De UbiTracker database bevat daarnaast twee andere tabellen. Één tabel wordt gebruikt om de bezochte locaties van de mobiele gebruiker te registreren. Hierbij wordt elke *tuple*

of rij in de tabel voorzien van een tijdsaanduiding samen met de *user ID* van de gebruiker, wat elke rij uniek maakt in de tabel. Deze tijdsaanduiding en ID maken het ook mogelijk de huidige locatie van de gebruiker op te vragen of zelfs al de bezochte locaties voor de *track and trace* mogelijkheid.

Hiernaast bevat de UbiTracker database eveneens een tabel met informatie waarop we onze *location-based services* baseren. Zo bevat deze de straatnaam bij elke geregistreerde mast in Antwerpen, wat extra informatie bij elke positie en de posities van de treinstations. Deze straat- en extra informatie worden weergegeven in de *user interface* van de applicatie, afhankelijk van de huidige locatie van de gebruiker. De positie-informatie van de treinstations daarentegen wordt gebruikt om het dichtstbijzijnde station te bepalen samen met de route hiernaartoe in het MS gedeelte en de online module. Momenteel bevat deze tabel enkel maar informatie betreffende de trainstations in de regio Antwerpen, waardoor dit dan ook de enige mogelijkheid is binnen de applicatie tot nu toe. Zou ik echter informatie in de database opnemen omtrent alle busstops, tankstations of parkings in Antwerpen, zou de functionaliteit eenvoudig uitgebreid kunnen worden, daar dit steunt op dezelfde principes.

### 6.3.3 Foutafhandeling

Één van de problemen waarmee applicaties continue te maken krijgen zijn *errors* of fouten. Dit maakt dat foutafhandeling een integraal onderdeel moet zijn van de applicatie, van het begin tot het einde, om een zo hoog mogelijk *service* niveau te kunnen garanderen. De *services* waar we gebruik van maken, kunnen soms al eens onbeschikbaar zijn, of de gebruiker kan handelingen verrichten die niet zijn toegelaten. Dit begint reeds bij het invoeren van de *User ID*. Vergeet de gebruiker deze in te geven, wordt hij hiervan onmiddellijk op de hoogte gebracht. WAP connecties met de databases en scripts op de server worden ook steeds op een manier verricht die de applicatie niet kunnen blokkeren. Het is bovendien ook steeds mogelijk dat een locatie niet wordt gevonden, waardoor de applicatie hier ook rekening mee houdt en de gebruiker wijst op het feit dat zowel de locatie als de route niet beschikbaar zijn. Indien er geen straatnaam of informatie beschikbaar is, wordt de gebruiker hier ook steeds van op de hoogte gebracht.

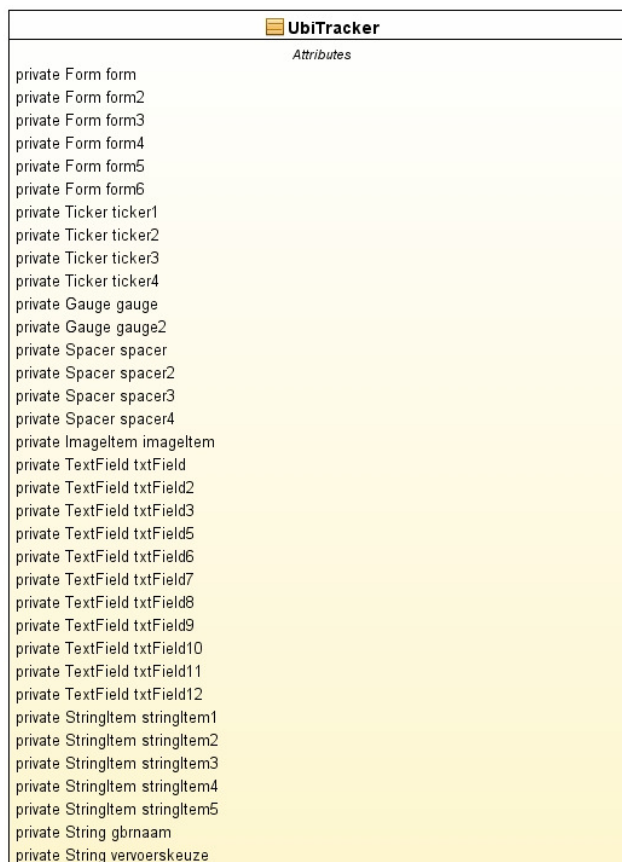


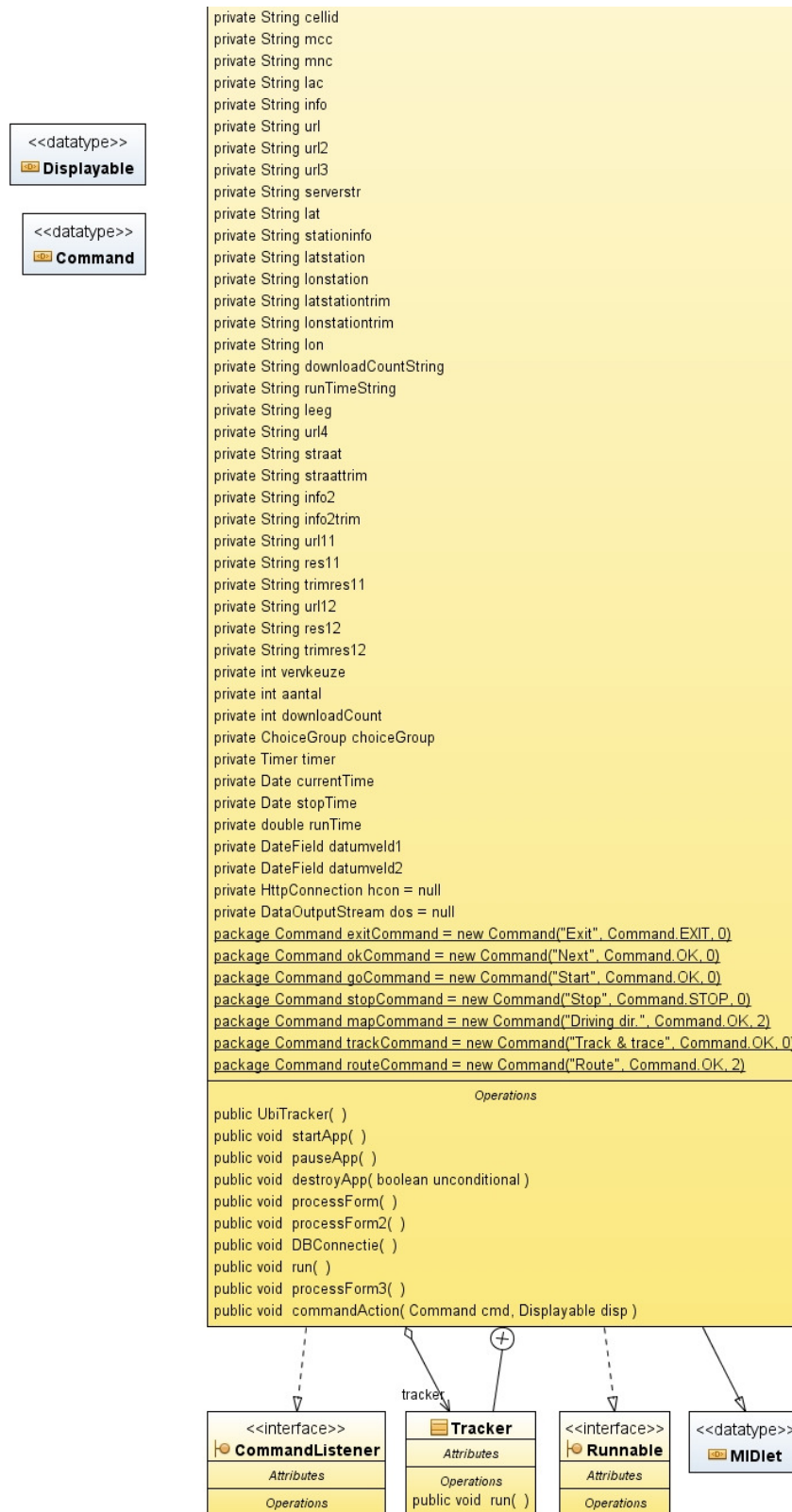
### 6.3.4 Reverse Engineering

*Reverse engineering* in deze context heeft betrekking op het omzetten van onze UbiTracker *source code* naar *Unified Modeling Language* (UML) modellen of diagrammen die bijdragen aan zowel de begrijpbaarheid, als de *reusability* van de code. We bekijken achtereenvolgens het klassendiagram en het activiteitendiagram van de *handset* module van de UbiTracker applicatie.

#### 6.3.4.1 Klassendiagram

Het klassendiagram beschrijft de structuur van de UbiTracker applicatie door haar klasse en subklasse weer te geven samen met diens attributen en methodes. Het klassendiagram kan u waarnemen in Figuur 6-7.

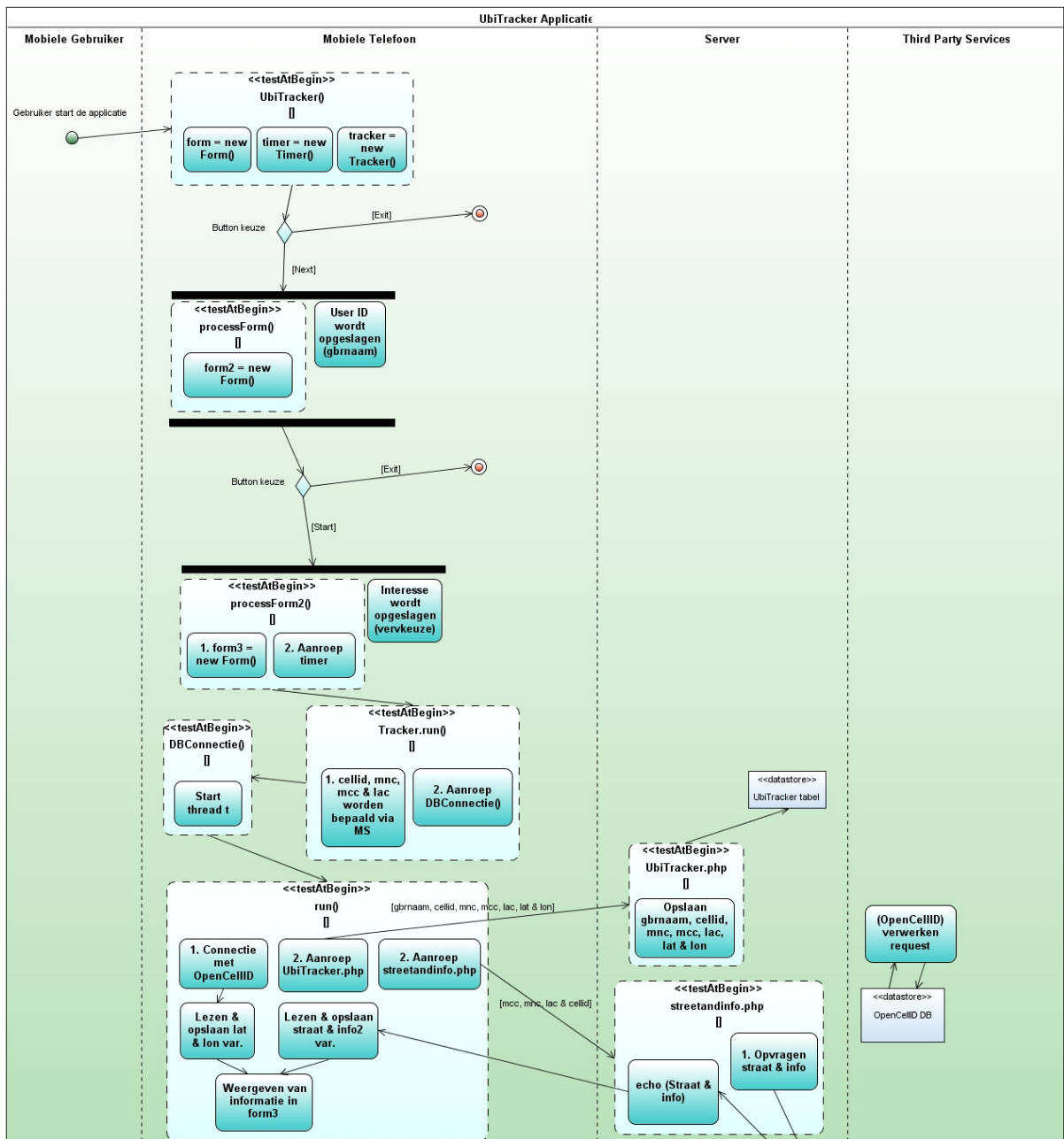


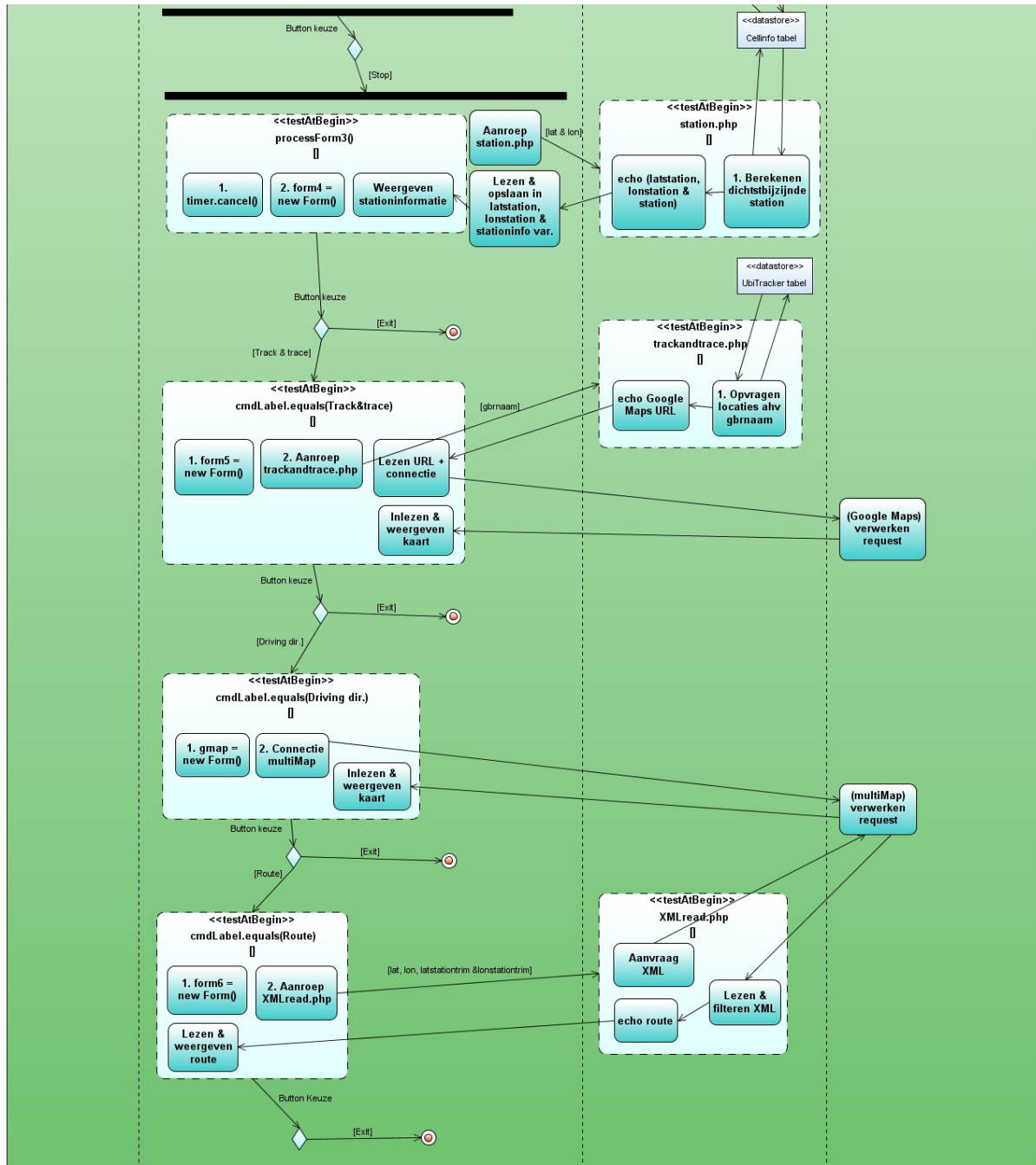


Figuur 6-7 Het UbiTracker klassendiagram

### 6.3.4.2 Activiteitendiagram

Het activiteitendiagram van de UbiTracker applicatie biedt ons vervolgens een overzicht van de verschillende stappen of activiteiten gedurende het uitvoeren van de *source code*, opgesplitst naar de verschillende actoren (*swimlanes*). Daar we echter de begrijpbaarheid van de code willen optimaliseren bevat dit activiteitendiagram tal van abstracties. Dit diagram kan u vervolgens waarnemen in Figuur 6-8.





Figuur 6-8 Het UbiTracker activiteitendiagram

## 6.4 Optimalisatiemogelijkheden

Hoewel ik zeer tevreden ben met het resultaat, durf ik ook te onderkennen dat de UbiTracker applicatie over tal van optimalisatiemogelijkheden beschikt. Eerst en vooral konden we tijdens de praktijktoetsing van de positioneringmethode opmerken dat de positionering momenteel nog niet echt van een leien dakje loopt. Daar dit grotendeels te wijten is aan het feit dat het OpenCellID project nog volop in haar ontwikkeling is, zouden we kunnen besluiten een andere techniek als basis voor de applicatie te nemen. Dit zou echter geen simpele opgave zijn, maar kan tot de mogelijkheden behoren. Daarnaast zouden we, eens we over informatie beschikken aangaande de radius van de verbonden cel, de gebruiker informatie kunnen bieden omtrent de accuraatheid van zijn positionering en bovendien de functionaliteit hier beter op afstemmen. Zo zouden we bijvoorbeeld een zekere *range* kunnen weergeven wanneer we de positie van de mobiele gebruiker mappen op een kaart (zie Figuur 6-9). De functionaliteit van de applicatie op zich kan ook steeds worden geoptimaliseerd of uitgebreid naar de toekomst toe. Zelfs in de architectuur van de applicatie zitten mogelijkheden ter verbetering.



**Figuur 6-9** Optimalisatiemogelijkheid aangaande de positionering van de gebruiker

## *Algemene Conclusie*

In het begin van deze scriptie werden een centrale onderzoeksvraag en een aantal deelvragen geformuleerd. De antwoorden op deze vragen werden u hopelijk duidelijk in de loop van dit werk. De belangrijkste vaststellingen zullen we hier vervolgens overlopen in een algemene conclusie.

*Ubiquitous computing* is een term verwijzend naar de alom aanwezigheid van computers in onze maatschappij, mede mogelijk gemaakt door de steeds verdergaande miniaturisatie van computertechnologieën. De mens wordt als het ware omsingeld door ontelbare, kleine en inter-verbonden computers, onzichtbaar ingebouwd in allerlei soorten van dagdagelijkse objecten en apparaten, die mensen assisteren in hun alledaagse leven. Zodoende worden deze objecten en apparaten voorzien van allerhande *processing*- en communicatiecapaciteiten, zodat deze kunnen handelen op een omgevingsgevoelige manier. *Ubiquitous computing* zal bovendien leiden tot een naadloze verbinding van de fysieke wereld met haar belichaming in informatie systemen. Verschillende nieuwe intelligente applicaties kunnen bovendien worden geïmplementeerd, steunend op het voorgaande. Echter, beperkingen omtrent deze toepassingen zullen eerder komen vanuit een economische, sociale en ethische hoek, dan vanuit de technologie zelf.

Op vlak van het bedrijfsleven biedt *ubiquitous computing* vervolgens drie grote voordelen. Ten eerste is dit het voorkomen van *media breaks* tussen de werkelijke en de digitale wereld, daar het de menselijke tussenkomst elimineert. Het volgende voordeel is de bewustwording van *smart* objecten, wat hun grotendeels zelfstandig maakt. Als laatste zorgt *ubiquitous computing* ervoor dat data worden gedistribueerd tot op het punt waar ze kunnen worden gebruikt of gegenereerd. Deze economie op autopiloot biedt echter evenzeer substantiële risico's. Zo kan de exclusie van mensen als beslissingsnemers leiden tot problemen onder abnormale omstandigheden, of kan de '*lack of slack*' ernstige consequenties met zich meebrengen, onder andere in de *supply chain*.

Naast haar impact op het bedrijfsleven stelt de *ubiquitous computing* visie vervolgens expliciet voor de maatschappij grondig te veranderen door ze volledig te informatiseren.

Ten gevolge hiervan zullen we sterk afhankelijk zijn van het betrouwbaar functioneren van de *ubiquitous* systemen rondom ons, wat zal resulteren in een aantal fundamentele sociale en ethische uitdagingen. Hiervan is privacy onbetwistbaar de meest prominente. Velen maken zich namelijk zorgen voor een potentieel misbruik ervan en een mogelijk gebrek aan controle erop. Hierdoor is er nood aan zowel wettelijke overwegingen, als nieuwe technische benaderingen en sociale inspanningen om deze uitdagingen te trotseren. Andere sociale en ethische uitdagingen hebben onder andere betrekking op de sociale comptabiliteit en aanvaarding van *ubiquitous computing*.

*Ubiquitous computing* biedt ons bovendien evenzeer talrijke mogelijkheden op vlak van transport en mobiliteit, maar ter realisatie hiervan moeten we evenzeer het hoofd kunnen bieden aan allerhande uitdagingen en risico's, welke vooral betrekking hebben op het leren omgaan met bewegende objecten in het 4D universum van plaats en tijd. Voor de ontwikkeling van deze mobiliteitstoepassingen hebben we namelijk nood aan een effectief *ubiquitous* systeem dat mobiliteitsdata aangaande bewegende objecten verzamelt, bewaart en verwerkt, teneinde de mobiele gebruiker te kunnen voorzien van allerhande relevante informatie en diensten, afhankelijk van diens geografische positie.

Aangaande de verzameling van mobiliteitsdata zien we dat nieuwe ontwikkelingen binnen mobiele communicatietechnologieën en *ubiquitous computing* ervoor zorgen dat we op een eenvoudigere manier toegang hebben tot een omvangrijker volume hiervan. Onze dagelijkse activiteiten en de manier waarop mensen leven en bewegen, laten namelijk digitale sporen na. Om mobiele gebruikers te lokaliseren focusten we ons binnen deze scriptie op twee prominente technologieklassen. Deze positioneringstechnologieën vormen als het ware de mobiliteitsdatabronnen. Eerst en vooral hebben we gekeken naar de *cellular-based* technologieën, welke voortbouwen op het mobiele telecommunicatienetwerk. Als alternatief hiervoor bekeken we eveneens de satellietplaatsbepaling, welke is gebaseerd op de reistijden van elektromagnetische impulsen tussen een zender en een ontvanger.

Daar deze data worden verwezenlijkt op een *ubiquitous* manier, beschikken ze bovendien over additionele karakteristieken en aspecten welke hun effect hebben in de verdere verwerking ervan. *Ubiquitous* data worden namelijk gecreëerd op een asynchrone, gedecentraliseerde manier door verschillende zwakgekoppelde, gedeeltelijk overlappende en mogelijk contradicterende bronnen. Hierdoor wordt onder andere hun centrale opslag

kostelijk, hebben we vaak te maken met *datastreams* en diverse datatypes en bestaat de mogelijkheid dat deze data elkaar overlappen of zelfs tegenspreken.

Naast de individuele posities van de mobiele gebruikers zijn we dikwijls eveneens geïnteresseerd in hun beweging of trajecten die deze beweging beschrijven. Data inzake deze trajecten bestaan op een abstract niveau vervolgens uit een tijds-, populatie- en ruimtecomponent, daar we een traject kunnen beschouwen als een functie die de populatie en tijd relateert aan locaties in de ruimte. De karakteristieken van deze trajecten kunnen we overigens opdelen in momentengebaseerde karakteristieken (bvb. snelheid en richting) en karakteristieken die trajecten in het algemeen, of een fragment hieruit beschrijven (bvb. bewegingsvorm en totale reistijd). Voor zowel de ruimte- als de tijdscomponent hebben we bovendien nood aan een referentiesysteem en moeten we de ruimte uit praktische overwegingen discretiseren. Deze trajecten moeten we bovendien reconstrueren uit een verzameling steekproefwaarnemingen aangezien we continue beweging onmogelijk kunnen vatten in haar volledigheid. Hiervoor maken we gebruik van een proces dat we interpolatie noemen. De accuraatheid van deze trajecten is bovendien afhankelijk van de locatieonzekerheden voortvloeiend uit dit interpolatieproces, evenals de *sampling rate* en de accuraatheid van de gebruikte positioneringstechniek. Aan de bekomen trajecten is zodoende een zekere inherente onzekerheid gekoppeld, welke we moeten managen binnen de reconstructie, representatie, opslag of het verder gebruik ervan. Dit kunnen we onder andere bewerkstelligen door gebruik te maken van een techniek die deze onzekerheid incorporeert door ze te expliciteren.

Vervolgens moet deze grote hoeveelheid aan *spatiotemporal* data aangaande bewegende objecten, ofwel snel worden verwerkt voor *real-time* doeleinden, ofwel onderworpen worden aan allerlei *knowledge discovery* algoritmes. Eender welke verwerking wordt vereist, het bewaren van deze data is essentieel. Hiervoor focussen we ons op een speciale vorm van *spatiotemporal* databases, *Moving Objects Databases* (MOD) genaamd, en maken we bovendien gebruik van datamodellen die beschrijven hoe deze data zijn gestructureerd. Betreffende deze opslag is het vervolgens belangrijk rekening te houden met enkele managementkwesties hieromtrent. Om *queries* te optimaliseren gaan *spatiotemporal* databases namelijk gebruikmaken van *spatiotemporal* indexeringsmethoden, welke objecten in een 3D ruimte zodanig rangschikken zodat ze efficiënt teruggevonden kunnen worden. Daarnaast is het eveneens belangrijk om te kunnen gaan met database *queries* wanneer er sprake is van locatieonzekerheden. Dit kunnen we



onder andere oplossen door *spatiotemporal* predicaten zodanig te formuleren zodat deze onderzekerheden worden geëxpliciteerd. Als laatste stelden we dat *Traject-Data Warehouses* (TDWs) onder andere gebruikt kunnen worden ter ondersteuning van de beslissingsdoeleinden in diverse mobiliteitstoepassingen.

*Knowledge discovery* vervolgens vindt vandaag de dag plaats in een des te meer kennisrijke en rekenkrachtige omgeving. Mobiliteitskennis, verborgen in de talrijk verzamelde en kennisrijke mobiliteitsdatarecords, biedt dan ook talloze mogelijkheden ter ondersteuning van allerhande *ubiquitous* transport- en mobiliteitstoepassingen. Vooraleer we ons echter richten op de *spatiotemporal knowledge discovery* aspecten en methoden hebben we nood aan een gedegen begrip van de aspecten en technieken betreffende haar *spatial* variant. Binnen het *knowledge discovery* proces beschouwen we eerst en vooral de *feature extraction* stap, welke we aanwenden om *spatial* of *spatiotemporal* karakteristieken die impliciet gecodeerd zijn in de objecten te extraheren, daar deze waardevolle informatie kunnen aanbrenge. Vervolgens spitsen we ons toe op het *minen* van deze *spatial* en *spatiotemporal* data. Hierbij is het belangrijk op te merken dat de mobiliteitscontext waarop we ons toespitsen allerhande extra uitdagingen met zich meebrengt, waardoor het essentieel is dat de algoritmes over de mogelijkheid beschikken deze te managen en te incorporeren. Zo handelen we steeds in een onstabiele omgeving en verandert het mobiele gedrag van individuen in de tijd. Kennis geëxtraheerd uit trajectdata zou ons vervolgens kunnen bijstaan in diverse taken betreffende alledaagse transport- en mobiliteitsfenomenen zoals het detecteren en voorspellen van files of het detecteren van relaties tussen allerhande gebeurtenissen.

## *Lijst van de Geraadpleegde Werken*

1. Abowd, G.D., & Mynatt, E.D. (2000). Charting past, present, and future research in ubiquitous computing. *ACM Transactions on Computer-Human Interaction*, 7, 29-58.
2. Alvares, L. O., et al. (2007). Towards semantic trajectory knowledge discovery. Universiteit Hasselt technisch rapport.
3. Alvares, L. O., Bogorny, V., Macedo, J., Moelans, B., & Spaccapietra, S. (2007, november). *Dynamic modelling of trajectory patterns using data mining and reverse engineering*. Paper gepresenteerd op de 26e International Conference on Conceptual Modelling, Auckland.
4. Andrienko, N., Andrienko, G., Peleklis, N., & Spaccapietra, S. (2007). Basic concepts of movement data. In F. Giannotti & D. Pedreschi (Reds.), *Mobility, Data Mining and Privacy – geographic knowledge discovery* (pp. 15-38). Berlijn: Springer.
5. Araya A. A. (1995). *Questioning ubiquitous computing*. Paper gepresenteerd op het 23e Annual Conference on Computer Science.
6. Bekiaris, E., Panou, M., Mizaras, V., & Petraki, E. (2004). *IM@GINE-IT*.
7. Be-Mobile. (z.d.). *Floating vehicle data*. Opgevraagd op 18 januari 2009, van de volgende website: <http://www.be-mobile.be/en/technology/floatingvehicledata>
8. Beresford, A. R. (2005). Location privacy in ubiquitous computing. *University of Cambridge Computer Laboratory*, 612.
9. Bohn, J., Coroama, V., Langheinrich, M., Mattern, F., & Rohs, M. (2004). Living in a world of smart everyday objects – social, economic, and ethical implications. *Journal of Human and Ecological Risk Assessment*, 10, 763-786.
10. Bonchi, F., et al. (2007). Privacy in spatiotemporal data mining. In F. Giannotti & D. Pedreschi (Reds.), *Mobility, Data Mining and Privacy – geographic knowledge discovery* (pp. 297-333). Berlijn: Springer.
11. Brakatsoulas, S., Pfoser, D., & Tryfona, N. (2004, juli). *Modeling, storing and mining moving object databases*. Paper gepresenteerd op het Database Engineering and Applications Symposium IDEAS '04, Coimbra.

12. Bulusu, N., Heidemann, J., & Estrin, D. (2000). GPS-less Low Cost Outdoor Localization For Very Small Devices. *IEEE personal communications magazine*, 7, 28-34.
13. Chakka, V. P., Everspaugh, A. C., & Patel, J. M. (2003). *Indexing large trajectory data set with SETI*. Paper gepresenteerd op de 2003 CIDR Conference.
14. Chu-feng, L. (2006). *A survey on cellular based localization and location based services*. Paper gepresenteerd op de WMS 2006 conferentie.
15. Crowcroft, J. (2003). *Scalable ubiquitous computing systems or just ubiquitous systems*. Paper gepresenteerd op de UK Computing Research conferentie 2003.
16. D'Auria, M., Nanni, M., & Pedreschi, D. (2006). *Time-focused density-based clustering of trajectories of moving objects*.
17. de Hoog, A., Steenbruggen, J. G. M., & Crombaghs, M. J. E. (2003). Plaatsbepaling met behulp van het GSM netwerk. *GeoNieuws*, 3.
18. Djukenic, M. G., & Richton, R. E. (2001). Geolocation and assisted GPS. *Computer*, 34(2), 123-125.
19. D'Roza, T., & Bilchev D. (2003). An overview of location-based services. *BT Technology Journal*, 21(1), 20-27.
20. ETSI. (2000). GSM 03.71, Digital cellular telecommunications system (Phase 2+); Location Services (LCS); (Functional description) - Stage 2, version 8.0.0.
21. ETSI. (2008). *Intelligent transport systems*. Opgevraagd op 14 januari 2009, van de volgende website:  
<http://www.etsi.org/WebSite/Technologies/IntelligentTransportSystems.aspx>
22. Fleisch, E. (2001). *Business perspectives on ubiquitous computing*, M-Lab working paper No. 4, University of St. Gallen.
23. Frentzos, E., Peleklis, N., Ntoutsi, I., & Theodoridis, Y. (2007). Trajectory database systems. In F. Giannotti & D. Pedreschi (Reds.), *Mobility, data mining and privacy – geographic knowledge discovery* (pp. 151-188). Berlijn: Springer.
24. Gama, J., & Cornuéjols, A. (2007). Resource aware distributed knowledge discovery. In M. May & L. Saitta, L. (Reds.), *Blueprint in ubiquitous knowledge discovery* (pp. 29-47).
25. Giannotti, F., & Pedreschi, D. (Reds.). (2007). *Mobility, data mining and privacy - geographic knowledge discovery*. Berlijn: Springer.
26. Giannotti, F., & Saygin, Y. (Reds.). (2007). Privacy and security in ubiquitous knowledge discovery. In M. May & L. Saitta, L. (Reds.), *Blueprint in ubiquitous knowledge discovery* (pp. 60-71).

27. Gülsen T. (2005). *Generalized inverse techniques in engineering*. Onuitgegeven afstudeerscriptie, Informatica Institute Istanbul Technical University.
28. Güting, R. H., et al. (2000). A foundation for representing and querying moving objects. *ACM Transactions Database Systems*, 25(1), 1-42.
29. Güting, R. H., & Schneider, M. (2005). *Moving objects databases*. San Francisco: Morgan Kaufman Publishers.
30. Hadjieleftheriou, M., Kollios, G., Gunopulos, D., & Tsotras, V. J. (2003). *On-line discovery of dense areas in spatio-temporal databases*. Paper gepresenteerd op het 8e International Symposium on Advances in Spatial and Temporal Databases (SSTD '03).
31. Hagerstrand, T. (1970). What about people in regional science? *Papers of the Regional Science Association*, 24, 7-21.
32. Hartikainen, P. (2007). *Usability of social ubiquitous computing*. Opgevraagd op 19 oktober 2008, van de volgende website:  
[http://www.hiit.fi/~oulasvir/58307110/ubic\\_pirkka\\_hartikainen\\_draft.pdf](http://www.hiit.fi/~oulasvir/58307110/ubic_pirkka_hartikainen_draft.pdf)
33. Hightower, J., & Borriello, G. (2001). Location systems for ubiquitous computing. *IEEE Computer magazine*, 34(8), 57-66.
34. Hightower, J., & Borriello, G. (2001). *Location sensing techniques*. UW CSE technisch rapport.
35. Hjelm, J. (2002). *Creating location services for the wireless web*. Londen: Wiley.
36. Hong, J. I., & Landay, J. A. (2004). *An architecture for privacy-sensitive ubiquitous computing*. Paper gepresenteerd op de 2e International Conference on Mobile Systems, Applications and Services (MobiSys '04), New York.
37. Hotho, A., et al. (2007). Ubiquitous Data. In M. May & L. Saitta, L. (Reds.), *Blueprint in ubiquitous knowledge discovery* (pp. 48-59).
38. Inmon, W. (1996). *Building the data warehouse*. Wiley.
39. IST. (2004). *Factsheet imagine-it*. Opgevraagd op 29 november 2008, van de volgende website:  
[http://ec.europa.eu/information\\_society/activities/esafety/doc/esafety\\_library/projects\\_fact\\_sheets/imagine\\_it.pdf](http://ec.europa.eu/information_society/activities/esafety/doc/esafety_library/projects_fact_sheets/imagine_it.pdf)
40. ITIS. (z.d.). *Cellular floating vehicle data*. Opgevraagd op 16 januari 2008, van de volgende website: <http://www.itisholdings.com/article.asp?id=296>
41. Kalnis, P., Mamoulis, N., & Bakiras, S. (2005). *On discovering moving clusters in spatio-temporal data*. Paper gepresenteerd op het 9e International Symposium on Spatial and Temporal Databases (SSTD '05).

42. Keßler C. (2006). *The revival of time geography*. Opgevraagd op 25 oktober 2008, van de volgende website: <http://www.gisblog.net/research/the-revival-of-time-geography/>
43. Koperski, K., & Han, J. (1995). *Discovery of spatial association rules in geographic information databases*. Paper gepresenteerd op het 4e International Symposium on Advances in Spatial Databases (SSD '95).
44. Lemmens, V. (2005). *Privacy preserving data mining*. Onuitgegeven afstudeerscriptie, Universiteit Hasselt, Diepenbeek.
45. Ley, D. (2007). Ubiquitous computing. *Emerging technologies for learning*, 2, 64-79.
46. Langheinrich, M. (2002, september). *Privacy invasions in ubiquitous computing*. Paper gepresenteerd op het Ubicomp 2002 congres, Göteborg.
47. Macedo, J., et al. (2007). Trajectory data models. In F. Giannotti & D. Pedreschi (Reds.), *Mobility, data mining and privacy – geographic knowledge discovery* (pp. 123-150). Berlijn: Springer.
48. Mahmoud, Q. (2004). J2ME and location-based services. *Sun Developer Network (SDN)*.
49. Manolopoulos, Y., Nanopoulos, A., Papadopoulos, A., & Theodoridis, Y. (2005). *Rtrees: theory and applications*. New York : Springer.
50. Mattern, F. (2001). The vision and foundations of ubiquitous computing. *Upgrade*, 2(5), 2-6.
51. Mattern, F. (2004). Ubiquitous computing: Scenarios for an informatized world. In A. Zerdick et al. (Reds.), *E-Merging Media* (pp. 155–174). New York: Springer.
52. Mattern, F. (2004). *Wireless future: ubiquitous computing*. Paper gepresenteerd op het Wireless Congress 2004, Zürich.
53. Mattsson, O. (2001). *Positioning of a cellular phone using the SIM*. Onuitgegeven afstudeerscriptie, Royal Institute of Technology, Stockholm.
54. May, M., & Saitta, L. (Reds.). (2007). *Blueprint in ubiquitous knowledge discovery*.
55. Miller, H. J. (1991). Modelling accessibility using space-time prism concepts within geographical information systems. *International Journal of Geographical Information Systems*, 5(3), 287-301.
56. Misra, P., & Enge, P. (2006). *Global positioning system – signals, measurements and performance*. Lincoln: Gange-Jamuna.

57. Moore, G. E. (1965). Cramming more components onto integrated circuits. *Electronics*, 38, 114-117.
58. Mutanen, T. (2007). *Consumer data and privacy in ubiquitous computing*. Onuitgegeven afstudeerscriptie, VTT Technical Research Centre of Finland.
59. Nanni, M., Kuijpers, B., Körner, C., May, M., & Pedreschi, D. (2007). Spatiotemporal data mining. In F. Giannotti & D. Pedreschi (Reds.), *Mobility, data mining and privacy – geographic knowledge discovery* (pp. 267-296). Berlijn: Springer.
60. Orlando, S., Orsini, R., Raffaetà, A., & Roncanto, A. (2007). Trajectory data warehouses: design and implementation issues. *Journal of Computing Science and Engineering*, 1(2), 211–232.
61. Openwave Developer Network. (2002). *Overview of location technologies*. Opgevraagd op 1 november 2008, van de volgende website:  
[http://developer.openwave.com/omtdocs/location\\_studio\\_sdk/pdf/Intro to Location Technologies.pdf](http://developer.openwave.com/omtdocs/location_studio_sdk/pdf/Intro_to_Location_Technologies.pdf)
62. Pedreschi, D., et al. (2007). Privacy protection: regulations and technologies, opportunities and threats. In F. Giannotti & D. Pedreschi (Reds.), *Mobility, data mining and privacy – geographic knowledge discovery* (pp. 101-119). Berlijn: Springer.
63. Pelekis, et al. (2007). Towards trajectory data warehouses. In F. Giannotti & D. Pedreschi (Reds.), *Mobility, data mining and privacy – geographic knowledge discovery* (pp. 189-212). Berlijn: Springer.
64. Pfoser, D., & Theodoridis, Y. (2000). *Generating semantics-based trajectories of moving objects*. Paper gepresenteerd op de International Workshop on Emerging Technologies for Geo-Based Applications.
65. Ranchordas, J., & Lenaghan, A. (2003). *A flexible framework for using positioning technologies in location-based services*. Paper gepresenteerd op de Eurocon 2003 conferentie, Ljubljana.
66. Renso, C., et al. (2007). Wireless network data sources: tracking and synthesizing trajectories. In F. Giannotti & D. Pedreschi (Reds.), *Mobility, data mining and privacy – geographic knowledge discovery* (pp. 73-100). Berlijn: Springer.
67. Rigaux, P., Scholl, M., & Voisard, A. (2001). *Spatial databases. With applications to GIS*. Morgan Kaufmann.

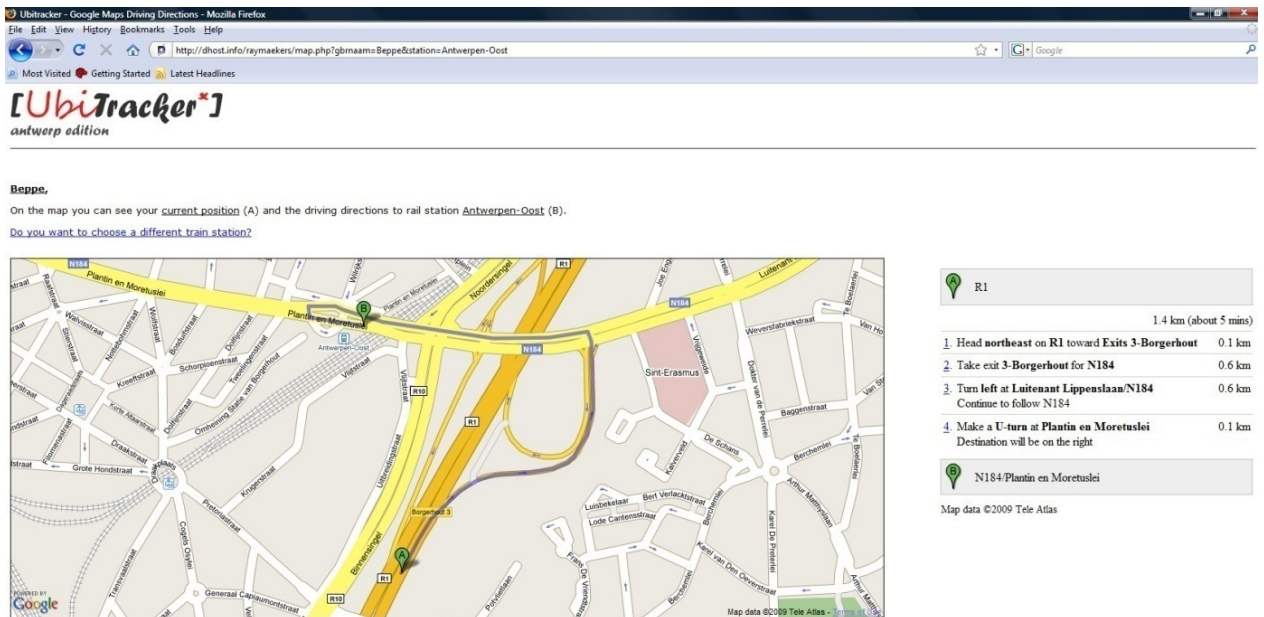
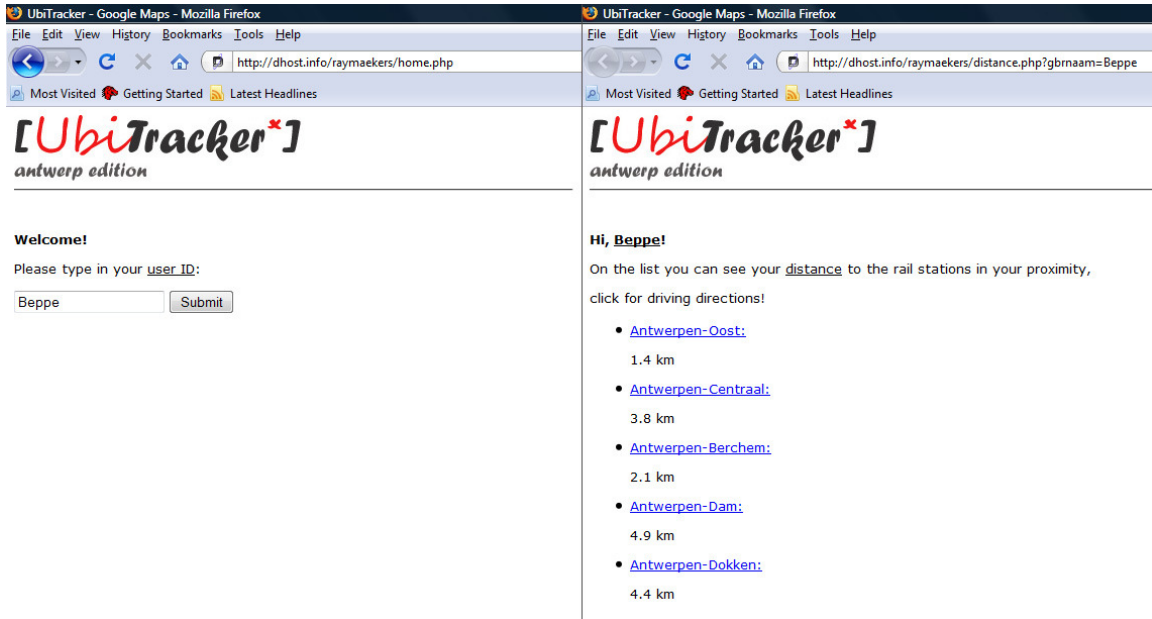
68. Rinzivillo, S., et al. (2007). Knowledge discovery from geographical data. In F. Giannotti & D. Pedreschi (Reds.), *Mobility, data mining and privacy – geographic knowledge discovery* (pp. 243-265). Berlijn: Springer.
69. Roadtraffic Technology. (z.d.). *LKW-MAUT electronic toll collection system for heavy goods vehicles, Germany*. Opgevraagd op 15 december 2008, van de volgende website: <http://www.roadtraffic-technology.com/projects/lkw-maut>
70. Schuster, A., & Wolff, R. (Reds.) (2007). Ubiquitous technologies. In M. May & L. Saitta, L. (Reds.), *Blueprint in ubiquitous knowledge discovery* (pp. 15-28).
71. Scourias, J. (1997). *Overview of the GSM cellular system*. Opgevraagd op 1 november 2008, van de volgende website:  
<https://styx.uwaterloo.ca/~jscouria/GSM/gsmreport.html>
72. Siegele, L. (2002). How about now? A survey of the real-time economy. *The Economist*, 362(8257), 3–18.
73. Strassner, M., & Schoch, T. (2002, augustus). *Today's impact of ubiquitous computing on business processes*. Paper gepresenteerd op de 1e International Conference on Pervasive Computing 2002, Zurich.
74. Swedberg, G. (1999). Ericssons mobile location solution. *Ericsson Review*, 4, 214-221.
75. Tanenbaum, A. S., & Van Steen, M. (2001). *Distributed systems: principles and paradigms*. New York: Prentice Hall.
76. Trajcevski, G. (2003). *Probabilistic range queries in moving objects databases with uncertainty*. Paper gepresenteerd op de Workshop on Data Engineering for Wireless and Mobile Access (MobiDE'03).
77. Trajcevski, G., Wolfson, O., Zhang, F., & Chamberlain, S. (2004). The geometry of uncertainty in moving objects databases. *ACM Transactions Database System*, 29(3), 463-507.
78. Vanhoof, K. (2007). Application areas. In M. May & L. Saitta, L. (Reds.), *Blueprint in ubiquitous knowledge discovery* (pp. 87-100).
79. Wang, S., Min, J., & Byung, K. (2008). *Location based services for mobiles: technologies and standards*. Paper gepresenteerd op de IEEE International Conference On Communication (ICC), Peking.
80. Weiser, M. (1991). The computer for 21st century. *Scientific American*, 265(9), 66–75.
81. Weiser, M., & Brown, J.S. (1996). *The coming age of calm technology*.

82. Witten, I. H., & Frank, E. (2000). *Data mining*. San Francisco: Morgan Kaufman Publishers.
83. Wolfson, O. (2002, juni). *Moving objects information management: the database challenge*. Paper gepresenteerd op de 5e Workshop on Next Generation Information Technologies and Systems, Caesarea.
84. Zeimpekis, V., Giaglis, G. M., Lekakos, G. (2003). A taxonomy of indoor and outdoor positioning techniques for mobile location services. *SIGecom Exchanges*, 3(4), 19-27.



# Bijlagen

## Bijlage A Online module van de UbiTracker applicatie



## Bijlage B Gebruikte codes

### Bijlage B.1 UbiTracker Java ME code

```
/**
 * UbiTracker.java
 */

import javax.microedition.lcdui.Form;
import javax.microedition.lcdui.Gauge;
import javax.microedition.lcdui.Spacer;
import javax.microedition.lcdui.ImageItem;
import javax.microedition.lcdui.TextField;
import javax.microedition.lcdui.DateField;
import javax.microedition.lcdui.StringItem;
import javax.microedition.lcdui.ChoiceGroup;
import javax.microedition.lcdui.Ticker;
import javax.microedition.lcdui.Image;
import javax.microedition.lcdui.Choice;
import javax.microedition.lcdui.Display;
import javax.microedition.lcdui.Command;
import javax.microedition.midlet.*;
import javax.microedition.lcdui.Displayable;
import javax.microedition.lcdui.CommandListener;
import java.io.*;
import javax.microedition.io.Connector;
import javax.microedition.io.HttpConnection;
import java.util.*;

/**
 * @author Beppe Raymaekers
 * @version Antwerp Edition
 */

public class UbiTracker extends MIDlet implements CommandListener, Runnable {

    private Form form, form2, form3, form4, form5, form6;
    private Ticker ticker1, ticker2, ticker3, ticker4;
    private Gauge gauge, gauge2;
    private Spacer spacer, spacer2, spacer3, spacer4;
    private ImageItem imageItem;
    private TextField txtField, txtField2, txtField3, txtField5, txtField6, txtField7, txtField8, txtField9,
    txtField10, txtField11, txtField12;
    private StringItem stringItem1, stringItem2, stringItem3, stringItem4, stringItem5;
    private String gbrnaam, vervoerskeuze, cellid, mcc, mnc, lac, info, url, url2, url3, serverstr, lat,
    stationinfo, latstation, lonstation, latstationtrim, lonstationtrim;
    private String lon, downloadCountString, runTimeString, leeg, url4, straat, straattrim, info2,
    info2trim, url11, res11, trimres11, url12, res12, trimres12;
    private int vevkeuze, aantal, downloadCount;
    private ChoiceGroup choiceGroup;
    private Timer timer;
    private Tracker tracker; //instantie van Tracker klasse waaronder de timertaak valt
    private Date currentTime, stopTime;
    private double runTime;
    private DateField datumveld1, datumveld2;
    private HttpConnection hcon = null;
    private DataOutputStream dos = null;
}
```

```
static final Command exitCommand = new Command("Exit", Command.EXIT, 0);
static final Command okCommand = new Command("Next", Command.OK, 0);
static final Command goCommand = new Command("Start", Command.OK, 0);
static final Command stopCommand = new Command("Stop", Command.STOP, 0);
static final Command mapCommand = new Command("Driving dir.", Command.OK, 2);
static final Command trackCommand = new Command("Track & trace", Command.OK, 0);
static final Command routeCommand = new Command("Route", Command.OK, 2);

public UbiTracker() {

    form = new Form("**UbiTracker**");

    // overlopende tekst
    ticker1 = new Ticker("!!Please enter your details!!");
    form.setTicker(ticker1);

    // progress bar
    gauge = new Gauge("Step 1 of 2", false, 2, 1);
    form.append(gauge);

    // spatie tussen de items
    spacer = new Spacer(20, 20);
    form.append(spacer);

    // Tekstfield
    txtField = new TextField("Enter your User ID: ", "", 20, TextField.ANY);
    form.append(txtField);

    spacer4 = new Spacer(20, 20);
    form.append(spacer4);

    //Afbeelding
    try {
        Image img = Image.createImage("/UbiTrackerALogo.JPG");
        ImageItem imgItem = new ImageItem("", img,
        ImageItem.LAYOUT_NEWLINE_AFTER |
        ImageItem.LAYOUT_CENTER, null);
        form.append(imgItem);
    }

    catch (Exception e) {
        e.printStackTrace();
    }

    // Timer & Tracker
    timer = new Timer();
    tracker = new Tracker();
}

public void startApp() throws MIDletStateChangeException {

    Display display = Display.getDisplay(this);
    form.addCommand(exitCommand);
    form.addCommand(okCommand);
    form.setCommandListener(this);
    display.setCurrent(form);
}

public void pauseApp() {
}

public void destroyApp(boolean unconditional) {
    notifyDestroyed();
}

public void processForm() { //actie Next Button
```

```
// we starten een nieuwe form voor de tweede stap
form2 = new Form("**UbiTracker**");

// overlopende tekst
ticker2 = new Ticker("!!Please enter your details!!");
form2.setTicker(ticker2);

// progress bar
gauge2 = new Gauge("Step 2 of 2", false, 2, 2);
form2.append(gauge2);

// spatie
spacer2 = new Spacer(20, 20);
form2.append(spacer2);

// keuzelijst
choiceGroup = new ChoiceGroup("Looking for: ", Choice.EXCLUSIVE,
    new String[] {"Train stations", "Bus stops", "Gas stations", "Parking lots"},
    null);
form2.append(choiceGroup);

// buttons toevoegen aan 2e pagina
form2.addCommand(exitCommand);
form2.addCommand(goCommand);
form2.setCommandListener(this);

Display display2 = Display.getDisplay(this);
display2.setCurrent(form2);
}

public void processForm2() { // voor de Start button

// we starten een nieuwe form voor de eigenlijke applicatie
form3 = new Form("**UbiTracker**");

// buttons toevoegen aan 3e form
form3.addCommand(stopCommand);
form3.setCommandListener(this);

// overlopende tekst
ticker3 = new Ticker("!!Tracking!!");
form3.setTicker(ticker3);

// Tekst en date fields
datumveld1 = new DateField("", DateField.DATE_TIME); //"Start Time",
form3.append(datumveld1);

txtField2 = new TextField("", "", 100, TextField.UNEDITABLE);
form3.append(txtField2);

txtField3 = new TextField("Street: ", "", 100, TextField.UNEDITABLE);
form3.append(txtField3);

txtField9 = new TextField("Additional info: ", "", 100, TextField.UNEDITABLE);
form3.append(txtField9);

//weergeven op scherm
Display display3 = Display.getDisplay(this);
display3.setCurrent(form3);

// Start Time weergeven
currentTime = new Date();
datumveld1.setDate(currentTime);

// Timer aanroepen (inclusief uit te voeren taak, tracker hier)
timer.scheduleAtFixedRate(tracker, 0, 10000);
```

```
} //van processForm2
```

```
private class Tracker extends TimerTask {
```

```
    public final void run() {
```

```
        //werkt enkel op SonyEricsson telefoons
        cellid = System.getProperty("com.sonyericsson.net.cellid");
        mcc = System.getProperty("com.sonyericsson.net.cmcc");
        mnc = System.getProperty("com.sonyericsson.net.cmnc");
        lac = System.getProperty("com.sonyericsson.net.lac");

        // test!!!!
        //mcc="206";mnc="20";cellid="931";lac="206"; //A'pen
        //mcc="206";mnc="20";cellid="621c";lac="00f2"; //Thuis
        //mcc="206";mnc="20";cellid="7991";lac="201"; //A'pen 2
        //mcc="206";mnc="20";cellid="2124";lac="206"; //A'pen 3
        //mcc="206";mnc="20";cellid="743";lac="201"; //A'pen 4
        //mcc="206";mnc="20";cellid="746";lac="201"; //A'pen 5 Dokken
```

```
        if(cellid!=null){
```

```
            info="Cell:"+cellid+" mcc:"+mcc+" mnc:"+mnc+" lac:"+lac; // om op
            scherm te zetten
```

```
            url="http://www.opencellid.org/cell/get?key=18d6cb043c526331240207c6d
            da7a76a&mcc="+mcc+"&mnc="+mnc+"&lac="+lac.toString()+"&cellid="+c
            ellid.toString()+"&fmt=txt";
```

```
            // clearen eerste textField (pos)
            txtField2.setString("");
```

```
            // Zet Cell Info in het textfield
            //txtField2.insert(info, 0);
```

```
            txtField2.setString("Requesting position...");
            txtField3.setString("Requesting street...");
            txtField9.setString("Requesting info...");
```

```
            // aanroepen DB Connector
            DBConnectie();
```

```
        } // van if
```

```
    } // van run (timertask)
```

```
} // van Tracker
```

```
public void DBConnectie() {
```

```
    Thread t=new Thread(this);
    t.start(); // public void run() wordt aangeroepen
```

```
}
```

```
public void run(){
```

```
    // connectie leggen met openCellID + eigen db
```

```
    try {
        HttpURLConnection cnx = (HttpURLConnection)Connector.open(url);
        InputStream is=cnx.openInputStream();
        StringBuffer b=new StringBuffer();
        int car;
        while( (car=is.read())!= -1){
            b.append((char)car);
        }
        is.close();
        cnx.close();
        String res=b.toString();
```

```
if(res.startsWith("err")){
    txtField2.setString("Cell not found!");

    //indien we geen locatie kunnen vaststellen, schrijven we dit weg naar db
    lat = "/";
    lon = "/";
}

//hier worden de strings overschreven indien toch een locatie wordt gevonden
else {
    int pos=res.indexOf(',');
    lat=res.substring(0,pos);
    int pos2=res.indexOf(',',pos+1);
    lon=res.substring(pos+1,pos2);
    downloadCount = downloadCount + 1; //om te tellen hoeveel keer hij
    //connectie legt met openCellID

    //om benaderende locaties eruit te filteren
    if(lat.equals("50.8583946807275")) {
        lat = "Location";
        lon = "not";
        txtField2.setString(lat+" "+lon+" available");
    }
    else if(lat.equals("0.0")) {
        lat = "Location";
        lon = "not";
        txtField2.setString(lat+" "+lon+" available");
    }
    else {
        txtField2.setString(lat+" "+lon);
    }
}

}

catch (IOException ex) {
    ex.printStackTrace();
    txtField3.setString(ex.toString());
}

try {
    //url3 vullen
    url3="http://dhost.info/raymaekers/streetandinfo.php?mcc="+mcc+"&mnc="+mnc
    +"&lac="+lac.toString()+"&cellid="+cellid.toString();
    txtField3.setString("");

    HttpConnection cnx = (HttpConnection)Connector.open(url3);
    InputStream is=cnx.openInputStream();
    StringBuffer b=new StringBuffer();
    int car;
    while( (car=is.read())!= -1){
        b.append((char)car);
    }
    is.close();
    cnx.close();
    String res=b.toString();

    int pos=res.indexOf(',');
    straat=res.substring(0,pos);
    int pos2=res.indexOf(',',pos+1);
    info2=res.substring(pos+1,pos2);

    straattrim = straat.trim();
    info2trim = info2.trim();
    //txtfielden clearen
    txtField3.setString("");
}
```

```
txtField9.setString("");

if (straattrim.equals(leeg)) {
    txtField3.setString("No info available");
}
else {
    txtField3.insert(straattrim, 0);
}

if (info2trim.equals(leeg)) {
    txtField9.setString("No info available");
}
else {
    txtField9.insert(info2trim, 0);
}

}

catch(IOException ex) {
    ex.printStackTrace();
    txtField3.setString(ex.toString());
}

try {
    //url2 vullen

    url2="http://dhost.info/raymaekers/UbiTracker.php?gbrnaam="+gbrnaam+
"&vervkeuze="+vervoerskeuze+"&lat="+lat+"&lon="+lon+"&mcc="+mcc+"&mnc="+mnc+
&lac="+lac.toString()+"&cellid="+cellid.toString();

    //the message string that you will send to the server
    serverstr = "";

    //open a connection to the specific URL
    hcon = ( HttpURLConnection )Connector.open
(url2);

    // set the request method to POST
    hcon.setRequestMethod( HttpURLConnection.POST );

    //set the content length of the message that you want to send
    hcon.setRequestProperty("Content-Length", Integer.toString(serverstr.length()));

    // obtain DataOutputStream for sending the request string
    dos = hcon.openDataOutputStream();

    // send request string to server

    //send the message through the dataoutputstream
    dos.write(serverstr.getBytes());

    hcon.close();
    dos.close();
}

catch(IOException ex) {
    ex.printStackTrace();
    txtField3.setString(ex.toString());
}
} // van run

public void processForm3() { //voor de Stop button

    timer.cancel();
    // berekenen looptijd
    stopTime = new Date();
```

```
runTime = stopTime.hashCode() - currentTime.hashCode();
runTime = runTime/1000; //aangezien dit miliseconden zijn
runTimeString = Double.toString(runTime);

// we starten een nieuwe form
form4 = new Form("**UbiTracker**");

// Buttons toevoegen aan 4e form
form4.addCommand(exitCommand);
form4.addCommand(mapCommand);
form4.addCommand(trackCommand);
form4.setCommandListener(this);

datumveld2 = new DateField("", DateField.DATE_TIME); //Stop Time,
form4.append(datumveld2);
datumveld2.setDate(stopTime);

// overlopende tekst
ticker4 = new Ticker("!!!Done Tracking!!!");
form4.setTicker(ticker4);

//txtField6 = new TextField("URL: ", "", 200, TextField.UNEDITABLE);
//form4.append(txtField6);
//txtField6.setString(url2);

txtField7 = new TextField("Run time: ", "", 200, TextField.UNEDITABLE);
form4.append(txtField7);
txtField7.setString(runTimeString+" sec");

txtField8 = new TextField("# Down & uploads: ", "", 200, TextField.UNEDITABLE);
form4.append(txtField8);
downloadCountString = Integer.toString(downloadCount);
txtField8.setString(downloadCount+" times");

txtField10 = new TextField("Closest train station: ", "", 200, TextField.UNEDITABLE);
form4.append(txtField10);
txtField10.setString(stationinfo);

Display display4 = Display.getDisplay(this);
display4.setCurrent(form4);
} //van Processform3

// handle commands
public void commandAction(Command cmd,Displayable disp) {

    String cmdLabel = cmd.getLabel();
    leeg = "";

    if (cmdLabel.equals("Stop")) { //om de stationsinfo op te vragen

        if (lat.equals("Location")) {

            stationinfo = "No info available";
        }
        else {
            url = "http://dhost.info/raymaekers/station.php?lat="+lat+"&lon="+lon;

            try {
                HttpURLConnection cnx = (HttpURLConnection)Connector.open(url);
                InputStream is=cnx.openInputStream();
                StringBuffer b=new StringBuffer();
                int car;
                while( (car=is.read())!= -1){
                    b.append((char)car);
                }
            }
        }
    }
}
```



```
is.close();
cnx.close();
String res=b.toString();

    int pos=res.indexOf(',');
    latstation=res.substring(0,pos);
    int pos2=res.indexOf(',',pos+1);
    lonstation=res.substring(pos+1,pos2);
    int pos3=res.indexOf(',',pos2+1);
    stationinfo=res.substring(pos2+1,pos3);
}

    catch (IOException ex) {
    ex.printStackTrace();
    txtField3.setString(ex.toString());
    }
} // van else

processForm3();
form3.setTicker(ticker3);
} // van if
else if (cmdLabel.equals("Exit")) {
    destroyApp(true);
}
else if (cmdLabel.equals("Next")) {

    // inhoud van ID naar variabele wegschrijven
    gbrnaam = txtField.getString();

    if (gbrnaam.equals(leeg)) {
        spacer3 = new Spacer(10, 10);
        form.append(spacer3);
        //stringItem4 = new StringItem("ERROR: User ID is empty!!", "");
        //form.append(stringItem4);
        txtField.setString("ERROR: No user ID!!");
    }
    else {
        processForm();
    }
}
else if (cmdLabel.equals("Start")) {

    // vervoersmiddel variabele inlezen
    vervkeuze = choiceGroup.getSelectedIndex();
    vervoerskeuze = Integer.toString(vervkeuze);

    downloadCount = 0; //initialisatie teller

    processForm2();
}
else if (cmdLabel.equals("Track & trace")) {
    //we starten een nieuwe form voor track & trace
    form5 = new Form("**Track & Trace**");
    form5.addCommand(exitCommand);
    form5.addCommand(mapCommand);
    form5.setCommandListener(this);
    Display display = Display.getDisplay(this);
    display.setCurrent(form5);

    try {
        url11 = "http://dhost.info/raymaekers/trackandtrace.php?gbrnaam="
            +gbrnaam;
        HttpConnection cnx11 = (HttpConnection)Connector.open(url11);
        InputStream is11=cnx11.openInputStream();
        StringBuffer b11=new StringBuffer();
        int car11;
```

```
        while( (car11=is11.read())!= -1){
            b11.append((char)car11);
        }
        is11.close();
        cnx11.close();
        res11=b11.toString();
        trimres11 = res11.trim();
    }
    catch (Exception e) {
        e.printStackTrace();
    }
}

try { // voor map
    url11 = trimres11;
    DataInputStream is = null;
    StringBuffer sb = new StringBuffer();
    Image img= null;

    HttpConnection c = (HttpConnection) Connector.open(url11);
    int len = (int)c.getLength();

    is = c.openDataInputStream();
    byte[] data = new byte[len];
    is.readFully(data);
    img = Image.createImage(data, 0, len);

    ImageItem imgItem = new ImageItem("", img,
    ImageItem.LAYOUT_NEWLINE_AFTER |
    ImageItem.LAYOUT_CENTER, null);
    form5.append(imgItem);

    is.close();
    c.close();
}
catch (Exception e) {
    e.printStackTrace();
}
}

else if (cmdLabel.equals("Route")) {

    //we starten een nieuwe form voor de route
    Form form6 = new Form("**Route**");
    form6.addCommand(exitCommand);
    form6.addCommand(mapCommand);
    form6.setCommandListener(this);
    TextField12 = new TextField("", "", 300, TextField.UNEDITABLE);
    form6.append(TextField12);
    Display display = Display.getDisplay(this);
    display.setCurrent(form6);
    url12 ="http://dhost.info/raymaekers/xmlread.php?lat="+lat+"&lon="+lon+"&lat2
    =" +latstationtrim+"&lon2="+lonstationtrim;

    try {
        HttpConnection cnx = (HttpConnection)Connector.open(url12);
        InputStream is=cnx.openInputStream();
        StringBuffer b=new StringBuffer();
        int car;
        while( (car=is.read())!= -1){
            b.append((char)car);
        }
        is.close();
        cnx.close();
        res12 =b.toString();
        trimres12 = res12.trim();
        TextField12.insert(trimres12, 0);
    }
}
```

```
        catch (IOException ex) {
            ex.printStackTrace();
            txtField3.setString(ex.toString());
        }
    }
    else if (cmdLabel.equals("Driving dir.")) {

        //we starten een nieuwe form voor de map
        Form gmap = new Form("**Driving Directions**");
        gmap.addCommand(exitCommand);
        gmap.addCommand(trackCommand);
        gmap.addCommand(routeCommand);
        gmap.setCommandListener(this);
        Display display = Display.getDisplay(this);
        display.setCurrent(gmap);

        if (lat.equals("Location")) {
            stringItem5 = new StringItem("ERROR: No map available!!", "");
            gmap.append(stringItem5);
        }
        else {
            latstationtrim=latstation.trim();
            lonstationtrim=lonstation.trim();

            url = "http://developer.multimap.com/API/map/1.2/OA090325115924825
72?routeKey=DC,"+lon+" "+lat+" "+lonstationtrim+" "+latstationtrim+"
,0&width=240& height=237&routeColor=0000FF,0,0&routeOpacity=1";

            DataInputStream is = null;
            StringBuffer sb = new StringBuffer();
            Image img= null;

            try {
                HttpConnection c = (HttpConnection) Connector.open(url);
                int len = (int)c.getLength();

                is = c.openDataInputStream();
                byte[] data = new byte[len];
                is.readFully(data);
                img = Image.createImage(data, 0, len);

                ImageItem imgItem = new ImageItem("", img,
                ImageItem.LAYOUT_NEWLINE_AFTER |
                ImageItem.LAYOUT_CENTER, null);
                gmap.append(imgItem);

                is.close();
                c.close();
            }

            catch (Exception e) {
                e.printStackTrace();
            }
            //van else
        } //van else if
    }
} //van ubitracker
```

## *Bijlage B.2 Db script*

```
<?php
$dbhost = 'mysql';
$dbusername = 'raymaekers';
$dbpasswd = '*****';
$database_name = 'raymaekers';
$connection = mysql_connect("$dbhost", "$dbusername", "$dbpasswd")
    or die ("Couldn't connect to server.");

$db = mysql_select_db("$database_name", $connection)
    or die("Couldn't select database.");
?>
```

## *Bijlage B.3 Streetandinfo script*

```
<?php

// DB GEGEVENS
include 'db.php';

// Ophalen gegevens uit header
$mcc = $_GET["mcc"];
$mnc = $_GET["mnc"];
$lac = $_GET["lac"];
$cellid = $_GET["cellid"];

$sql = "SELECT * FROM cellinfo WHERE (cellid='$cellid' AND mcc='$mcc' AND mnc='$mnc' AND lac='$lac')";

    $result = mysql_query($sql);
    if (!$result) {
        echo ("");
        exit;
    }

    $rij = mysql_fetch_array($result);
    $straat = $rij["straat"];
    $info = $rij["info"];

    echo ($straat);
    echo (",");
    echo ($info);
    echo (",");
?>
```

## *Bijlage B.4 UbiTracker script*

```
<?php
```

```
// DB GEGEVENS
include 'db.php';

// Ophalen gegevens uit header
$gbrnaam = $_GET["gbrnaam"];
$vervkeuze = $_GET["vervkeuze"];
$lat = $_GET["lat"];
$lon = $_GET["lon"];
$mcc = $_GET["mcc"];
$mnc = $_GET["mnc"];
$lac = $_GET["lac"];
$cellid = $_GET["cellid"];

$sql2 = "INSERT INTO ubitracker (gbrnaam, vervkeuze, lat, lon, mcc, mnc, lac, cellid) VALUES ('$gbrnaam',
'$vervkeuze', '$lat', '$lon', '$mcc', '$mnc', '$lac', '$cellid')";

$result2 = mysql_query($sql2);
?>
```

## *Bijlage B.5 Station script*

```
<?php
// DB GEGEVENS
include 'db.php';

// Ophalen gegevens uit header
$lat1 = $_GET["lat"];
$lon1 = $_GET["lon"];

$kleinste = 1000; //initialisatie

$sql = "SELECT * FROM cellinfo WHERE (info='Station')";

$result = mysql_query($sql);

if (!$result) {
echo ("");
exit;
}

$num_rows = mysql_num_rows($result);

for ( $i = 1; $i <= $num_rows; $i += 1) {

    $rij = mysql_fetch_array($result);
    $lat2 = $rij["lat"];
    $lon2 = $rij["lon"];
    $info2 = $rij["info2"];

    $afstand = (sqrt((69.1*($lat1 - $lat2))*(69.1*($lat1 - $lat2)) + (53*($lon1 -
$lon2))*(53*($lon1 - $lon2)))*(1.609344));

    if ($afstand <= $kleinste) {
        $kleinste = $afstand;
        $station = $info2;
        $latstation = $lat2;
        $lonstation = $lon2;
    }
}
}
```

```
echo ($latstation);
echo (",");
echo ($lonstation);
echo (",");
echo ($station);
echo (",");
echo ("6000");
?>
```

## *Bijlage B.6 Trackandtrace script*

```
<?php
// DB GEGEVENS
include 'db.php';

// Ophalen gegevens uit header
$gbrnaam = $_GET["gbrnaam"];

$url="http://maps.google.com/staticmap?size=240x273&maptype=mobile%27&markers=";
//235 voor gsm
echo ($url);

$sql = "SELECT * FROM ubitracker WHERE gbrnaam='$gbrnaam' ORDER BY starttime ASC";

$result = mysql_query($sql);

if (!$result) {
echo ("");
exit;
}

$num_rows = mysql_num_rows($result);

for ( $i = 1; $i <= $num_rows-1; $i += 1) {

    $rij = mysql_fetch_array($result);
    $lat = $rij["lat"];
    $lon = $rij["lon"];
    echo ($lat);
    echo (",");
    echo ($lon);
    echo (",blue");
    echo ($i);
    echo ("|");
}

$sql1 = "SELECT * FROM ubitracker WHERE gbrnaam = '$gbrnaam' ORDER BY starttime DESC";

$result1 = mysql_query($sql1);

if (!$result1) {
echo "Kon de query1 ($sql1) niet uitvoeren: " . mysql_error();
exit;
}

$rij1 = mysql_fetch_array($result1);

// current position
$lat1 = $rij1["lat"];
```

```
$lon1 = $rij1["lon"];

echo ($lat1);
echo ("");
echo ($lon1);
echo ("");
echo ("red");
echo ($num_rows);
echo ("&key=ABQIAAAAtGxy7yLxP6BX14gpWnQisBT2K
      ZN2-Qk79l60UKwojJ9fFFTUwBTGxNTYKPJz427zc8HN6ilt05y2NA&sensor=true");
?>
```

## *Bijlage B.7 XMLread script*

```
<?php

// Ophalen gegevens
$lat1 = $_GET["lat1"];
$lon1 = $_GET["lon1"];
$lat2 = $_GET["lat2"];
$lon2 = $_GET["lon2"];

$file = "http://clients.multimap.com/API/route/1.2/OA09032511592482572?lat_1=" . $lat1 . "&lon_1=" . $lon1
      . "&lat_2=" . $lat2 . "&lon_2=" . $lon2 . "&mode=driving";
$xml = simplexml_load_file($file);

foreach($xml->children() as $child)
{
    foreach($child->children() as $child2)
    {
        $naam1 = $child2->getName();
        if ($naam1 == "Minutes")
        {
            echo $child2->getName() . ": " . $child2 . "\n";
        }
        if ($naam1 == "KM")
        {
            echo $child2->getName() . ": " . $child2 . "\n";
        }
        foreach($child2->children() as $child3)
        {
            foreach($child3->children() as $child4)
            {
                foreach($child4->children() as $child5)
                {
                    $naam = $child5->getName();

                    if ($naam == "Instruction")
                    {
                        echo $child5;
                    }
                    if ($naam == "RoadName")
                    {
                        if ($child5 == "") {}
                        else {
                            echo " " . $child5;
                        }
                    }
                    if ($naam == "RoadNumber")
                }
            }
        }
    }
}
```

```
        {
            if ($child5 == "") {
                echo "\n";
            }
            else {
                echo " " . $child5 . "\n";
            }
        }
    }
}
}
?>
```

## *Bijlage B.8 Home script*

```
<html>
<head>
  <title>UbiTracker - Google Maps</title>
</head>
<body>
  
  <hr />
  
  <h1 style="font-family:verdana; color:black; font-size:13px;">Welcome!</h1>
  <p style="font-family:verdana; color:black; font-size:13px;">Please type in your <u>user ID</u>: </p>
  <form name="input" action="http://dhost.info/raymaekers/distance.php?" method="get">
  <input type="text" name="gbrnaam" value="" size="20">
  <input type="submit" value="Submit">
  </form>
</body>
</html>
```

## *Bijlage B.9 Distance script*

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<title>UbiTracker - Google Maps</title>

<?php
// DB GEGEVENS
include 'db.php';

// Ophalen gegevens db
$gbrnaam = $_GET["gbrnaam"];

// Query1
$sql1 = "SELECT * FROM ubitracker WHERE gbrnaam = '$gbrnaam' ORDER BY starttime DESC";
```



```
$result1 = mysql_query($sql1);

if (!$result1) {
echo "Kon de query1 ($sql1) niet uitvoeren: " . mysql_error();
exit;
}

$rrij1 = mysql_fetch_array($result1);

// current position
$lat1 = $rij1["lat"];
$lon1 = $rij1["lon"];
?>

<script type="text/javascript"
src="http://www.google.com/jsapi?key=ABQIAAAAtGxy7yLxP6BX14gpWnQisBT2KZN2-
Qk79l60UKwojJ9FFFTUwBTGxNTYKPJz427zc8HN6ilt05y2NA"></script>
<script type="text/javascript">
//
google.load("maps", "2");
function load() {
    if (GBrowserIsCompatible()) {
        gdir = new google.maps.Directions();
        gdir2 = new google.maps.Directions();
        gdir3 = new google.maps.Directions();
        gdir4 = new google.maps.Directions();
        gdir5 = new google.maps.Directions();
        google.maps.Event.addListener(gdir, "load", handleLoad);
        google.maps.Event.addListener(gdir2, "load", handleLoad);
        google.maps.Event.addListener(gdir3, "load", handleLoad);
        google.maps.Event.addListener(gdir4, "load", handleLoad);
        google.maps.Event.addListener(gdir5, "load", handleLoad);
        gdir.load("from: &lt;?php echo ($lat1); ?&gt;, &lt;?php echo ($lon1); ?&gt; to:
51.2082527777700,4.4381611111000", {getSteps: true});
        gdir2.load("from: &lt;?php echo ($lat1); ?&gt;, &lt;?php echo ($lon1); ?&gt; to:
51.216086111111,4.421030555556", {getSteps: true});
        gdir3.load("from: &lt;?php echo ($lat1); ?&gt;, &lt;?php echo ($lon1); ?&gt; to:
51.199722222222,4.432294444444", {getSteps: true});
        gdir4.load("from: &lt;?php echo ($lat1); ?&gt;, &lt;?php echo ($lon1); ?&gt; to:
51.232058333333,4.425827777778", {getSteps: true});
        gdir5.load("from: &lt;?php echo ($lat1); ?&gt;, &lt;?php echo ($lon1); ?&gt; to:
51.230369444444,4.430919444444", {getSteps: true});
    }
}
function handleLoad() {
    document.getElementById("oost").innerHTML = gdir.getDistance().html;
    document.getElementById("centraal").innerHTML = gdir2.getDistance().html;
    document.getElementById("berchem").innerHTML = gdir3.getDistance().html;
    document.getElementById("dam").innerHTML = gdir4.getDistance().html;
    document.getElementById("dokken").innerHTML = gdir5.getDistance().html;
}
window.onload = load;
window.onunload = google.maps.Unload;
//]]&gt;

&lt;/script&gt;
&lt;/head&gt;

&lt;body&gt;
&lt;img src="http://dhost.info/raymaekers/UbiTrackerALogo.jpg" /&gt;
&lt;hr /&gt;
&lt;img src="http://dhost.info/raymaekers/space.jpg" /&gt;
&lt;h1 style="font-family:verdana; color:black; font-size:13px;"&gt;Hi, &lt;u&gt;&lt;?php echo ($gbrnaam);
?&gt;&lt;/u&gt;!&lt;/h1&gt;</pre></div>
```

```
<p style="font-family:verdana; color:black; font-size:13px;">On the list you can see your <u>distance</u> to
the rail stations in your proximity, </p>
<p style="font-family:verdana; color:black; font-size:13px;">click for driving directions! </p>
<ul>
<li><p style="font-family:verdana; color:black; font-size:13px;"><a
href="http://dhost.info/raymaekers/map.php?gbrnaam=<?php echo ($gbrnaam); ?>&station=Antwerpen-
Oost">Antwerpen-Oost:</a> </p> <div style="font-family:verdana; color:black; font-size:13px;",
id="oost"></div></li>
<li><p style="font-family:verdana; color:black; font-size:13px;"><a
href="http://dhost.info/raymaekers/map.php?gbrnaam=<?php echo ($gbrnaam); ?>&station=Antwerpen-
Centraal">Antwerpen-Centraal:</a> </p> <div style="font-family:verdana; color:black; font-size:13px;",
id="centraal"></div></li>
<li><p style="font-family:verdana; color:black; font-size:13px;"><a
href="http://dhost.info/raymaekers/map.php?gbrnaam=<?php echo ($gbrnaam);
?>&station=Berchem">Antwerpen-Berchem:</a> </p> <div style="font-family:verdana; color:black; font-
size:13px;", id="berchem"></div></li>
<li><p style="font-family:verdana; color:black; font-size:13px;"><a
href="http://dhost.info/raymaekers/map.php?gbrnaam=<?php echo ($gbrnaam); ?>&station=Antwerpen-
Dam">Antwerpen-Dam:</a> </p> <div style="font-family:verdana; color:black; font-size:13px;",
id="dam"></div></li>
<li><p style="font-family:verdana; color:black; font-size:13px;"><a
href="http://dhost.info/raymaekers/map.php?gbrnaam=<?php echo ($gbrnaam); ?>&station=Antwerpen-
Dokken">Antwerpen-Dokken:</a> </p> <div style="font-family:verdana; color:black; font-size:13px;",
id="dokken"></div></li>
</ul>
</body>
</html>
```

## *Bijlage B.10 Map script*

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:v="urn:schemas-microsoft-com:vml">
<head>
<title>Ubitracker - Google Maps Driving Directions</title>
<meta http-equiv="content-type" content="text/html; charset=UTF-8"/>

<?php

// DB GEGEVENS
include 'db.php';

// Ophalen gegevens
$gbrnaam = $_GET["gbrnaam"];
$station = $_GET["station"];

// Query1
$sql1 = "SELECT * FROM ubitracker WHERE gbrnaam = '$gbrnaam' ORDER BY starttime DESC";

$result1 = mysql_query($sql1);

if (!$result1) {
echo "Kon de query1 ($sql1) niet uitvoeren: " . mysql_error();
exit;
}

$rrij1 = mysql_fetch_array($result1);
// current position
$lat1 = $rrij1["lat"];
```

```
$lon1 = $rij1["lon"];

// Query2

$sql = "SELECT * FROM cellinfo WHERE (info2='$station)";

    $result = mysql_query($sql);

    if (!$result) {
        echo ("");
        exit;
    }

    $rij = mysql_fetch_array($result);
    $lat2 = $rij["lat"];
    $lon2 = $rij["lon"];
    $latstationtrim = trim($lat2);
    $lonstationtrim = trim($lon2);

?>

<script
src="http://maps.google.com/maps?file=api&v=2.x&key=ABQIAAAAtGxy7yLxP6BX14gpWnQisBT2KZN2-
Qk79l60UKwojJ9fFTUwBTGxNTYKPJz427zc8HN6ilt05y2NA"
    type="text/javascript"></script>
<script type="text/javascript">
    // Create a directions object and register a map and DIV to hold the
    // resulting computed directions

var map;
var directionsPanel;
var directions;

function initialize() {
    map = new GMap2(document.getElementById("map_canvas"));
    map.setCenter(new GLatLng(51.2097320556641, 4.42310667037964), 14);
    directionsPanel = document.getElementById("route");
    directions = new GDirections(map, directionsPanel);
    directions.load("from: <?php echo ($lat1); ?>, <?php echo ($lon1); ?> to: <?php echo ($latstationtrim);
?>, <?php echo ($lonstationtrim); ?>");
}
</script>
</head>

<body onload="initialize()">

<hr />


<h1 style="font-family:verdana; color:black; font-size:13px;"><u><?php echo ($gbrnaam); ?></u></h1>
<p style="font-family:verdana; color:black; font-size:13px;"> On the map you can see your <u>current
position</u> (A) and the driving directions to rail station <u><?php echo ($station); ?></u> (B).</p>
<div style="font-family:verdana; color:black; font-size:13px;"><a
href="http://dhost.info/raymaekers/distance.php?gbrnaam=<?php echo ($gbrnaam); ?>">Do you want to
choose a different train station?</a> </div>

<div id="map_canvas" style="width: 70%; height: 480px; float:left; border: 1px solid black;"></div>
<div id="route" style="width: 25%; height:480px; float:right; border: 1px solid black;"></div>
<br/>
</body>
</html>
```