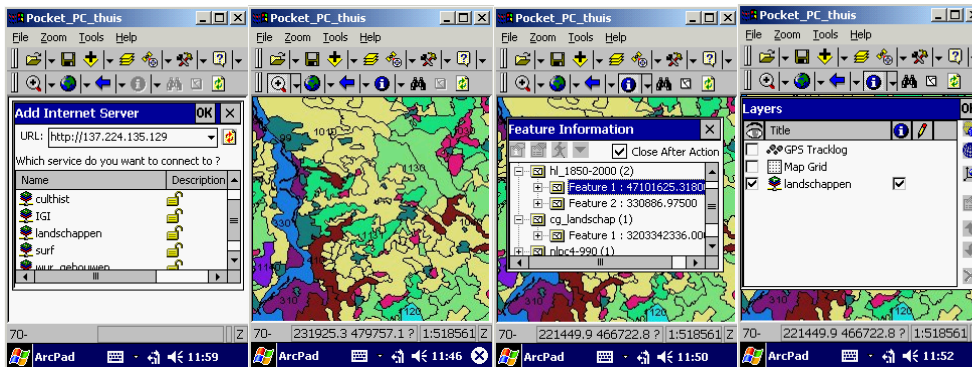




Kaart zonder draad

Voorstudie draadloze geo-informatie



Versie: 2.0 mei 2002
 Auteurs: Arend Ligtenberg, Ron van Lammeren, Jan Dirk Bulens, Aldo Bergsma
 Project: **GIPSY** 3^e projectenronde (ICT 2002/3) SURFtender 2001 ICT en Onderwijs,

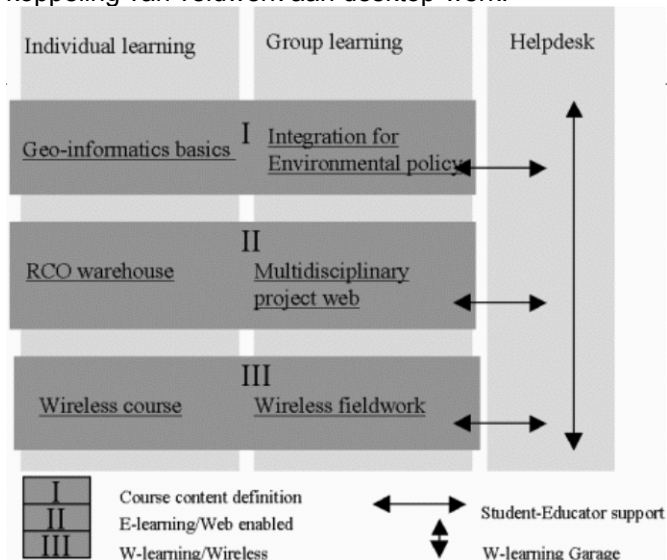
Voorwoord

In januari 2002 startte het SURF Educatie<F> GIPSY-project via een samenwerkingsverband tussen Katholieke Universiteit Nijmegen (Centrum voor Milieustudies), Vrije Universiteit Amsterdam (SpinLab) en Wageningen Universiteit (Centrum voor Geo-Informatie).

De projectnaam GIPSY is een knipoog naar het niet tijd- en plaatsgebonden leven en leren van de toekomstige student. Uit onderwijsevaluatie bleek dat studenten een toenemende behoefte hebben om binnen opleidingen eigen leertrajecten (naar inhoud en tijd) te kunnen doorlopen. In het GIPSY project is het uitgangspunt dan ook dat ICT wel degelijk kans biedt voor dergelijke, individuele leertrajecten, waarbij zelfs de plaats- (“pak je school op en leer”) en tijdsafhankelijkheid (“24-uurs leren”) tussen onderwijsvrager en –aanbieder kan verdwijnen. Daar tegenover staat de maatschappelijke vraag naar assemblage van kennis, bij voorkeur op multi- of interdisciplinaire wijze. Juist via bundeling van kennis, vaardigheden en houdingen kunnen nieuwe kennis, vaardigheden en houdingen worden geleerd. ICT biedt ook in dit opzicht mogelijkheden om teamwerk te bevorderen.

Het GIPSY-project speelt in op beide vragen en richt zich daarbij met name op het terrein van universitaire opleidingen in het kader van milieubeheer, natuurbeheer en omgevingsbeleid. In ieder van deze opleidingen neemt de vraag naar het gebruik van geografische gegevens (locatie gekoppelde gegevens) toe. Het project richt zich dan ook op het ontwikkelen van “mobiele” leeromgevingen ter ondersteuning van een tweetal vakken waarin geo-informatiekundige aspecten een rol spelen: een basiscursus geo-informatie en een cursus integratie omgevingswetenschappen.

De basiscursus richt zich vooral op het individuele leertraject van studenten in de Bachelor-fase van de opleiding, waarbij de kansen voor plaats- en tijdonafhankelijk leren worden verkend. De integratiecursus is bedoeld om het projectonderwijs in de Master-fase verder te brengen via een koppeling van veldwerk aan desktop-werk.



Figuur 1. Projectdoelen

In de ontwikkeling van beide cursussen is daarbij gezocht naar een digitale leeromgeving waarin de student letterlijk meer "mobiel" is en niet perse studeert via de aan de kabel gebonden digitale werkplek. De term wireless learning (W-learning) staat in dit project dan ook centraal.

Figuur 1 laat zien op welke wijze hoe de twee cursussen zijn geoperationaliseerd. Voor zowel het individuele leertraject (de basis cursus) als het project- of groepstraject (de integratie cursus) zijn allereerst didactische en inhoudelijke modellen ontwikkeld (niveau I). Vervolgens zijn die modellen vertaald naar de implementatie als een Internet-leeromgeving (niveau II). Tenslotte zijn delen van de cursussen omgezet naar een 'wireless' implementatie (niveau III).

Om de confrontatie van zowel studenten als docenten met de 'nieuwe' technologie-laag zo toegankelijk mogelijk te maken is tevens gewerkt aan een helpdesk die als het ware assisteert bij het voorbereiden als uitvoeren van de cursus.

In dit rapport wordt verslag gedaan van een vooronderzoek mbt. de meer technische geaardheid van het locatie gebaseerd leren en computer ondersteund veldwerk (niveau III).

In eerste instantie is in het GIPSY project gezocht naar aansluiting bij de mobiele telefonie markt via oa. WAP en I-mode. In dit verslag zijn de verkennende studies naar het gebruik van deze technologieën gedaan. Uiteindelijk is op basis van dit vooronderzoek gekozen voor aanpassing van commerciële software (ArcPad) in combinatie met Internet Map Server. In GIPSY rapport 2004-3 is dit gebruik voor uitgewerkt.

Versie: 2.0 mei 2002	1
Voorwoord	3
1 Inleiding	6
1.1 Achtergrond	6
1.2 Doelstellingen	6
1.3 Onderzoeksopzet	7
1.4 Opbouw van dit rapport	7
2. Mobiele data communicatie	8
2.1 Inleiding	8
2.2 Hardware en Software	8
2.3 Mobiele telefoons	8
2.4 Personal Digital Assistants	9
2.5 Verbindingprotocollen	10
2.6 Data en presentatieprotocollen	11
2.7 Eisen voor een mobiele GIS oplossing	13
3. Geo-Data Viewers voor mobiele devices	15
3.1 De Geo-Wap applicatie	15
3.2 Geo I-mode applicatie	19
3.3 Client-Server applicatie	24
Bijlagen	26
Appendix 1: Chtml notitie van het W3 consortium (niet compleet; complete versie via internet)	26
Appendix 2 : Geo-Imode pilot applicatie (incomplete code	33
Appendix 3: Veldwerk met GPS gekoppeld aan ArcPad.	43

1 Inleiding

1.1 Achtergrond

Een stabiele infrastructuur voor het inwinnend, analyseren en verspreiden van geo-informatie is ook binnen het GIPSY-project van belang. Hiervoor wordt gebruik gemaakt van moderne GIS en Remote Sensing software en opslagsystemen.

Een overgang naar internet technieken vindt momenteel plaats. Op deze manier kan onafhankelijk van plaats en tijd geo-informatie ter beschikking worden gesteld aan de onderzoekers en studenten van Wageningen-UR, KU-Nijmegen en VU- Amsterdam en derden.

Vanuit het werkveld van Wageningen-UR wordt in toenemende mate de vraag gehoord of het mogelijk is deze informatie ook digitaal in het veld aan te bieden. dit betekend een draadloze verbinding met een database via een mobile telefoon, een zogenaamde personal digital assistant (PDA) of een laptop. Dit heeft een aantal voordelen:

- In het veld kan ad hoc geografische informatie worden opgevraagd.
- Het is niet meer nodig data te laden op een laptop of veldcomputer.
- Gegevens opgeslagen in de database zijn direct te vergelijken met een situatie zoals die aangetroffen in het veld.
- Gegevens zijn eventueel direct aan te passen in de database.
- Geen dure en kwetsbare apparatuur (laptop) hoeft mee het veld in.
- Data is eenvoudiger te beschikking te stellen aan derden.

In het kader van het GIPSY-project en het Strategisch Expertise Onderzoek (SEO), project “de Virtuele Groene Ruimte”, is een onderzoek uitgevoerd naar het draadloos benaderend van geo-informatie met behulp van mobiele telefoons, een laptop en een zogenaamde “personal digital assistant” (PDA).

Dit vooronderzoek is uitgevoerd door een projectteam van het Centrum voor Geo-Informatie, het projectteam van GIPSY en in samenwerking met Wageningen Software Labs (W!SL)

Het vernieuwende van dit project is niet zozeer het met een mobiel apparaat contact maken met een database en daar gegevens uit op kunnen vragen maar met name het locatie afhankelijk kunnen opvragen van deze gegevens. Dit laatste liefst zo transparant mogelijk naar de gebruiker toe.

1.2 Doelstellingen

Het belangrijkste doel van deze verkenning is te komen tot kennis over hoe een technische infrastructuur kan worden ontwikkeld waarmee op een robuuste en beheerbare manier geografische informatie kan worden gepresenteerd, gedigitaliseerd en ge-edit met behulp van een PDA of een moderne mobiel telefoon.

Dit doel brengt de volgende vragen met zich mee:

- Welke software voldoet als platform voor het uitleveren van geo-informatie voor mobiele apparaten (de zogenaamde map-servers en web-servers).
- Welke presentatie, overdrachtsprotocollen en clients zijn geschikt voor het presenteren en visualiseren van geo-informatie op een mobiel apparaat.
- Welke apparaten zijn geschikt voor gebruik als terminal.
- Op welke manier kan locatie informatie worden doorgegeven via het pda of telefoon aan de map-server.
- Welke GPS apparaten zijn bruikbaar hiervoor en welke alternatieve methoden zijn eventueel beschikbaar voor het bepalen van de geografische locatie.

1.3 Onderzoeksopzet

Om de gewenste doelstellingen te kunnen realiseren zijn de volgende acties uitgevoerd:

- Een programma van eisen is opgesteld waaraan een mobiele gis implementatie aan moet voldoen.
- Eenvoudig software is in huis ontwikkeld waarmee een eenvoudig kaartje kan worden gegenereerd en verzonden naar een mobiel telefoon. Drie pilots zijn uitgewerkt
 - Een Geo-Wap applicatie voor WAP telefoons.
 - Een I-mode/PDA applicatie.
 - Een client-server applicatie geschikt voor PDA's en Laptops.
- Op basis van een aantal tests waarin diverse databronnen zijn opgenomen is aan de hand van het programma van eisen:
 - Een aantal mobiele apparaten vergeleken op de aspecten: koppeling/integratie mogelijkheden met rand apparatuur (GPRS en GPS devices), bedieningsgemak en prestaties.
 - Client software beoordeeld op aspecten als visualisatie, query, editing en communicatie met andere software
- Aan de hand van de resultaten van de stappen 2 en 3 is een voorstel gedaan voor een architectuur waarmee geo-informatie via verschillende apparaten en verbindingstypen kan worden uitgeleverd.

1.4 Opbouw van dit rapport

Het rapport is als volgt opgebouwd:

Hoofdstuk twee beschrijft het generiek concept van mobiel datatransport, de beschikbare technieken protocollen en apparaten. Hierin wordt ook het testproject en het programma van eisen voor een "kaart zonder draad" implementatie beschreven.

Hoofdstuk drie beschrijft een aantal oplossingen voor draadloze geo-informatie. De eerste oplossing is een WAP applicatie ontwikkeld in samenwerking met WISL. Hiermee kunnen eenvoudige kaartje op een standaard wap telefoon kunnen worden opgevraagd. De tweede applicatie is een op ESRI/ArcIMS gebaseerde oplossing waarmee kaartjes kunnen worden benaderd op I-mode en PDA's. De derde applicatie bestaat uit een server-client oplossing die is gebaseerd op bestaande software van ESRI.

2. Mobiele data communicatie

2.1 Inleiding

Onder mobiel datacommunicatie wordt verstaan het draadloos verzenden en ontvangen van data via een mobiel apparaat (telefoon, pda of laptop). Dit kunnen bijvoorbeeld e-mail, webpagina's, afbeeldingen of muziek en video zijn. In dit project betreft het geo-informatie.

Van mobiel datacommunicatie wordt verwacht dat het een grote vlucht neemt de komende jaren. Telecom organisaties zijn bereid hoge bedragen te betalen om voor frequenties waarmee in de toekomst met een grote bandbreedte data en informatie kan worden verzonden naar allerlei mobiele apparaten

Dit hoofdstuk beschrijft bestaande mogelijkheden voor wat betreft apparaten, verbindingsprotocollen en presentatieprotocollen.

2.2 Hardware en Software

Om mobiel data te kunnen ontvangen of verzenden is aan de kant van de gebruiker een van de volgende apparaten nodig:

- Een mobiel telefoon geschikt voor datatransport (wap of I-mode) of
- Een personal digital assistant (PDA) uitgerust met een draadloos modem of geïntegreerde module of een bluetooth verbinding met een mobiele telefoon.
- Een laptop voorzien van een pc-insteek kaart geschikt voor draadloze communicatie of een irda of bluetooth verbinding met een mobiele telefoon. (voor uitleg bluetooth en irda zie paragraaf 2.3).

2.3 Mobiele telefoons

Een telefoon kent slechts beperkte mogelijkheden voor het ontvangen en presenteren van data.

Op dit moment (begin 2002) zijn er twee typen telefoons op de Nederlandse markt (zie figuur 1):

1. de Wap-telefoon
2. de I-mode telefoon



Figuur 1: Links een Wap-telefoon, rechts een I-Mode telefoon

Wap-telefoons (figuur 1) zijn uitgerust met een eenvoudige "browser". Hiermee kan eenvoudig gestructureerde tekstuele informatie en eenvoudige grafische afbeeldingen in het WBMP formaat (geen kleur en een lage resolutie) worden weergegeven.

Op voorhand kunnen een aantal restricties met betrekking tot het gebruik van een Wap-telefoon voor het opvragen van geografische data worden genoemd:

- De displaygrootte is over het algemeen beperkt voor het afbeelden van een cartografische representatie. Oriëntatie op de kaart zal over het algemeen worden bemoeilijkt doordat er een onvoldoende groot deel van de kaart kan worden afgebeeld.
- De grafische mogelijkheden van de WAP telefoon zijn the beperkt om een cartografische symbolen voldoende te kunnen afbeelden. Het grafische format (WBMP) kent slecht een binair pallet. De af te beelden kaarten kunnen daarom slecht zeer eenvoudig van opmaak zijn.
- Opvragen en bevragen van geografische data bestanden wordt bemoeilijkt door de gebrekkige mogelijkheden voor data invoer. Voor invoer is in alleen een numeriek toetsenbord beschikbaar.
- De meeste telefoons zijn niet te koppelen met randapparaten zoal een Global Positioning System (GPS). Locatiebepaling is op deze manier moeilijk. GPS informatie zal meestal via het numeriek toetsenbord moeten worden ingevoerd in de mobiele telefoon. Een directe locatiebepaling via het mobiele netwerk van een telecom organisatie is in principe mogelijk maar wordt op dit moment in Nederland niet breed ondersteunt.

In 2002 is de I-mode telefoon geïntroduceerd. Deze telefoons kennen over het algemeen een groter kleuren display en multimedia mogelijkheden. Daarnaast kan op de moderne telefoons eenvoudige software worden geïnstalleerd.

De I-mode telefoon biedt een aanzienlijk hogere scherm resolutie (120 * 128) en 256 kleuren in vergelijking met een WAP-telefoon. Als enige grafisch formaat wordt op dit moment gif ondersteunt (eventueel animated met een maximaal aantal frames van 5, transparency en interlaced).

De maximale grote van een pagina voor een I-mode telefoon (afbeelding en tekst) is 10 kB. Het wordt aanbevolen om de pagina niet groter te maken dan 5 kB. Het valt te verwachten dat, met name in geval van cartografische representaties met veel verloopkleuren, deze restrictie problemen oplevert.

Wat betreft invoer gelden de zelfde beperking als voor WAP-telefoons. Ook koppeling met gps is bij de meeste huidig terminals niet mogelijk.

2.4 Personal Digital Assistants

Een PDA biedt meer mogelijkheden om geografische informatie cartografisch te visualiseren. De meest PDA's bieden een schermresolutie van ongeveer 240 bij 320 pixels in kleur. Dit is aanzienlijk meer de een telefoon.

Daarnaast biedt een PDA een meer mogelijkheden om data in te voeren. Typische zijn een touchscreen, in combinatie met handschrift herkenning of een vitueel toetsenbordje. Eenvoudige tekstinvoer in formulieren is mogelijk maar uitgebreide invoer blijft tijdrovend.

Er zijn een aantal aangepaste GIS pakketjes verkrijgbaar voor een PDA. Binnen het CGI is op dit moment ArcPad in gebruik. ArcPad heeft de mogelijkheid om data op te vragen via mapserver.

De meeste PDA's zijn uitgerust met of het Palm OS besturingssystem of de micro uitvoering van Windows genaamd Pocket PC. Voor gebruik in een draadloze GIS oplossing lijkt Pocket PC uitvoeringen de meest mogelijkheden te bieden vanwege compatibiliteit met courante ontwikkelomgevingen als Visual Basic en de compatibiliteit met het Windows besturingssystem.



Figure 2: links een Pocket PC PDA, rechts een Palm OS PDA

De huidige trend laat zien dat PDA's en mobiele telefoons in toenemende mate geïntegreerd worden. Vanuit de telefoonkant zien we de zogenaamde smartphones (bijvoorbeeld de Nokia 9210) en vanuit de PDA kant handhelds met ingebouwde gsm en gprs mogelijkheden (bijvoorbeeld de XDA van O2).

2.5 Verbindingprotocollen

Naast een mobiel apparaatje stelt de transmissie van data andere eisen aan de gebruikte protocollen dan voice. Het GSM netwerk in Nederland is met name ingericht voor spraak. De laatste jaren zijn verschillende ontwikkelingen door gevoerd voor het verzenden van data:

Op dit moment zijn de manieren mogelijk voor het verzenden van data

- GSM
- GPRS
- UMTS
- WIFI
- Bluetooth
- IRDA

a. GSM

Het GSM protocol is in principe primair ontworpen voor spraak. Daarnaast kan het worden ingezet voor de overdracht van data.

GSM maakt gebruik van een zogenaamde circuitschakeling. Dit betekent dat er een verbinding exclusief voor de gebruiker wordt opgezet met een bepaalde capaciteit. De maximale bandbreedte die gerealiseerd kan worden via het GSM protocol is 9,6 kb/s.

De kosten voor deze wijze van datatransmissie zijn afhankelijk van de verbindingduur. De tarieven zijn gelijk aan die voor spraak met een mobiele telefoon. Voor het maken van een verbinding moet worden ingebeld op een inbelpunt. Dit duurt typisch iets van 30 a 40 seconden.

b. GPRS

GPRS staat voor Global Packet Radio Service. GPRS is een zogenaamde pakketgeschakeld protocol specifiek ontworpen voor datatransmissie. De data die via een draadloos netwerk moet worden verzonden wordt aan stukje "gehakt". Elk pakketje wordt afzonderlijk voorzien van een adres en vervolgens het netwerk opgestuurd. Aan de ontvangerkant worden deze pakketjes vervolgens opgevangen en weer aan elkaar gezet. Een verbinding met een IP netwerk, wat volgens het zelfde principe werkt, is eenvoudig.

Een voordeel van GPRS is dat de gebruiker altijd een vaste verbinding heeft (always on) er hoeft niet ingebeld te worden zodra er data te verzenden of te ontvangen valt gebeurt dit vanzelf.

Theoretisch is er een snelheid mogelijk van 115 KB/sec. In de praktijk zijn maximale snelheden aangeboden door de meeste netwerkoperators in Nederland: 56 Kb/s downstream en 14 Kb/s

upstream. Of deze snelheden gerealiseerd kunnen worden hangt af van de hoeveelheid tijdsloten die door het mobiele apparaat kunnen worden gebundeld. De hoeveelheid data per tijdslot hangt af van de gebruikte foutcorrectie. Er zijn vier klassen te onderscheiden

Een goede vuistregel is op dit moment een downstreamsnelheid van rond de 25 Kb/s en een downstream.

c.UMTS

Het Universal Mobile Telecommunications System (UMTS) protocol is in Nederland nog niet beschikbaar. Verwachting is dat over ongeveer 1 a 2 jaar de eerste implementaties van UMTS gereed zijn. UMTS is een zogenaamd derde generatie mobiele communicatie systeem waarbij theoretisch snelheden mogelijk zijn tot 2 Mbit/s, snelheden vergelijkbaar met een vaste aansluiting.

d. WIFI pm

e. Bluetooth

Bluetooth is een verbindingprotocol specifiek ontwikkeld voor dataverbinding over korte afstanden (max 10 m). Het wordt met name toegepast als alternatief voor het verbinden van apparaten via een kabeltje of infrarood.

Het principe van Bluetooth is gebaseerd op communicatie tussen een master en maximaal 7 slaves. Dit wordt ook wel aangeduid met de term piconet. Daarnaast kan apparatuur uit de verschillende piconets met elkaar communiceren waardoor een zogenaamd scatternet ontstaat. De datasnelheid is maximaal 1 Mb/s waarbij een deel gebruikt wordt voor foutcorrectie. De effectieve bandbreedte ligt daarom wat lager.

De naam Bluetooth komt uit een sage waarin een Scandinavische koning de verschillende vikingstammen wist te verenigen tot één volk. Een hierbij opgelopen klap bezorgde hem een blauwe tand en de toepasselijke bijnaam 'Blaetand'.

De toepassing binnen het een mobiel-gis implementatie ligt vooral in het verbinden van een terminal (een modem in de vorm van een mobiele telefoon) met een laptop of PDA. De telefoon kan gewoon in de zak of tas blijven.

f. IRDA

2.6 Data en presentatieprotocollen

Naast een verbindingprotocol zijn er een aantal protocollen geschikt om data efficiënt via een verbindingprotocol te versturen. Met efficiënt wordt bedoelt:

Een zo gering mogelijke omvang

Zo geschikt mogelijk voor de beperkte display en verwerking capaciteit van mobiele apparaten.

De volgende protocollen zijn in principe beschikbaar:

- Wireless Application Protocol (WAP)
- cHTML
- XML
- SVG

WAP

De kern van WAP wordt gevormd door WML: Wireless Markup Language. Er wordt dus geen gebruik gemaakt van het op het Web gehanteerde HTML: Hypertext Markup Language, maar er is een aparte op XML gebaseerde taal ontwikkeld. Naast WML omvat de WAP standaard protocollen gedefinieerd in een viertal lagen (layers): Wireless Application Environment (WAE), Wireless Session Layer (WSL), Wireless Transport Layer (WTL) en Wireless Transport Layer Security (WTLS).

Het WAP protocol is ontworpen voor mobiele telefoons met kleine schermjes. In principe is WAP textgeorieerd. Beperkte grafische afbeeldingen zijn te realiseren via afbeeldingen in het WBMP format. Deze zijn echter van een lage resolutie en monochroom.

cHTML

cHTML is een subset van HTML specifiek ontwikkeld voor kleine apparaten. Op dit moment wordt cHTML het meest toegepast op de zogenaamde i-mode telefoons (zie figuur 1). Het belangrijkste verschil met HTML is dat de invoermogelijkheden sterk beperkt zijn. Het protocol gaat uit van een eendimensionale cursorbewegingen (vooruit en achteruit) in tegenstelling tot HTML waar tweedimensionale cursorbewegingen worden ondersteunt. Tables en imagemaps worden daarom niet ondersteunt in cHTML. Daarnaast zijn worden ook de volgende mogelijkheden van HTML niet ondersteunt

- Multiple character fonts and styles
- Background color and image
- Frames
- Tables
- Style sheets

Wat betreft afbeeldingen worden alleen het gif formaat ondersteunt.

Het grote voordeel van cHTML is dat het maken van pagina's voor de i-mode telefoon eenvoudig is aangezien het een subset is van HTML. cHTML kan op een gewone browser bekeken worden.

Aangezien navigatie binnen een cHTML pagina slechts eendimensionaal is (alleen scrollen) is het niet mogelijk "image maps" of ander tweedimensionale navigatie structuren te gebruiken.

XML

De vorige twee protocollen zijn primair gericht op het presenteren van informatie. XML is een protocol specifiek bedoelt het structureren van data ten behoeve van uitwisseling van data tussen verschillende applicaties. De reden dat XML hier genoemd wordt is dat veel map servers XML gebruiken. Deze servers kunnen aangestuurd worden via XML en leveren output in de form van XML. Voor het structureren van geografische data wordt door het OpenGis consortium de zogenaamde GML standaard ontwikkeld. Veel mapservers zoals bijvoorbeeld Arclms van Esri gebruiken zelf ontwikkelde XML varianten die niet voldoen aan de OpenGis standaarden.

XML is dus niet bedoelt voor het presenteren van data maar uitsluitend voor het uitwisselen van data (zie kader 1). Via zogenaamde style sheets (CSS en XSL) of kan een XML file naar een presentatie protocol zoals HTML, cHTML of eventueel WML worden omgezet. Het grote voordeel

van deze aanpak indien een nieuw presentatie protocol ondersteunt moet worden hiervoor geen nieuwe applicatie voor hoeft worden te ontwikkeld maar slechts een aantal stylesheets. Software zoals de "wireless application server" van Oracle kan dan relatief eenvoudig de gegevens omzetten naar een representatie geschikt voor de applicatie die deze informatie opvraagt. Deze software "herkent" het apparaat dat contact zoekt met initieert aan de hand daarvan de nodig stylesheets ed.

```
<?xml version="1.0"?>
  <purchase id="p001">
    <customer db="cust123">
      <product db="prod345">
        <amount>23.35</amount>
      </product>
    </purchase>
```

Kader 1: voorbeeld van een eenvoudige XML structuur

2.7 Eisen voor een mobiele GIS oplossing

In bovenstaande zijn de technische mogelijkheden beschreven zoals die op dit moment beschikbaar zijn of binnen afzienbare tijd beschikbaar worden. De volgende paragrafen gaan dieper in op de eisen zoals die vanuit het domein van het CGI aan een mobiele GIS oplossing worden gesteld.

- De eisen zijn onderverdeelt in :
- Functionele eisen vanuit de client
- Functionele eisen vanuit de server
- Functionele eisen vanuit de data

Client

Vanuit de client applicatie (dus de applicatie die draait op de laptop, pda of mobiele telefoon) zijn de volgende eisen gesteld aan de testsituatie

- Geografische informatie visualiseren
 - Inzoomen
 - Uitzoomen
 - Pannen
 - Legenda
- Geografische informatie bevragen
 - Bevraging adhv attribuutwaarden. Selectie laten zien in de visualisatie
 - Bevraging gebaseerd op geometrische afstand rondom een object
 - Bevraging adhv handmatig selectie van een object
- Geometrische informatie Aanpassen
 - Toevoegen van een lijn, punt of vlak
 - Verwijderen van een lijn,punt of vlak
- Attribuut informatie Aanpassen
 - Attribuut informatie aan een punt, lijn of vlak kunnen toevoegen of aanpassen
 - Vrije text opmerking aan een attribuut kunnen toevoegen

Server

De eisen die aan de mapserver zijn gesteld hebben betrekking op de volgende aspecten:

- koppeling met diverse databronnen
 - koppeling SDE

- koppeling Oracle Spatial
- Installeerbaarheid en beheerbaarheid
- Data formats
 - vector
 - grid
- Query mogelijkheden
- Edit en Update mogelijkheden
- Performance en scalability

Verschillende mapservers (GIPSY rapport 2004-6) worden met elkaar vergeleken volgens de QUINT methodiek.

Data

De eisen die aan een wireless applicatie worden gesteld voor wat betreft de geo-data zijn eenvoudig. Een applicatie moet in staat zijn geo-data zoals die op dit moment gebruikt worden binnen het CGI te verwerken. Dit betekent dat tenminste ESRI grid en vector datasets gehanteerd moeten kunnen worden en geo-data opgeslagen binnen Oracle Spatial.

3. Geo-Data Viewers voor mobiele devices

3.1 De Geo-Wap applicatie

In samenwerking met het Wageningen Software Labs (WISL) en het Software Expertise en Research Centre (SERC) is er een software pilot uitgevoerd. Het doel van deze pilot was met name ervaring opdoen in de technische aspecten van het ontwikkelen van een Geo-Wap applicatie.

Onderstaande documentatie is deels gebaseerd op de rapportage van WISL (Steven Hoek).

Aanpak

De volgende technieken zijn uitgediept:

- XSLT
- Aanroepen van componenten vanuit ASP
- Genereren van kaartje voor distributie over het internet.

De volgende functionaliteiten zijn gerealiseerd:

- Demonstratie van XSLT waarmee content afhankelijk van de cliënt kan worden gepresenteerd in HTML of WML
- Inlog functionaliteit, waarbij gebruikers worden gecontroleerd op basis van gegevens in een database.
- Geografisch zoeken op basis van een x,y coördinaat in het RD-stelsel. Geografische objecten binnen een door de gebruiker op te geven zoekstraal presenteren
- Presentatie van de resultaten middels een kaartje en alfanumerieke beschrijvingen.

Voor de pilot is een landsdekkende bestand gebruikt met recreatieve verblijfsrecreatie objecten.

Voor datatransmissie is het GSM protocol gebruikt (zie 2.3.1) als presentatie protocol is gebruik gemaakt van WAP (zie 2.4.1). De reden voor de keuze ligt met name in de technische mogelijkheden ten tijde van de implementatie (begin 2001). Er is echter geen restrictie om een GPRS gebaseerde WAP verbinding op te zetten.

XSLT

Om de presentatie van informatie los te koppelen van de inhoud (data) van de informatie is gekozen om de data te beschrijven volgens het XML protocol (zie 2.4.3). Door deze oplossing te kiezen kan op een flexibele manier de data worden gepresenteerd op manier; in dit geval volgens HTML en WML

Voor de transformatie is gekozen voor de techniek XSLT (XSL transformations). Volgens deze techniek wordt de data gestileerd:

- zodat de data beantwoord aan de semantische definitie van de cliënt
- en deze geformatteerd wordt naar een voor mensen leesbare lay-out.

Voor een op XSL gebaseerde transformatie is een XSL document (eXtensible Stylesheet Language) nodig die het gewenste format beschrijft. Het XSL document is daarmee afhankelijk van de informatiebehoefte van de gebruiker. Niet alle attributen die zijn opgeslagen in het XML document hoeven te worden opgenomen in het XSL document. De vocabulaire voor een XSL document moet worden herkend door een XSLT processor (of rendering agent) .

Als XSL-processor is de PERL module XML::XSLT gebruikt (versie 0.32) ontwikkeld door Geert Josten (gjosten@sci.kun.nl) en Egon Willighagen (egonw@sci.kun.nl). Deze module is afhankelijk van de Perl modules XML:Parser, XML::DOM en LWP::Simple. De module XML::Parser maakt gebruik van de XML parser van James Clark (<http://www.jclark.com/sml/>)

Een eenvoudiger oplossing is mogelijk via de Microsoft XML Parser. Deze is direct aan te roepen vanuit bijvoorbeeld ASP via VbScript. Deze oplossing heeft als belangrijk nadeel de

afhankelijkheid van xmldom.dll en de Microsoft Internet Explorer. De platform onafhankelijkheid is hiermee niet gebaat.

Om de structuur van de XML file vast te leggen voor deze applicatie is een DTD gemaakt. Vanwege tijdgebrek zijn niet alle transformatieregels voor de elementen uit de DTD opgesteld.

Veel kennis over bovenstaande is te vinden in: (<http://www.w3schools.com>)

Aanroepen van componenten vanuit ASP

Om de WML pagina's van een dynamisch content te voorzien is gekozen voor de Active Server Pages (ASP) oplossing van Microsoft. Alhoewel deze specifiek voor de IIS webserver van Microsoft is ontwikkeld zijn er oplossingen voor ander platforms ontwikkeld. Voor UNIX is er een implementatie voor ASP ontwikkeld door ChiliSoft. Een betaalbare oplossing is Instant ASP van HalcyonSorf.

Voor de Apache mapserver is er een opensource oplossing ontwikkeld door Microsoft partner ActiveState (www.activestate.com) genaamd Apache::ASP. Deze oplossing maakt het mogelijk om ASP met PerlScript te combineren op alle platforms waarop Apache kan draaien. De toepassing van de ASP-cum-Perlscript methode is binnen dit project is beperkt gebleven tot experimenten. Gezien de tijd en enkele lastige problemen (niet gedocumenteerd, voor meer informatie Steven Hoek) is er uiteindelijk gekozen voor een IIS specifieke oplossing in combinatie met VbScript.

Aangezien ASP geïnterpreteerd worden kan bij hogere belasting een ASP web applicatie traag worden. Door rekenintensieve acties via een component te laten uitvoeren kan er aan performance gewonnen worden. Op het wintell platform zijn dit over het algemeen ActiveX dll's . Deze worden vanuit ASP aangeroepen via het Server.CreateObject statement.

Een bijkomend voordeel van het gebruik van componenten is dat deze in een afzonderlijk deel van het geheugen kunnen draaien. Het gebruik van componenten kan daarmee de webserver stabiel maken aangezien een crash van een component geen gevolgen heeft voor de webserver zelf.

Voor deze pilot zijn twee componenten in Delphi ontwikkeld:

- een inlogcomponent
- een component voor het opvragen van een URL vanuit een ASP script.

Functioneel ontwerp

Om kaartje te creëren uit een geodatabase is gebruik gemaakt van een component van ESRI (www.esri.com), MapObjects (versie 2.1). Het is vaak moeilijk een dusdanig toepassingen te bouwen in een "out-of-process" component omdat de webserver de component (bijvoorbeeld een CGI-executable) vaak niet de tijd geeft om alles af te ronden.

Om dit probleem te voorkomen is gebruik gemaakt van de IdHTTPServer component uit de Indy suite (een alternatief is de WebLink component van ESRI). De relevante MapObjects componenten zijn opgenomen in een zelfstandig draaiende executable die op TCP/IP poort 5062 luistert naar inkomende verzoeken. Deze executable (bekend onder de werknaam W!SL map server) genereert een bitmap met het kaartje. Deze bitmap wordt verzonden naar een Delphi bitmap component die het omzet in het benodigde formaat voor visualisatie op een wap-terminal (wbmp formaat).

De geïmplementeerde oplossing werkt in grote lijnen als volgt:

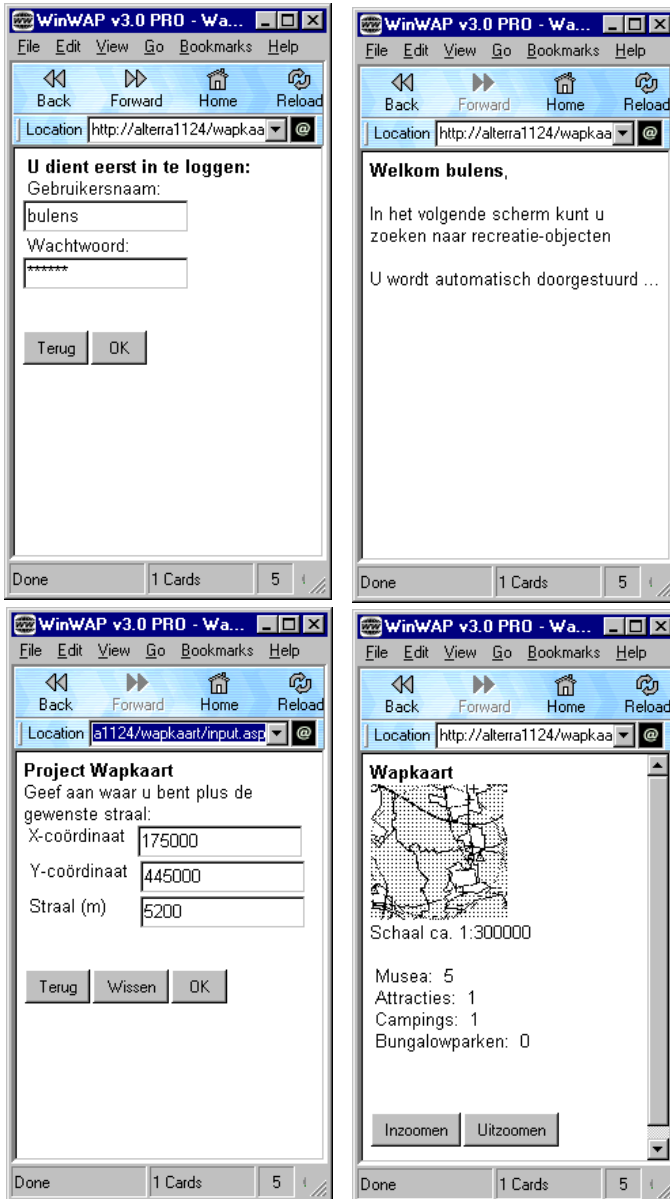
- a. Er komt via TCP/IP poort 5062 een verzoek binnen in de vorm van een URL met een query string. Deze query string bevat de x,y coördinaten en een gewenste zoekstraal.
- b. Met behulp van de "W!SL map server" wordt het gewenste kaartje gemaakt
- c. Het kaartje wordt op het clipboard gezet
- d. Van het clipboard wordt het geladen in de Delphi bitmap component
- e. De bitmap wordt omgezet naar wbmp
- f. het wbmp plaatje wordt streaming teruggestuurd naar de aanroepende socket.

De gekozen oplossing via het clipboard levert problemen op . De applicatie is niet multithreading. Om hiermee problemen te voorkomen is een CGI-executable gemaakt die actief blijft en de verzoeken voor plaatjes sequentieel doorstuurt naar de "W!SL mapserver". Intussen zijn ook andere oplossingen bekend om transfer via het clipboard te voorkomen.

Om naast de puur grafische inhoud ook attribuutgegevens (informatie over de ruimtelijke objecten in het kaartje) in de WML (of HTML) pagina te laten zien was het noodzakelijk om XML te produceren. Hiervoor is de ASP component ontwikkeld voor het opvragen van een URL vanuit een ASP-script. De ontvangen tekst kan zonodig nog getransformeerd worden.

Resultaat

Figuur 3 geeft een aantal schermafbeelding van de ontwikkelde applicatie. De Geo-wap cliënt bestaan uit een viertal schermen



figuur 3: Schermafbeeldingen van de geo-wap applicatie

- Inloggen scherm: hierin voert de gebruiker zijn inlognaam en wachtwoord in waarna hij of zij toegang krijgt tot de applicatie (functionaliteit 2 zie par. 3.1.1)
- Welkom scherm: hierin krijgt de gebruiker instructies en wordt er daadwerkelijk contact gelegd met de mapserver
- Het scherm waar de gebruiker de gegevens van de locatie en de gewenste zoekstraal in kan voeren (functionaliteit 3 zie par. 3.1.1)
- Het resultaat scherm met het wbmp kaartje en een alfanumeriek overzicht van de gevonden objecten (functionaliteit 4 zie par. 3.1.1).

Discussie en Conclusies

Een van de belangrijkste restricties in het gebruik van het wap protocol voor het uitleveren van geo-informatie ligt in de haalbare resolutie de beperkte schermgrootte en de lage toegestane pagina grootte (enkel kb's) van de wap-telefoons. Hierdoor zijn alleen zeer eenvoudige kaartbeelden te visualiseren. Symboliek op de kaarten kan alleen weergegeven worden via diverse arceringen.

Attribuut gegevens zijn daarentegen wel goed mogelijk.

De gekozen implementatie (mapobjects in combinatie met Delphi, ASP, IIS en clipboard-transfer) op het Windows NT platform levert de nodige portabiliteits problemen op. Een installatie op een IIS onder windows 2000 bleek niet mogelijk. Een oplossing van dit probleem is niet eenvoudig (voor meer informatie Steven Hoek)

Voor het bepalen van de locatie is op dit moment de enige reële optie de koppeling met een GPS. Aangezien mobiele telefoons een koppeling met GPS apparatuur niet ondersteunen komt de locatiebepaling erop neer dat coördinaat informatie handmatig in de geo-wap applicatie moet worden ingetoetst.

De verschillende interpretatie van WAP pagina's door verschillende merken mobiele telefoons is een ander probleem waarmee terdege rekening mee moet worden gehouden. Met name als, zoals in deze applicatie, meer dan alleen tekst wordt aangeboden is het erg moeilijk om een goede visualisatie te krijgen op verschillende apparaten. Een tweetal oplossingen is mogelijk: een ontwerp geschikt voor slechts een of enkele type wap-telefoons de opzet van een database met daarin de gegevens en parameters voor de verschillende telefoons die op dit moment in omloop zijn.

Als we de eisen zoals gesteld in paragraaf 2.5 in ogenschouw nemen is de conclusie dat een op het wap-protocol gebaseerde applicatie in combinatie met een telefoon niet of nauwelijks bruikbaar is voor serieuze toepassing in het werkveld van geo-informatie. De mogelijke overdracht van informatie en de interactiemogelijkheden met gebruiker en randapparatuur zijn te beperkt.

3.2 Geo I-mode applicatie

Een tweede pilot is uitgevoerd in samenwerking met het GYPSY project. Doel van dit project was een I-mode (zie paragraaf 2.2.1 en 2.2.4) site te ontwerpen waarop geo-informatie kan worden opgevraagd. I-mode wordt gepresenteerd als de opvolger van het WAP protocol en biedt een duidelijk rijkere omgeving voor het presenteren en visualiseren van informatie op mobiele devices.

Aanpak

De volgende aanpak onderscheidt zich van de Geo-Wap-applicatie in het gebruik van bestaande mapserversoftware. De gebruikte Arc/Im software 3.0 biedt dusdanige functionaliteit en ontwikkelmogelijkheden dat zelf implementeren van server software niet meer loont.

De volgende functionele eisen hebben wel gesteld aan deze pilot :

- Connectie tussen i-mode toestel en ArcIm
- Keuze mogelijkheid om te verbinden met verschillende geo-databases
- Verschillende databronnen (layers) binnen een geo-database kunnen selecteren en visualiseren
- Geografisch zoeken op basis van een x,y coördinaat in het RD-stelsel .
- Geografische zoeken op basis van postcode
- Zoeken op een attribuut en resultaten presenteren middels kaartjes en attribuut informatie

- Geografische object binnen een door de gebruiker op te geven zoekstraal presenteren zowel middels een kaartje als middels alfanumerieke informatie
- Pannen en zoomen op een kaartje

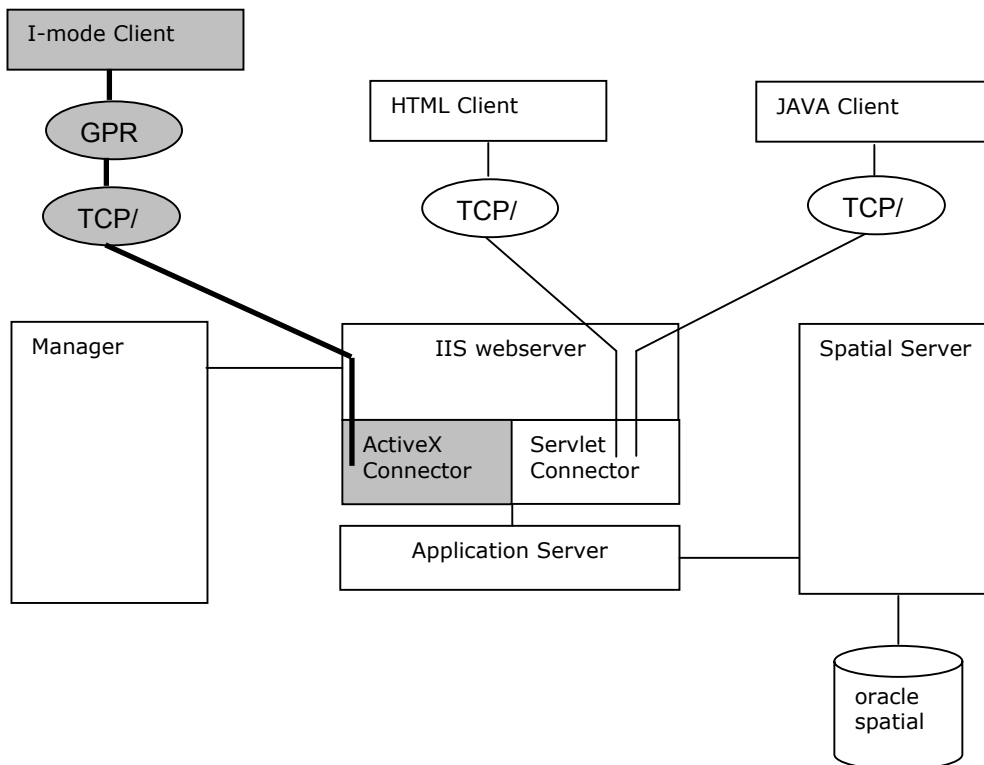
Daarnaast is onderzocht in hoeverre geo-databases kunnen worden ge-edit of nieuwe geo-objecten kunnen worden toegevoegd via de I-mode aanpak.

ArcIms

Als basis voor de server is gebruik gemaakt van de ArcIms software van ESRI. De reden van deze keuze is een pragmatische. Op moment van bouw was dit de enige map-server software waarmee voldoende ervaring is opgedaan om deze relatief snel te kunnen inzetten. Figuur 4 laat de architectuur van de server zien zoals deze is geïnstalleerd binnen het CGI. (zie voor meer informatie ESRI 2000) De grijs gearceerde delen zijn uitwerking zoals gerealiseerd binnen deze pilot. De dikke lijnen geven de datastromen aan zoals deze geïmplementeerd zijn. De default communicatie tussen webserver en applicatieserver gebeurt met de Servlet Connector in combinatie met de ArcXML taal ontwikkeld door ESRI. In deze opzet is gekozen de ActiveX connector te installeren waardoor het mogelijk wordt de functionaliteit van de spatial server direct via ASP in combinatie met VbScript aan te spreken. Gezien de ervaring met ASP en het gebruik van IIS is dit een logische keuze. Het nadeel is de verminderde platform onafhankelijkheid.

cHtml

Om informatie op een I-mode toestel te presenteren wordt gebruik gemaakt van cHTML (zie paragraaf 2.4.2). De definitie van het cHTML protocol is beschreven appendix 1. Een belangrijke beperking waarmee in deze applicatie rekening mee moet worden gehouden is de 10 Kb grens die aan een pagina wordt gesteld. Deze grens is niet zozeer een cHTML beperking als wel een beperking opgelegd door de meeste I-mode telefoons.



figuur 4: Architectuur van de ArcIms mapserver

cHTML is een subset van het reguliere HTML wat gebruikt wordt voor het ontwikkelen van websites voor het reguliere internet. cHTML is daarom ook leesbaar door normale webbrowsers op pc's en pda's.

ASP

De Active Server Pages is een technologie ontwikkelde door Microsoft als een antwoord op de soms complexe problemen die ontstaan bij het gebruik van CGI-toepassingen. In principe kan elke scripttaal in combinatie met ASP technologie gebruikt worden (VbScript, Python, Perl, JavaScript). In deze pilot is gekozen voor de server-side VbScript oplossing, verruit de meest gebruikte. ASP is ontwikkeld voor gebruikt met de IIS webserver van Microsoft. Zie voor een uitgebreidere discussie over ASP de paragrafen 3.1.3 en 3.1.4

ASP in deze applicatie kent een tweeledig doel:
het aanspreken van de ArcIMS ActiveX objecten
het genereren van een deel van de cHTML tags.

De ActiveX library van de ArcIms server bestaat uit een aantal objecten die direct toegang geven tot de functionaliteit van de mapserver . Het gebruik van deze object maakt het mogelijk om direct de functionaliteit van de map-server aan te spreken.

Functioneel ontwerp

Via de ActiveX connector van de Arclms mapserv kan direct de data worden benaderd. Het opvragen van een kaartje gaat via de volgende procedure

- a. Er wordt een arclmsConnector object geïnitieerd:
`Server.CreateObject("aims.ArcIMSConnector")`
- b. via dit server object wordt contact gelegd met de mapserver.
- c. Vervolgens wordt er op de server een Mapobject geïnitieert:
`Server.CreateObject("aims.Map")`
- d. Via de MapObject kan vervolgens een dataset worden opgevraagd en geconverteerd naar een bitmap met een bepaalde omvang en achtergrondkleur. De symbolen en kleuren van het kaartje zelf zijn a-priori bepaald via de authoring module van de mapserver.7
- e. De kaart kan op een pagina worden gezet door de url van de kaart op te vragen bij het MapObject:
`urlImage = mMap.GetImageAsUrl()`
 en vervolgens deze via ASP in een CHtml tag te zetten bv:
`Response.write ""`

Daarnaast zijn functionaliteiten gebouwd voor het bevragen, pannen, zoomen, opvragen van attributen, aan- en uitzetten van layers ed. Het gaat te ver om in het kader van dit rapport de exacte implementatie van deze functionaliteiten te beschrijven. Ter illustratie is in Appendix 2 een deel van de ASP pagina en routine opgenomen.

Een belangrijke restrictie in de ontwikkeling van dit soort applicaties is het niet accepteren van zgn cookies door een I-mode telefoon. Aangezien CHtml een zgn stateless protocol is, is het hierdoor is het moeilijk (niet onmogelijk) om een referentie met de geïnitieerde objecten op de server te onderhouden. In deze applicatie is dit opgelost door telkens bij het opvragen van een pagina object opnieuw te initialiseren. Het voordeel dat de applicatie op elke geschikt apparaat werkt zonder de noodzaak tot het opslaan van cookies. Het nadeel is een duidelijke extra belasting voor de server.

Resultaten

Figuur 5 geeft een aantal afbeelding van de Geo I-mode pagina's van de. Als voorbeeld is een locatie bepaling via de postcode genomen. De Geo-wap cliënt bestaan uit een aantal schermen. De eerste keuze die een gebruiker moet maken is die van de dataset die hij of zij wil benaderen. Deze datasets dienen uiteraard van te voren via de Arclms server beschikbaar worden gesteld.

Op dit moment kan op vier manieren een locatie worden gevonden:

- a. door interactief te pannen en te zoomen naar te locatie;
- b. door een 2 of 3 cijferige postcode in te voeren ;
- c. door een gemeentenaam in te voeren of;
- d. door een rijkdriehoek coördinaat in te voeren

 <p>Welcome to the Mobile GIS server of the Centre for Geo-information</p> <p>This site is specially designed to be viewed on PDA's and i-mode devices</p> <p>Click here to start the mobile mapserver</p>	<p>Currently we have the following geo-databases available for viewing on mobile devices.</p> <p><i>(please note that i-mode currently is not supported. Soon it will.)</i></p> <p>IGI</p> <p>Cultuurhistorie</p> <p>Landschappen</p>	<p>Choose of the following actions for database :</p> <ul style="list-style-type: none"> • Pan and Zoom on a map • Find location by 4 digit zipcode • Find location by 3 digit zipcode • Find location by 2 digit zipcode • Find location by municipality name • Find location by RD-coordinates
<p>please enter a 4-digit zipcode:</p> <p>zip:</p> <input type="text" value="3744"/> <p><input type="button" value="Submit"/> <input type="button" value="Reset"/></p>	<p>MapNavigator database: landschappen</p>  <p>change layers</p> <p>show legend</p> <p>identify object</p> <p>Zoom:</p> <p>Level: <input type="text" value="1"/></p> <p><input type="radio"/> FullExtent</p> <p><input type="radio"/> ZoomIn</p> <p><input type="radio"/> ZoomOut</p>	

figuur 5: de geo i-mode applicatie

Discussie en Conclusies

Het grote nadeel van WAP dat elk toestel zijn eigen WAP parameters nodig had is met I-mode ondervangen. Een site gedefinieerd in cHTML heeft een grotere kans om correct gerepresenteerd te worden op verschillende mobiele telefoons (mits de maxima gesteld aan scherm grootte en pagina omvang niet overschreden wordt).

Daarnaast is de kwaliteit van de grafische informatie duidelijk beter. Ondanks de kleine schermafmeting zijn kaartjes goed te interpreteren mits de extensie niet te groot is en schaal van de kaart niet te klein.

Een belangrijke beperking blijft het navigeren. Omdat clickable maps niet worden ondersteund in html zal pannen, zoomen ed via workarounds en via de numeriek toetsen moeten gebeuren.

I-mode is echter wel een aantrekkelijke manier om geo-informatie diensten aan te bieden die beperkt zijn tot het opvragen van informatie. Voorbeelden zijn bijvoorbeeld "waar is..." diensten of route informatie. Deze diensten vereisen relatief weinig interactie met de data. Een beperking is echter ook hierbij het ontbreken van locatiebepaling via de telefoon.

Wel is gebleken dat met een standaard mapserver relatief eenvoudig een I-mode dienst in de lucht is te brengen. Veel problemen die samenhangen met het WAP protocol zijn vervallen door het gebruik van cHTML

Voor het editen, en update van geo-informatie in I-mode niet geschikt. Daarmee vervalt I-mode als oplossing voor veldwerk.

3.3 Client-Server applicatie

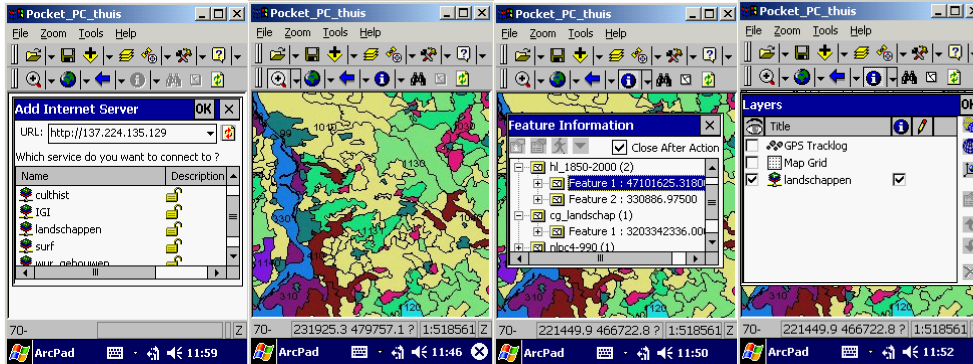
Bij de derde pilot die is uitgevoerd wordt gebruik gemaakt van een client op een PDA in combinatie met een draadloze verbinding met een server (of laptop). De client die gebruikt is in dit geval was ArcPad. Het voordeel van het gebruik van een client zijn krachtiger bevaag en bewerking mogelijkheden in vergelijking tot de statische WAP en Chtml oplossingen. Daarnaast kan via een client nieuwe geometrische object worden gedigitaliseerd iets wat bij de andere oplossing lastig is te realiseren.

Aanpak

Voor de test is ArcPad versie 6.0 toegepast. Eerdere versie hebben problemen met het juist projecteren van RD coördinaten. Als apparaat is compaq iPAQ gebruikt in combinatie met een navman gps systeem wat via een achterwan op de iPAQ can worden bevestigd. Op deze manier zijn GPS en PDA met elkaar geïntegreerd. Appendix 3 geeft een exacte handleiding voor het gebruik van de GPS in combinatie met Arc?pad. Daarnaast is de applicatie getest op een XDA van O2. Deze laatste PDA heeft niet de mogelijkheid om te koppelen met de Navman PGS.

Resultaten

Als test is de zelfde dataset beschikbaar gesteld via Arc/IMS. Het binnenhalen van de data is vrij traag via een GPRS verbinding (e.e.a. ook afhankelijk van het gebruikte apparaat) maar loopt in principe goed. Figuur 6 geeft een indruk van het gebruik van ArcPad.



Figuur 6: De ArcPad client applicatie

Discussie en Conclusies

Een belangrijk voordeel in het gebruik van ArcPad is de mogelijkheid om flexibel bevragingen op de data te kunnen doen en nieuwe geometrische objecten te kunnen digitaliseren. Daarnaast geen het gebruik van ArcPad betere mogelijkheden om gegevens te kunnen invoeren en te verwerken. De aanwezigheid van bijvoorbeeld een touchscreen maakt het selecteren van ruimtelijke objecten, zoomen en pannen een stuk eenvoudiger. Daarnaast is een koppeling met een GPS eenvoudig. Verschillende oplossingen zijn mogelijk. Met name het gebruik van de geïntegreerde Navman GPS levert een goed werkbare oplossing op.

Een belangrijk nadeel van ArcPad is de slechte mogelijkheden om de koppelingen met andere servers dan Arc/Info. Koppeling met Oracle Spatial bijvoorbeeld is niet mogelijk (alleen door gebruik te maken van SDE).

Daarnaast kan geometrische gegevens die in het veld zijn opgenomen lastig of niet direct (streaming) worden opgeslagen op de server. Dit kan alleen via een synchronisatie actie.

Veel van de eisen zoals gesteld in paragraaf 2.5 zijn met behulp van ArcPad te verwezenlijken. Enkele zaken zoals geometrische bevragingen kunnen niet met behulp van Arc/Info niet uit te voeren. Hiervoor zijn server gebaseerde oplossingen nodig. Wel moet een ArcPad client van tevoren op een pda zijn geïnstalleerd. Dit maakt deze oplossing minder geschikt voor het aanbieden van diensten aan derden.

Bijlagen

Appendix 1: Chtml notitie van het W3 consortium (niet compleet; complete versie via internet)

1. Introduction

The Compact HTML is a well-defined subset of HTML 2.0[1], HTML 3.2[2] and HTML 4.0[3] recommendations, which is designed for small information appliances. HTML defines flexible, portable, and practical document format for the documents on the Internet. One direction of HTML is to grow toward richer multimedia document format. A new recommendation HTML 4.0[3] includes new additional features. For example, CSS(Cascading Style Sheets) give a wider range of document styles. On the other hand, there must be another direction for small information appliances. Small information appliances have several hardware restrictions such as small memory, low power CPU, small or no secondary storage, small display, mono-color, single character font, and restricted input method (no keyboard and mouse). The browser for Compact HTML proposed in this document can be implemented in such a restricted environment. Once such a subset of HTML is defined, contents providers and information appliance manufacturers can rely on this common standard. We believe that Compact HTML definitely contributes to the rapid growth of small information appliance market.

2. Requirements of Small Information Appliances

2.1. Scope of the Products

First we describe the scope of target small information appliances. The categories of these devices are often referred to as smart phones, smart communicators, and mobile PDAs. There are some hardware restrictions for these devices. From the hardware point of view, we pick up main characteristics of the target devices.

Small memory

Typical case: 128-512Kbytes RAM
and 512K-1Mbytes ROM

Low power CPU

Typical case: 1-10 MIPS class CPU for embedded systems

Small display

Typical case: 50x30 dots, 100x72 dots, and 150x100 dots

Restricted colors

Typical case: mono-color (black and white)

Restricted character fonts

Typical case: only single font

Restricted input method

Typical case: several control buttons and number buttons (0-9)

2.2. Requirements

To realize the WWW browsing function for such small devices, a suitable subset of HTML is necessary. The requirements are derived from the above hardware restrictions. Also these devices should be easy to use from the standpoint as consumer products. The browser software for a subset of HTML should run within the small memory: e.g., 150-200Kbytes for the working data and also 150-200Kbytes for the program code. The minimum requirement for the CPU power should be 1-2 MIPS, though it may depend on the CPU power required for network communication processing. Easy navigation is also one of the key features for consumer devices. It means that the users can navigate information with a minimum number of operations. A subset of HTML should satisfy this requirement.

2.3. *Wireless Network*

Compact HTML does not depend on the underlying network protocol. In the typical cases, the transport protocol for Compact HTML is assumed to be HTTP over TCP/IP. However, current wireless communication networking for cellular phones is low band and low speed. In this area, the transport protocol should be defined as light protocol for better performance on the physical packet layer. It also seems useful to compress HTML contents so that most of HTML data can be stored within one packet data.

3. Definition of Compact HTML

3.1 *Design Principles*

The Compact HTML is designed to meet the requirements of small information appliances described above. It is designed based on the following four principles.

(1) Completely based on the current HTML W3C recommendations

Compact HTML is defined as a subset of HTML 2.0, HTML 3.2 and HTML 4.0 specifications. This means that Compact HTML inherits the flexibility and portability from the standard HTML.

(2) Lite Specification

Compact HTML has to be implemented with small memory and low power CPU. Frames and tables which require large memory are excluded from Compact HTML.

(3) Can be viewed on a small mono-color display

Compact HTML assumes a small display space of black and white color. However, it does not assume a fixed display space, but it is flexible for the display screen size. Compact HTML also assumes single character font.

(4) Can be easily operated by the users

Compact HTML is defined so that all the basic operations can be done by a combination of four buttons; Cursor forward, Cursor backward, Select, and Back/Stop(Return to the previous page). The functions which require two-dimensional focus pointing like "image map" and "table" are excluded from Compact HTML.

The definition of Compact HTML is derived straightforwardly from the above principles.

3.2 *Features of Compact HTML*

The Compact HTML is a subset of HTML 2.0, HTML 3.2 and HTML 4.0. We describe the major features which are excluded from Compact HTML, as follows.

- JPEG image
- Table
- Image map
- Multiple character fonts and styles
- Background color and image
- Frame
- Style sheet

We define that Compact HTML includes GIF image support. It should be noted that this subset does not require two-dimensional cursor moving, that is, it can be operated by using only four buttons. We can also expect that well-designed pages for small display fit the screen space and

the scrolling is not necessary. Actually the Compact HTML browser can display the pages like "deck of cards" by HDML[4].

Since the memory capacity is the most important issue in implementing the Compact HTML browser, we recommend the buffer limit for some functions.

INPUT

The maximum buffer size is 512 bytes.

SELECT

The maximum buffer size is 4096 bytes.

Though such a limitation belongs to the implementation issues, the common criteria is useful while developing devices.

One recommended implementation for the browser is to support the direct selection of anchors by using number buttons. For example, when five anchors are contained in an HTML page, the third anchor can be selected just by pressing the "3" button. (The HTML 4.0 specification includes a new attribute "accesskey" for the similar purpose of direct key assignment.)

Compact HTML Tag List

--Updated

--HTML(2.0:HTML2.0,

3.2:HTML3.2,

1998/01/27

4.0:HTML4.0)

--CH FCompact HTML

No	Elements	Attributes	HTML	CH	Comments
1	!-	-	2.0	CH	--
2	!DOCTYPE	-	2.0	CH	--
3	&xxx;	-	2.0	CH	--©,>,<,",®, ; --� `
4	A	name= href="URL" rel= rev= title= urn=(deleted from HTML3.2) methods=(deleted from HTML3.2)	2.0	CH CH - - - -	--
5	ABBR	-	4.0	-	--
6	ACRONYM	-	4.0	-	--
7	ADDRESS	-	2.0	-	--Only one font.
8	APPLET	-	3.2	-	--(Deprecated element in HTML4.0)
9	AREA	shape= coords= href="URL" alt= nohref	3.2	-	--
10	B	-	2.0	-	--Only one font.
11	BASE	href="URL"	2.0	CH	--
12	BASEFONT	size=	3.2	-	--Only one font. --(Deprecated element in HTML4.0)
13	BDO	-	4.0	-	--
14	BIG	-	3.2	-	--Only one font.
15	BLOCKQUOTE	-	3.2	CH	--
16	BODY	- bgcolor= background= text= link= vlink= alink=	2.0 3.2 3.2 3.2 3.2 3.2 3.2	CH - - - - - -	--Non-white colors are drawn as black.

17	BR	- clear=all/left/right	2.0 3.2	CH CH	--
18	BUTTON	-	4.0	-	--
19	CAPTION	-	3.2	-	--
20	CENTER	-	3.2	CH	--(Deprecated element in HTML4.0)
21	CITE	-	2.0	-	--Only one font.
22	CODE	-	2.0	-	--Only one font.
23	COL	-	4.0	-	--
24	COLGROUP	-	4.0	-	--
25	DD	-	2.0	CH	--
26	DEL	-	4.0	-	--
27	DFN	-	3.2	-	--
28	DIR	- compact	2.0	CH -	--(Deprecated element in HTML4.0)
29	DIV	- align=left/center/right	3.2	CH CH	--
30	DL	- compact	2.0	CH -	--
31	DT	-	2.0	CH	--
32	EM	-	2.0	-	--Only one font.
33	FIELDSET	-	4.0	-	--
34	FONT	size=n size=+n/-n color=	3.2	- - -	--Only one font. --(Deprecated element in HTML4.0)
35	FORM	action= method=get/post enctype=	2.0	CH CH CH	--
36	FRAME	-	4.0	-	--(Frameset DTD)
37	FRAMESET	-	4.0	-	--(Frameset DTD)
38	HEAD	-	2.0	CH	--
39	Hn	- align=left/center/right	2.0 3.2	CH CH	--
40	HR	- align=left/center/right size= width= noshade	2.0 3.2 3.2 3.2 3.2	CH CH CH CH CH	--
41	HTML	- version=	2.0 3.2	CH CH	--version="C-HTML 1.0".
42	I	-	2.0	-	--Only one font.
43	IFRAME	-	4.0	-	--(Frameset DTD)
44	IMG	src= align=top/middle/bottom align=left/right width= height= hspace= vspace= alt= border= usemap= ismap=	2.0 2.0 3.2 3.2 3.2 3.2 3.2 2.0 3.2 3.2 2.0	CH CH CH CH CH CH CH CH - -	--Large images compressed automatically.
45	INPUT	type=text name= size= maxlength= value=	2.0	CH CH CH CH CH	--Max character buffer 512 bytes.

		type=password name= size= maxlength= value=	2.0	CH CH CH CH CH	--
		type=checkbox name= value= checked	2.0	CH CH CH CH	--
		type=radio name= value= checked	2.0	CH CH CH CH	--
		type=hidden name= value=	2.0	CH CH CH	--
		type=image name= src= align=top/middle/bottom/left/right	2.0 2.0 2.0 3.2	-	--
		type=submit name= value=	2.0	CH CH CH	--
		type=reset name= value=	2.0	CH CH CH	--
		type=file name= value=	3.2	-	--
46	INS	-	4.0	-	--
47	ISINDEX	- prompt=	2.0 3.2	- -	--(Deprecated element in HTML4.0)
48	KBD	-	2.0	-	--Only one font.
49	LABEL	-	4.0	-	--
50	LEGEND	-	4.0	-	--
51	LI	- type=1/A/a/I/i type=circle/disc/square value=	2.0 3.2 3.2 3.2	CH - - -	--
52	LINK	href="URL" rel= rev= urn= methods= title= id=	2.0	-	--
53	LISTING	-	2.0	-	--Only one font. --(Obsolete element in HTML4.0)
54	MAP	name=	3.2	-	--
55	MENU	- compact	2.0	CH -	--(Deprecated element in HTML4.0)
56	META	name= http-equiv= content=	2.0	CH	--http-equiv="refresh" only.
57	NEXTID	n=	2.0	-	--Deleted from HTML3.2.
58	NOFRAMES	-	4.0	-	--(Frameset DTD)
59	NOSCRIPT	-	4.0	-	--
60	OBJECT	-	4.0	-	--
61	OL	-	2.0	CH	--

		type=1/A/a/I/i start= compact	3.2 3.2 2.0	- - -	
62	OPTGROUP	-	4.0	-	--
63	OPTION	- selected value=	2.0	CH CH -	--
64	P	- align=left/center/right	2.0 3.2	CH CH	--
65	PARAM	-	4.0	-	--
66	PLAINTEXT	-	2.0	CH	--(Obsolete element in HTML4.0)
67	PRE	- width=	2.0 3.2	CH -	--
68	Q	-	4.0	-	--
69	S	-	2.0	-	--(Deprecated element in HTML4.0)
70	SAMP	-	2.0	-	--Only one font.
71	SCRIPT	-	3.2	-	--
72	SELECT	name= size= multiple	2.0	CH CH CH	--Max character buffer 4 Kbytes.
73	SMALL	-	3.2	-	--Only one font.
74	SPAN	-	4.0	-	--
75	STRIKE	-	2.0	-	--(Deprecated element in HTML4.0)
76	STRONG	-	2.0	-	--Only one font.
77	STYLE	-	2.0	-	--
78	SUB	-	3.2	-	--
79	SUP	-	3.2	-	--
80	TABLE	- align=left/center/right etc. border= width= cellspacing= cellpadding=	3.2	-	--
81	TBODY	-	4.0	-	--
82	TD	- align=left/center/right valign=top/middle/bottom/baseline rowspan= colspan= width= height= nowrap	3.2	-	--
83	TEXTAREA	name= rows= cols=	2.0	CH CH CH	--Max character buffer 512 bytes.
84	TFOOT	-	4.0	-	--
85	TH	- align=left/center/right valign=top/middle/bottom/baseline rowspan= colspan= width= height= nowrap	3.2	-	--
86	THEAD	-	4.0	-	--
87	TITLE	-	2.0	CH	--
88	TR	-	3.2	-	--

		align=left/center/right valign=top/middle/bottom/baseline			
89	TT	-	2.0	-	--Only one font.
90	U	-	3.2	-	--(Deprecated element in HTML4.0)
91	UL	- type=disk/circle/square compact	2.0 3.2 2.0	CH - -	--
92	VAR	-	2.0	-	--Only one font.
93	XMP	-	2.0	-	--Only one font. --(Obsolete element in HTML4.0)

Appendix 2 : Geo-lmode pilot applicatie (incomplete code)

<pre> INIT <% 'init routine ' requires parameter mName sub Init() Set mConnector = Server.CreateObject("aims.ArcIMSConnector") mConnector.ServerName = "137.224.135.129" mConnector.ServerPort = 5300 Set mMap = Server.CreateObject("aims.Map") resultInit = mMap.InitMap(mConnector, mName) mMap.Width = 100 mMap.Height = 100 mMap.BackColor = 16777215 Set Session("MapObj") = mMap end sub %> </pre>
<pre> INITMAPPOSITION <% 'initMapPosition Sub initMapPosition 'kaartje terugbrengen op de vorige locatie ***** xMin = Request("xMin") yMin = Request("yMin") xMax = Request("xMax") yMax = Request("yMax") If (xMax = "") or (yMax = "") or (xMin = "") or (yMin = "") Then mMap.DoZoomToFullExtent() Set mExtent = mMap.Extent Zoomfac = 0 PanFac = 0 PanStep = 0 Else Set mExtent = mMap.Extent mExtent.XMin = xMin mExtent.YMin = yMin mExtent.XMax = xMax mExtent.YMax = yMax mMap.DoZoomToExtent mExtent End If End Sub %> </pre>
<pre> BUILD LEGEND <% 'buildLegend Sub buildLegend Set mLegend = mMap.Legend mLegend.CellSpacing = 10 mLegend.Title = "Legend" legendURL = mLegend.URL End Sub %> </pre>
<pre> BUILD LAYER COLLECTION <% 'buildLayerCollection sub BuildLayerCollection() </pre>

<pre> layer collectie bouwen ***** Set mLayers = mMap.Layers Layer = Request("layers") If Request("layers")="on" Then For i = 1 to mLayers.Count LayerName = "Layer" & i LayerVisibility = Request(LayerName) If LayerVisibility = "on" Then mLayers.item(i).visible = true Else mLayers.item(i).visible = false End If Next End If End sub %> </pre>
<p>Choose layers</p> <pre> <html> <head> <meta http-equiv="Content-Type" content="text/html; charset=windows-1252"> <meta name="GENERATOR" content="Microsoft FrontPage 4.0"> <meta name="ProgId" content="FrontPage.Editor.Document"> <title>Choose layers</title> </head> <body> <FORM action="gypsy_show_map.asp" > <!-- #include file="gypsy_init.inc" --> <% Dim mMap Dim mLayers Dim mName Dim Layer Dim mExtent Dim xMin, xMax, yMin, yMax mName = Request("MapName") Response.write "Choose layer for: "&mName&"
" Response.write "(!): attributes can be queried<p>" 'de webserver contacten (routine opgeslag in include file gypsy_init.inc) ***** call Init() 'layer collectie bouwen ***** set mLayers = mMap.Layers For i = 1 to mLayers.Count LayerName = "Layer" & i LayerVisibility = Request(LayerName) if LayerVisibility = "on" then mLayers.item(i).visible = true else mLayers.item(i).visible = false end if Next 'afhandeling keuze mogelijkheden voor layers ***** For i = mLayers.Count to 1 step -1 Set gr = mLayers.item(i).Renderer If TypeName(gr) = "GroupRenderer" Then For c = 1 To gr.Count If TypeName(gr.Item(c)) = "SimpleLabelRenderer" Then idoke = "(!)" End IF Next Else idoke = "" </pre>

```

                End If
                chkBoxName = "Layer" & i
                str = str & "<INPUT id=" & chkBoxName & " name=" & chkBoxName
                str = str & " type=checkbox"
            If mLayers.Item(i).Visible then
                str = str & " checked>"
            else
                str = str & ">"
            End If
                str = str & mLayers.Item(i).Name
                str = str & idoke
            str = str & "<br>"
        Next
        Response.write str

        'resultaten ophalen en opnieuw setten van de extend
        *****
        Set mExtent = mMap.Extent

%>
<INPUT type="hidden" value=<%=mName%> name="MapName">
<INPUT type="hidden" value=<%=Request("xMin")%> name="xMin">
<INPUT type="hidden" value=<%=Request("xMax")%> name="xMax">
<INPUT type="hidden" value=<%=Request("yMin")%> name="yMin">
<INPUT type="hidden" value=<%=Request("yMax")%> name="yMax">
<INPUT type="hidden" value="on" name="layers">
<INPUT type="Submit" value=" Refresh " >
</FORM>
</body>
</html>

```

ZOOM TO SEARCH STRING

```

<HTML>
<HEAD>
<TITLE></TITLE>
</HEAD>
<body>
<FORM action="gypsy_show_map.asp" method=post>
<!-- #include file="gypsy_init.inc" -->
<!-- #include file="initMapPosition.inc" -->
<!-- #include file="buildLayerCollection.inc" -->
<!-- #include file="doZoom.inc" -->
<!-- #include file="buildLegend.inc" -->
<%
    Dim mMap
    Dim mRecordset
    Dim keyNaam
    Dim zoekLaagNaam, layername
    Dim zoekVeldNaam
    Dim Action
    Dim zoekLaag
    Dim mapName
    Dim res
    Dim layernummer
    Dim mExtent
    Dim xMin, xMax, yMin, yMax

    'de webserver contacten (routine opgeslag in include file gypsy_init.inc)
    *****
    mName = Request("MapName")
    call Init()

    'collectie van layers die aangeboden worden bouwen
    *****
    call BuildLayerCollection()

    'zoomen naar de gewenste zoekstring
    *****
    Action = Request("action")

```



```
</body>
</HTML>
```

ZOOM TO RD COORDINATES

```
<HTML>
<HEAD>
<TITLE></TITLE>
</HEAD>
<body>
<FORM action="gypsy_show_map.asp" method=post>
<!-- #include file="gypsy_init.inc" -->
<!-- #include file="initMapPosition.inc" -->
<!-- #include file="buildLayerCollection.inc" -->
<!-- #include file="doZoom.inc" -->
<!-- #include file="buildLegend.inc" -->
<%
    Dim mMap
    Dim mRecordset
    Dim keyNaam
    Dim Action
    Dim zoekLaag
    Dim mapName
    Dim res
    Dim layernummer
    Dim mExtent
    Dim xMin, xMax, yMin, yMax
    Dim radius

    'de webserver contacten (routine opgeslag in include file gypsy_init.inc)
    *****
    mName = Request("MapName")
    call Init()

    'collectie van layers die aangeboden worden bouwen
    *****
    call BuildLayerCollection()
    If Action="zoompan" Then
        'kaart initialiseren wat betreft positie
        *****
        call initMapPosition()

        'zoom en panfunctie activeren
        *****
        call doZoomPan()
    End If

    ' zoomen naar de gewenste zoekstring
    *****
    Action = Request("action")
    If Action="Find" Then
        rd_X = Request("rd_X")
        rd_Y = Request("rd_Y")
        Set mapPoint = Server.CreateObject("aims.PointObject")           'Create
new point

        mapPoint.X = rd_X
        mapPoint.Y = rd_Y
        Set spobj = CreateObject("aims.Envelope")
        radius = Request("radius")
        if radius = 0 or radius = "" then
            radius = 500
        end if
        spobj.XMax = mapPoint.x + radius
        spobj.XMin = mapPoint.x - radius
        spobj.YMax = mapPoint.y + radius
        spobj.YMin = mapPoint.y - radius
        mMap.DoZoomToExtent spobj
    End If
```



```

Dim Action
Dim zoekLaag
Dim mapName
Dim res
Dim layernummer
Dim mExtent
Dim xMin, xMax, yMin, yMax

'de webservice contacten (routine opgeslag in include file gypsy_init.inc)
*****
    mName = Request("MapName")
    call Init()

'kaart initialiseren wat betreft positie
*****
    call initMapPosition()

'layer collectie bouwen
*****
        set mLayers          = mMap.Layers

        For i = 1 to mLayers.Count

            LayerName = "Layer" & i
            LayerVisibility = Request(LayerName)
            if LayerVisibility = "on" then
                mLayers.item(i).visible = true
            else
                mLayers.item(i).visible = false
            end if
        Next

'Aantal active layers tellen
*****
    Dim nrActive
    Dim mLayers
    Dim activeLayer
    Dim td
    Dim i
    Dim c
    nrActive = 0
    Set mLayers = mMap.Layers
    For i = mLayers.Count to 1 step -1
        If mLayers.Item(i).Visible Then
            nrActive = nrActive + 1
            activeLayer = i
            str =str&"&amp;Layer"&i&"=on"
            Response.write "<INPUT type='hidden' value='on' name=Layer"&i&" >"

        End If
    Next
    If nrActive > 1 Then
        Response.write "SORRY, cannot perform request<p>"
        Response.write "More then one layer active.<br>"
        Response.write "Please return to the MapNavigator and use choose layers to set only one layer active<br>"
    Else
        mMap.Refresh
        Set mExtent = mMap.Extent
        Set gr = mLayers.item(activeLayer).Renderer
        If TypeName(gr) = "GroupRenderer" Then
            For c = 1 To gr.Count
                If TypeName(gr.Item(c)) = "SimpleLabelRenderer" Then
                    Response.write "ObjectIdentifier<p>"
                    keyfield = gr.Item(c).Field
                    urlImage = mMap.GetImageAsUrl()

                    Response.write "<img src='&urlImage&'><p>"
                    Response.write "Object_id: <INPUT type='text' value='0' name='ObjectId'
size='5'><p>"

                    Response.write "<input type='submit' value='Identify' name='Verstuur'> <p>"

```

```

                                Exit For
                            End If
                        Next
                    Else
                        Response.write "The active layer does not support indentification of objects<br>"
                        Response.write "Please contact the webmaster to activate indentification support<br>"
                        Response.write "Please return to the MapNavigator and optionally choose another layer<p>"

                    End If

                    If Request("Action")= "identify" Then
                        Set mRecordset = mLayers.item(activeLayer ).Recordset
                        Set td = mRecordset.TableDesc
                        count = mRecordset.TableDesc.Count
                        key = request("ObjectId")
                        'Response.write "layer: " & mLayers.item(activeLayer ).Name&"<br>"
                        'Response.write keyfield&"="&key
                        For cc = 1 To td.Count
                            If td.FieldName(CLng(cc))= keyfield Then
                                If td.FieldType(CLng(cc))= 12 Then
                                    mRecordset.Filter.WhereExpression = keyfield&"="&key&""
                                Else
                                    mRecordset.Filter.WhereExpression = keyfield&"="&key
                                End IF
                            End IF
                        Next
                        If mRecordset.Count > 0 Then
                            mRecordset.MoveFirst
                            For i = 1 To td.Count
                                Response.write                                td.FieldName(CInt(i))&":
                                "&mRecordset.Fields.FieldValue(CLng(i))&"<br>"
                                Next
                            Else
                                Response.write "Geen resultaten<br>"
                            End If
                            'mRecordset.Filter.WhereExpression = ""
                        End If
                    End If
                End if
            %>
            <a
            href="gypsy_show_map.asp?xMin=<%=mExtent.XMin%>&amp;yMin=<%=mExtent.YMin%>&amp;xMax=<%=mExtent.XMax%
            >&amp;yMax=<%=mExtent.YMax%>&amp;Layers=on&amp;MapName=<%=mName%><%=str%>">MapNavigator</a>
            <INPUT type="hidden" value="identify" name="Action">
            <INPUT type="hidden" value=<%=mExtent.Ymax%> name="yMax">
            <INPUT type="hidden" value=<%=mExtent.Ymin%> name="yMin">
            <INPUT type="hidden" value=<%=mExtent.Xmax%> name="xMax">
            <INPUT type="hidden" value=<%=mExtent.Xmin%> name="xMin">
            <INPUT type="hidden" value=<%=Zoomfac%> name="zoomfac">
            <INPUT type="hidden" value=<%=mName%> name="MapName">
            </form>
            </body>
            </HTML>

```

SHOW MAP

```

<HTML>
<HEAD>
<TITLE>CGI I-mode maps</TITLE>
</HEAD>

<BODY>

<FORM action="gypsy_show_map.asp" method="post" >

<INPUT type="hidden" value="on" name="layers">

<!-- #include file="gypsy_init.inc" -->
<!-- #include file="initMapPosition.inc" -->

```



```

<!-- #include file="buildLayerCollection.inc" -->
<!-- #include file="doZoom.inc" -->
<!-- #include file="buildLegend.inc" -->
<%

    'vars
    '****
    Dim mMap
    Dim mLayers
    Dim mName
    Dim Layer
    Dim doZoom, doPan
    Dim Zoomfac, ZoomValue
    Dim Pandir, Panstep, PanFac, direction
    Dim mExtent
    Dim xmin, xmax, ymin, ymax
    Dim legendURL

    'defaults
    '*****
    ZoomValue= 0
    Pandir = " "
    Panstep = 0

    mName = Request("MapName")

    'de webserver contacten (routine opgeslag in include file gypsy_init.inc)
    '*****
    call Init()

    'collectie van layers die aangeboden worden bouwen
    '*****
    call BuildLayerCollection()

    'kaart initialiseren wat betreft positie
    '*****
    call initMapPosition()

    'zoom en panfunctie activeren
    '*****
    call doZoomPan()

'Legenda maken
'*****
    call buildLegend()

'resultaten ophalen en opnieuw zetten van de extend
'*****
    mMap.Refresh
    Set mExtent = mMap.Extent
    urlImage = mMap.GetImageAsUrl()
    Response.write "MapNavigator<br>"
    Response.write "database: "& mName&"<p>"
    Response.write ""

'afhandeling keuze mogelijkheden voor layers
'*****
    For i = mLayers.Count to 1 step -1
        If mLayers.Item(i).Visible Then
            str =str&"&Layer"&i&"=on"
            Response.write "<INPUT type='hidden' value='on' name=Layer"&i&" >"
        End If
    Next

%>

<br><a
href="gypsy_choose_layers.asp?xmin=<%=mExtent.XMin%>&amp;ymin=<%=mExtent.YMin%>&amp;xmax=

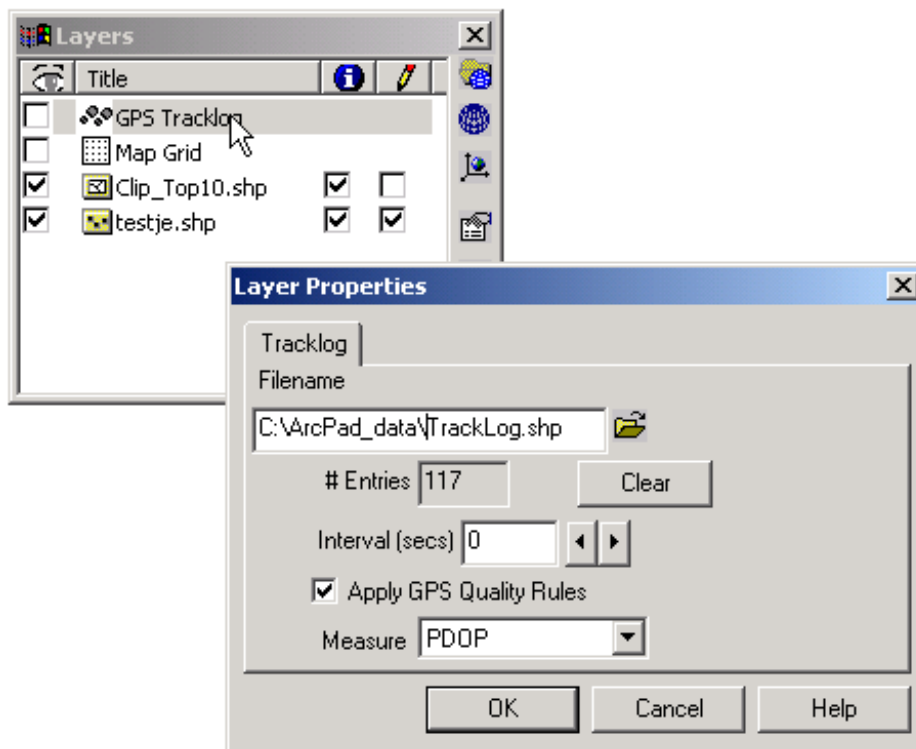
```


Appendix 3: Veldwerk met GPS gekoppeld aan ArcPad.

Een van de grote voordelen van het gebruik van ArcPad is de standaard mogelijke input via GPS. Dit biedt onder andere de mogelijkheid op locatie ruimtelijke gegevens direct in te voeren en te presenteren. Hieronder volgt een korte omschrijving van het gebruik van GPS input in ArcPad. Onderscheidt wordt hierbij gemaakt in 1. Het vastleggen van een route en 2. Het toevoegen van punt gegevens aan een dataset.

Het vastleggen van een route

ArcPad heeft als standaard optie het vastleggen van een route. De locatiebepaling hierbij gaat via GPS. De routepunten worden hierbij opgeslagen in een shape-file. Onder layers/tracklog kan een filenaam en directory ingesteld worden (dubbelklik GPS Tracklog). Via het vastleggen van opeenvolgende routes kan desgewenst gekozen worden om dit in verschillende shapes te doen. Hiervoor hoeft alleen de naam van de tracklogfile aangepast te worden.



De opgeslagen punten in een tracklog shape-files staan in lon/lat stelsel, dit is het standaard coördinatenstelsel waarin een GPS de gegevens naar een applicatie stuurt. Voor een verder gebruik van de tracklog shape-files zal deze dus eerst naar het juiste coördinatenstelsel geprojecteerd moeten worden.

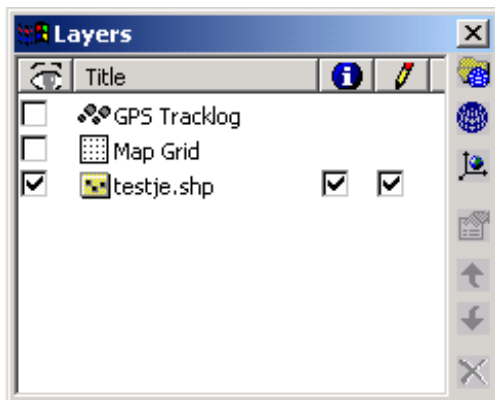
Indien een tracklog alleen dient voor het presenteren van een gelopen route in ArcPad, dan is voor het terughalen van de gegevens alleen het instellen van de gewenste tracklog shape-file onder layers/tracklog voldoende, ArcPad zorgt hier voor het juist projecteren van de gegevens.

Het starten van een vast te leggen route gebeurt door eerst de GPS onder de 'GPS' te activeren. Hierna kan de tracklog 'GPS Tracklog' en aangezet.

Het toevoegen van punt gegevens aan een dataset.

Voor het vastleggen van veldopnames met behulp van ArcPad is het mogelijk gebruik te maken van de digitaliseer optie met input dmv. een GPS. Voor het gebruik van deze optie is het noodzakelijk eerst een lege shape-file aan te maken. Dit kan in Arcpad mbv. de new layer button deze is te vinden onder:

Bij het aanmaken van een nieuwe shape-file wordt ook de table structuur vastgelegd, hierbij moeten de veld types worden gedefinieerd. Voeg als eerste veld altijd een ID met een numerieke waarde in. De andere velden zijn afhankelijk van de in te voeren veldwaarnemingen. Wanneer de nieuwe shape-file in ArcPad wordt aangemaakt is deze direct de te 'editen' laag. Wanneer veldgegevens aan een bestaande shape-file moeten worden toegevoegd, dan moet de laag eerst 'editable' gemaakt worden. Dit gebeurt onder layers . Hiermee wordt het window dat de aanwezige layers toont geopend. Om een laag editable te maken moet de kolom onder het pennetje aangevinkt z in het onderstaande voorbeeld is testje.shp de te 'editen' laag

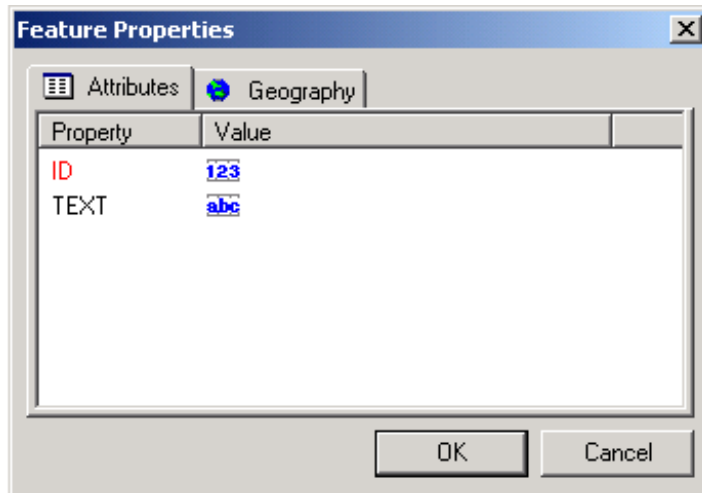


Er kunnen nu gegevens aan de dataset worden toegevoegd. In onderstaande beschrijving wordt er van uit gegaan dat veldgegevens op puntlocaties worden opgenomen. Naast het toevoegen van punten is het in ArcPad ook mogelijk lijnen en vlakken in te voeren. De in te voeren types zijn gekoppeld aan de 3 types shape-files (punt/lijn/vlak).

In te voeren punten kunnen op 2 manieren gedigitaliseerd worden. De locatie kan aan de hand van onderliggende referentielagen (bijv. top10) worden bepaald, waarbij de afwijking ten opzichte van de daadwerkelijke locatie relatief groot zal zijn. Of een punt kan bepaald worden met behulp van een GPS, hierbij zal de fout in locatiebepaling een stuk kleiner zijn.

Om gebruik te kunnen maken van een GPS moet deze eerst geactiveerd worden, Onder zit een opt GPS Active , met deze opt ontvangt ArcPad signalen vanuit het GPS.

In ArcPad is er een standaard tool aar...zig voor het invoeren van locaties met een GPS Bij de ir...r van elk punt wordt een scherm geopend voor het invoeren van de bijbehorende attribuu...veldgegevens. Velden kunnen worden gevuld door het te vullen veld in de property kolom aan te klikken.



De ingevoerde gegevens staan bij het invoeren met behulp van GPS direct in het coördinatenstelsel waarin de nieuwe shape-file is aangemaakt. Dit coördinatenstelsel komt overeen met het stelsel van de aanwezige referentielagen. In het voorbeeld is als onderlaag de topografische kaart (Top10) aanwezig, ArcPad gebruikt het coördinatenstelsel van de eerste laag die in een project aanwezig is. In de layer window is te zien met welk stelsel gewerkt wordt.

