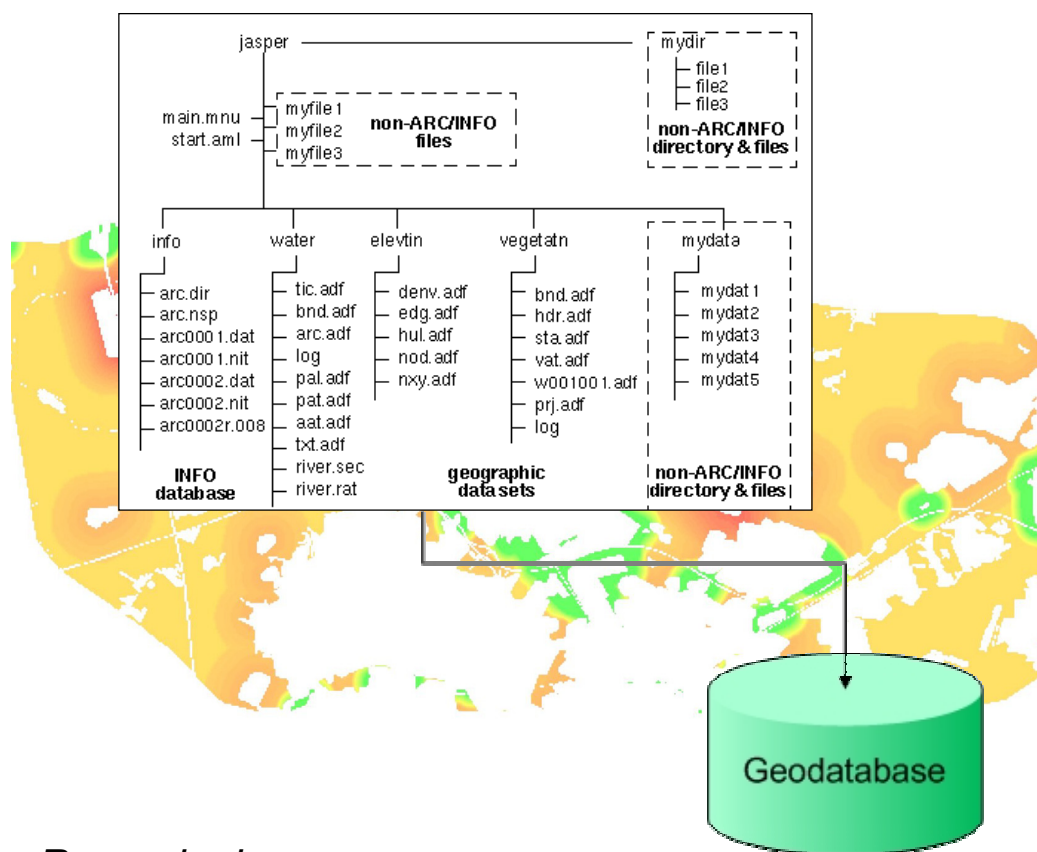


De migratie van een file directory structuur naar een GIS Geodatabase voor de applicatie GISMER

The migration from a file directory structure to a GIS Geodatabase of the application GISMER

Date: May 2004



By: *Wouter Rozendaal*
Reg. nr: 80.07.18.714.120



WAGENINGEN UNIVERSITY

WAGENINGEN UR

De migratie van een file directory structuur naar een GIS Geodatabase voor de applicatie GISMER

W.J.B. Rozendaal

Thesis report 2004-29

Wageningen Universiteit
In opdracht van Grontmij Geo Informatie

Inhoud

INHOUD	4
BIJLAGEN	6
FIGUREN	7
TABELLEN	8
WOORD VOORAF	9
SAMENVATTING	10
SUMMARY	12
1 INLEIDING	14
1.1 CONTEXT EN ACHTERGRONDEN	14
1.2 PROBLEEMVERKENNING	15
1.3 DOELSTELLING VAN HET ONDERZOEK	16
2 DE BESTAANDE GISMER APPLICATIE	17
2.1 FUNCTIONELE EN TECHNISCHE EISEN BESTAANDE GISMER APPLICATIE	17
2.2 FUNCTIONEEL ONTWERP BESTAANDE GISMER APPLICATIE	17
2.3 BENADERINGEN EN METHODEN IN GISMER	20
2.3.1 Beoordelingsmethode	20
2.3.2 Kwetsbaarheidbenadering	21
2.3.3 MCA-methoden	21
3 ONDERZOEKSMETHODEN	23
3.1 'SURVEY' NAAR FUNCTIONELE EISEN EN WENSEN TEN AANZIEN VAN GISMER	24
3.1.1 Opzetten van de 'survey'	24
3.1.2 Uitvoeren van de 'survey'	25
3.1.3 Analyseren resultaten van de 'survey'	26
3.2 HET MAKEN VAN EEN ONTWERP	26
3.2.1 Functioneel ontwerp	28
3.2.2 Systeem ontwerp	28
3.2.3 Technisch ontwerp	28
3.2.4 Implementatie	29
3.3 VERGELIJKING BESTAANDE MET NIEUWE APPLICATIE	30
3.3.1 Vergelijking functionaliteit en gebruikersinterface	30
3.3.2 Vergelijking databeheer	30
3.3.3 Vergelijking programmeeromgeving	30
4 RESULTATEN 'SURVEY' EN APPLICATIEONTWIKKELING	31
4.1 RESULTATEN 'SURVEY'	31
4.1.1 Kwetsbaarheidbenadering	31
4.1.2 Werking GISMER versie 1993; voorbereiding en voorbewerking	31
4.1.3 Werking GISMER versie 1993; bewerking en analyse	32
4.1.4 Beoordelingsmethode	33
4.1.5 MCA-methoden	33
4.1.6 ArcGIS	33
4.1.7 Overig	34
4.2 EEN ONTWERP VOOR DE NIEUWE APPLICATIE	34
4.2.1 Functioneel ontwerp	34
4.2.2 Systeem ontwerp	41
4.2.3 Technisch ontwerp	44
4.2.4 Implementatie	49

5	VERGELIJKING BESTAANDE EN NIEUWE APPLICATIE	52
5.1	VERANDERINGEN MET BETREKKING TOT DE FUNCTIONALITEIT EN DE GEBRUIKERSINTERFACE	52
5.2	VERANDERINGEN MET BETREKKING TOT HET DATABASEER.....	53
5.3	VERANDERINGEN MET BETREKKING TOT DE PROGRAMMEEROMGEVING	54
6	EVALUATIE	55
6.1	CONCLUSIES	55
6.2	KANTTEKENINGEN	57
6.3	AANBEVELINGEN	57

Bijlagen

BIJLAGE I: ONDERZOEKSOPZET (OORSPRONKELIJK ONDERZOEKSVORSTEL)	62
BIJLAGE II: CONTACTPERSONEN ‘SURVEY’	65
BIJLAGE III: VRAGENLIJST M.B.T. DE FUNCTIONELE EN TECHNISCHE EISEN EN WENSEN TEN AANZIEN VAN GISMER	68
BIJLAGE IV: OBJECT ORIËNTATIE EN SOFTWARE ONTWIKKELING.....	78
BIJLAGE V: OBJECTKLASSEN EN ATTRIBUTEN FUNCTIONEEL ONTWERP GISMER	85
BIJDRAGE VI: DATABASEHEER EN ONTWIKKELMETHODEN IN ARCINFO EN ARCGIS.....	95
BIJLAGE VII: SCRIPTS EN DIALOOGSCHERMEN GEGENEREERD DOOR ARCGIS DESKTOP 9.0 PRE RELEASE	105

Figuren

Fig. 1.1 Schaalbaarheid ArcGIS	14
Fig. 2.1 Globale functie-eisen bestaande GISMER applicatie (Grontmij. 1993, 3).....	17
Fig. 2.2 E.R. diagram GISMER	18
Fig. 2.3 Databeheer in GISMER	18
Fig. 2.4 Component 1; verrasteren	18
Fig. 2.5 Component 2; definiëren harde randvoorwaarden	19
Fig. 2.6 Component 3; selecteren harde randvoorwaarde.....	19
Fig. 2.7 Component 4; beoordelen van de aspecten	19
Fig. 2.9 Component 6; samenstellen visie.....	20
Fig. 2.10 Gewogen somming in de kwetsbaarheidbenadering	21
Fig. 3.1 Onderzoeksopzet.....	23
Fig. 3.3 Stappen in een ‘survey’ (Bron: Creative Research Systems, 2003).....	24
Fig. 3.4 Project-, Applicatie- en Applicatie ontwerp fasering (Vrij naar: www.geo-informatie.nl)	27
Fig. 4.1 Opbouw integrale kwetsbaarheidkaart volgens drie niveaus	32
Fig. 4.2 Lagen en partities van GISMER	34
Fig. 4.3 Componenten van GISMER	35
Fig. 4.4 Stroomdiagram component 1; “projectinstellingen”.....	36
Fig. 4.5 Legenda stroomdiagrammen Functioneel ontwerp	36
Fig. 4.6 Stroomdiagram component 2; importeren basisbestanden	37
Fig. 4.7 Stroomdiagram Component 3; creëren aspecten.....	38
Fig. 4.8 Stroomdiagram component 4; “beoordelen aspecten”	38
Fig. 4.9 Stroomdiagram component 5; “samenvoegen beoordeelde aspecten”	39
Fig. 4.10 Stroomdiagram component 6; “samenvoegen thema’s”	39
Fig. 4.11 Stroomdiagram component 7; “samenstellen harde randvoorwaarden”	40
Fig. 4.12 Stroomdiagram component 8; “toevoegen samengestelde harde randvoorwaarden” ...	41
Fig. 4.13 E.R.-diagram procesbewaking database.....	42
Fig. 4.14 Systeem ontwerp gebruikersinterface.....	43
Fig. 4.15 Legenda Data-actiemodellen Technisch ontwerp	44
Fig. 4.16 Data-actiemodel component 1; “projectinstellingen”, keuzeoptie; “nieuw project”	44
Fig. 4.17 Data-actiemodel component 1; “projectinstellingen”, keuzeoptie; “bestaand project”	45
Fig. 4.18 Data-actiemodel component 2; “importeren basisbestand”, keuzeoptie; “vector basisdata”	45
Fig. 4.19 Data-actiemodel component 2; “importeren basisbestand”, keuzeoptie; “vector basisdata”	46
Fig. 4.20 Data-actiemodel component 3; “creëren aspecten”, keuzeoptie; “conversie”	47
Fig. 4.21 Data-actiemodel component 3; “creëren aspecten”, keuzeoptie; “Bufferen in rasterformaat”	47
Fig. 4.22 Data-actiemodel component 3; “creëren aspecten”, keuzeoptie; “Beoordelingsattribuut aanpassen”	48
Fig. 4.23 Menustructuur GISMER	51

Tabellen

Tabel 3.1 Onderzoeksopzet	23
Tabel 5.1 Veranderingen met betrekking tot de functionaliteit	52
Tabel 5.2 Veranderingen met betrekking tot de gebruikersinterface	53
Tabel 5.3 Veranderingen met betrekking tot het databaseer	53
Tabel 5.4 Veranderingen met betrekking tot de programmeeromgeving	54

Woord vooraf

Voor mij was het in januari 2003 vrijwel zeker dat ik in September 2003 aan mijn afstudeerscriptie Geo-Informatiekunde zou kunnen beginnen. Mijn wens was dit in opdracht van en bij een ingenieursbureau te doen om zo naast het uitvoeren van wetenschappelijk onderzoek extra werkervaring op te doen.

Naar aanleiding van een college dat Sander Melieste had verzorgd over de toepassing van GIS en EVS bij locatiekeuzestudies voor de leerstoelgroep landgebruiksplanning van Wageningen Universiteit heb ik geïnformeerd naar mogelijkheden voor een afstudeervak bij Grontmij Geo Informatie. Daarbij had ik aangegeven dat mijn voorkeur uitging naar een onderwerp dat ligt op het raakvlak GIS en ruimtelijke ordening. Al snel kreeg ik een positief antwoord en na enkele malen contact te hebben gehad was het onderwerp van mijn scriptie globaal afgebakend. Vanuit Grontmij Geo Informatie ben ik gedurende het uitvoeren van dit onderzoek begeleid door Sander Melieste en Ronald van Lanen.

Vanuit Wageningen Universiteit vond de begeleiding plaats door Joep Cromptvoets tijdens de eerste fase van het afstudeeronderzoek en verder door John Stuiver.

Bij deze wil ik mijn vier begeleiders van harte bedanken voor de leerzame en prettige samenwerking. Mede dankzij hun heb ik tijdens de periode van dit onderzoek ook veel nieuwe kennis opgedaan van verscheidene GIS en IT concepten en heb ik altijd met veel plezier gewerkt.

Wouter Rozendaal

Samenvatting

Met de introductie van object georiënteerd programmeren en het beheer van (geografische) data in GIS Geodatabases is een compleet nieuwe generatie GIS software beschikbaar gekomen. Voor gebruikers betekende de introductie van deze nieuwe concepten een grote verandering, er wordt zelfs gesproken van migratie wat veel inspanning vraagt van de software gebruikers.

GISMER is een uit 1993 daterende Gisapplicatie die binnen Grontmij gebruikt wordt om Milieu Effect Rapportages (MER) en tracé- en locatiekeuzestudies mee te ondersteunen. De huidige applicatie is gebouwd in een commando gestuurde omgeving en het databeheer vindt plaats in een file directory structuur. Deze applicatie moet gemigreerd worden naar een object georiënteerde applicatie waarbij het databeheer plaatsvindt in een GIS Geodatabase.

De migratie van intern ontwikkelde applicaties kan gevolgen hebben vanuit drie aspecten. Deze aspecten zijn, de wensen ten aanzien van de gebruikersinterface en de functionaliteit van de applicatie, de dataopslag structuren en ten derde de programmeeromgeving.

Onduidelijk is nog welke gevolgen het object georiënteerd programmeren en het databeheer in een GIS Geodatabase hebben voor het ontwikkelen van een applicatie.

Door eerst de wensen die aan een nieuwe GISMER applicatie gesteld worden te onderzoeken, vervolgens een functioneel, systeem en technisch ontwerp voor de nieuwe applicatie te maken en tot slot de bestaande en de nieuwe GISMER applicaties met elkaar te vergelijken worden deze gevolgen in beeld gebracht.

Gewenste functionaliteit nieuwe GISMER applicatie (paragrafen 4.1)

Om de gewenste functionaliteit voor de nieuwe GISMER applicatie in beeld te brengen is een ‘survey’ gehouden onder potentiële toekomstige gebruikers en opdrachtgevers van de applicatie.

De nieuwe applicatie moet als basis dezelfde functionaliteit bezitten als de bestaande applicatie. Wel moet de bestaande functionaliteit uitgebreid worden en moet de applicatie anders op worden gebouwd. Een van de belangrijkste wensen die aan de functionaliteit worden gesteld is dat gemaakte keuzes bij het gebruik van GISMER op worden geslagen en vervolgens reproduceerbaar zijn.

Een ontwerp voor de nieuwe GISMER applicatie (paragrafen 3.2 en 4.2)

Bij het maken van het functioneel ontwerp is de applicatie eerst verdeeld in een achttal min of meer op zichzelf staande delen, applicatie-componenten. Voor alle acht componenten is de functionaliteit verder uitgewerkt.

Daarna is voor de hele applicatie een systeem ontwerp gemaakt. Hierin is de structuur van de gebruikersinterface ontworpen. In het menu moet een keuze gemaakt worden uit een component of component keuzeoptie. Binnen een component is er per stap een dialoogschermb. In het systeem ontwerp is gekozen voor een relationele ruimtelijke database voor de opslag van gegevens.

Vervolgens is voor de eerste drie componenten en voor de database een technisch ontwerp gemaakt. Hier zijn binnen de componenten de verschillende stappen en acties bepaald. Het technisch ontwerp is uitgewerkt aan de hand van data-actiemodellen.

Tot slot is het ontwerp voor de eerste drie componenten en de database geïmplementeerd in ArcGIS Desktop 9.0 Pre Release. Het uitgangspunt hierbij was dat alleen gebruik werd gemaakt van standaardfunctionaliteit. Daarmee was het niet mogelijk alle functionaliteit te implementeren zoals deze in het ontwerp staat omschreven.

Vergelijking (paragraaf 3.3 en hoofdstuk 5)

De bestaande en de nieuwe applicatie zijn met elkaar vergeleken op basis van drie aspecten, de functionaliteit en de gebruikersinterface, het databeheer en de programmeeromgeving.

Functionaliteit en gebruikersinterface

Per component van de nieuwe applicatie is bekeken of en hoe de functionaliteit in zowel de bestaande als de nieuwe applicatie voorkomt. Ook is met betrekking tot de gebruikersinterface gekeken naar de opbouw van beide applicaties.

De nieuwe applicatie heeft meer uitgebreide functionaliteit die zich voornamelijk uit in het toegenomen aantal keuzeopties en de introductie van procesbewaking. Ook is de nieuwe applicatie flexibeler opgebouwd door het mogelijk inbrengen van samengestelde harde randvoorwaarden (SHRV's) op drie niveau's.

Databeheer

Het databeheer in de nieuwe applicatie is aanzienlijk veranderd ten opzichte van de bestaande applicatie. De belangrijkste verandering is dat in de nieuwe applicatie een relationele ruimtelijke database (Geodatabase) gebruikt wordt als centrale opslagplaats voor data en gemaakte keuzes in tegenstelling tot een file directory structuur in de bestaande applicatie. Gemaakte keuzes zijn hierdoor reproduceerbaar.

Programmeeromgeving

De programmeeromgeving van de nieuw GISMER applicatie is gebaseerd op het Component Object Model (COM). Hierdoor zijn de resultaten van de verschillende ontwikkeltechnieken van ArcGIS uitwisselbaar. De ontwikkelaar kan de techniek gebruiken die voor het ontwikkelen van een bepaalde tool of functie het meest geschikt is. Voor het programmeren heeft de ontwikkelaar de keuze uit iedere COM-ondersteunende programmeertaal. De bestaande applicatie is geïmplementeerd in ArcInfo Macro Language, waarbij verder geen keuzemogelijkheden voor de ontwikkelaar aanwezig zijn.

Summary

When the object oriented programming method and the management of (geographic) data in GIS Geodatabases was introduced, a whole new generation of GIS software became available. For users of GIS software, the introduction of these new concepts meant a big change. There was even spoken about a migration what asks for a lot of effort of the software users.

GISMER is a GIS-application of 1993 which is used by Grontmij to carry out environment effect studies or to support location choice studies. The current application is build in a command based development environment and the data is managed in a file directory structure. This application must be migrated into an object oriented application in which the data is managed in a GIS Geodatabase. The migration of internal developed applications can have consequences for three aspects. These three aspects are user interface and functionality, data management and the development environment. It is not clear yet which consequences the object oriented programming method and data management in a Geodatabase cause for the development of an application.

By first investigating the wishes for the functionality of the new application and after that making a newfunctional, system and technical design GISMER and finally comparing the existing and the new application which each other, these consequences can be brought visible.

Wishes for functionality new GISMER application (paragraph 4.1)

To find out the wishes for the functionality of the new GISMER application, a survey is executed among potential future users and commissioners of the application.

The new application should have the same basic functionality as the existing application. The current functionality should be extended and the application itself should differently be structured. One of the most important wishes for functionality is that all choices that are made while using the application will be stored. These stored choices should also be reproducible.

A design for the new GISMER application (paragraphs 3.2 and 4.2)

The first step in making the functional design was to split up the application in eight more or less stand-alone parts, the application components. For every of the eight components the functionality is further worked out.

After that a system design is made for the entire application. In this system design the user interface structure is designed. In the menu a choice must be made out of the components or component choice options. Within a component there is a dialog box for every step. In the system design a relational, spatial database is chosen to store all the data.

After the system design, a technical design is made for the first three component of the application and for the database. In this technical design the components are further parted into steps and actions. The technical design is worked out with the help of data action models.

Finally, the first three components and the database are implemented in ArcGIS Desktop 9.0 Pre Release. Only standard functionality of this package is used. With the standard functionality it appeared not possible to implement all the functionality as it is described in the design.

Comparison

The existing and new applications have been compared on three aspects, the functionality and user interface, data management and the development environment.

Functionality and user interface

For every component of the new application has been looked if and how the functionality in both, existing and new application occur. Also has been looked towards the user interface to the structure of both applications.

The new application has more extensive functionality which is expressed mostly by the increased number of component choice options and the introduction of proceswatching. The new application has also a more flexible structure by means of the possibility to apply the “samengestelde harde randvoorwaarden (SHRV’s)” at three levels.

Data management

Data management in the new application has considerably changed compared to the existing application. The most important change is the use of a relational spatial database (Geodatabase) in the new application as the central place to store all the data and choices that have been made in contradiction to the file directory structure to store the data in the existing application. Choices that have been made are reproducible in the new application.

Development environment

The development environment of the new GISMER application is built on the Component Object Model (COM). Because of that, the products of the different development techniques of ArcGIS are exchangeable. The developer can choose the most appropriate development method to develop a certain tool or function. The programming language can be chosen by the developer; the only demand is that the language supports COM. The existing application is developed with ArcInfo Macro Language. With this language there are no further choosing options for the developer.

1 Inleiding

1.1 Context en achtergronden

Met de introductie van object georiënteerd programmeren en het beheer van (geografische) data in GIS Geodatabases is een compleet nieuwe generatie GIS software beschikbaar gekomen. Een van de nieuwe software pakketten is ArcGIS van ESRI. Dit pakket biedt een complete set standaardfuncties om data te creëren, beheren, integreren, analyseren en presenteren (ESRI, 2003). ESRI heeft aangekondigd dat alle nieuwe ontwikkelingen gericht zullen zijn op ArcGIS en dat er geen uitbreidingen meer komen van Arcview 3.3 (Knippel, 2002). Dit zijn enkele redenen waarom er bij Grontmij Geo Informatie, de opdrachtgever voor dit onderzoek, voor gekozen is over te stappen naar de nieuwe generatie GIS software en gebruik te maken van ArcGIS.

Voor gebruikers betekende de introductie van deze nieuwe generatie software een vrij grote verandering (Burgerhart *et. al.* 2002). Er wordt zelfs gesproken van migratie, wat inhoudt dat er een situatie is ontstaan waarin veranderingen veel groter zijn dan bij de overgang naar een totaal ander product het geval zou zijn. Migratie kan betekenen dat het opnieuw trainen van gebruikers en het opnieuw ontwikkelen van data, applicaties en procedures nodig is. Dit in tegenstelling tot een upgrade waarbij software uitgebreid wordt. De software behoudt in het geval van een upgrade wel de eigen bekende gebruikers interface en werkwijze (Knippel, 2002). De ontwikkeling van GIS-applicaties die voor de introductie van object georiënteerd programmeren plaatsvond gebeurde veel met behulp van commando gestuurde talen zoals ArcInfo Macro Language (AML) (ESRI, 1990) van ESRI. Applicatie ontwikkeling binnen de nieuwe generatie GIS-software is gebaseerd op de COM structuur (microsoft, 2003, 1). Applicaties worden opgebouwd uit een aantal losse COM modules en de programmeertaal is vrij te kiezen.

De migratie van intern ontwikkelde applicaties van een kan gevolgen hebben vanuit drie aspecten. Deze aspecten zijn, de wensen ten aanzien van de gebruikersinterface en de functionaliteit van de applicatie, de dataopslag structuren en ten derde de programmeeromgeving.

GISMER is een uit 1993 daterende Gisapplicatie die binnen Grontmij gebruikt wordt om Milieu Effect Rapportages (MER) en tracé- en locatiekeuzestudies mee te ondersteunen. De huidige applicatie is gebouwd met AML en bestaat uit een aantal aan elkaar gekoppelde macro's (Van Capelleveen *et. al.*, 1997). Grontmij Geo Informatie wil graag weten wat de consequenties zijn van de migratie van GISMER naar de nieuwe generatie software met betrekking tot de drie bovengenoemde aspecten. De kennis die wordt opgedaan bij dit project kan dan verder gebruikt worden bij het migreren van andere applicaties of het ontwikkelen van nieuwe applicaties.

De functionaliteit van commando gestuurde programmeertalen en software pakketten zoals AML dat gebruikt wordt door ArcInfo workstation is in vele gevallen omslachtig en beperkend ten opzichte van de functionaliteit van object georiënteerde software en programmeertalen (Fitzgerald, 2002). Wanneer AML als ontwikkeltaal gebruikt wordt worden gebruikers van applicaties die met deze taal geprogrammeerd zijn gedwongen ook gebruik te maken van een dure ArcInfo licentie terwijl dit met ArcGIS niet nodig is. ArcGIS bestaat uit een aantal losse componenten die onderling een hiërarchische structuur hebben. Gebruikers kiezen het pakket dat de functionaliteit biedt die zij nodig hebben. Wanneer de eisen van de gebruiker toenemen, kan er altijd een upgrade plaatsvinden (ESRI, 2003). ArcGIS Desktop software is schaalbaar en is te verkrijgen in drie varianten, zoals in Fig. 1.1 te zien is. Hierbij is Arcview de goedkoopste variant waarmee GIS-data bewerkt en geanalyseerd kan worden en heeft ArcInfo de meest uitgebreide bewerking- en analysefunctionaliteit. De standaardfuncties van de lijn ArcGIS producten kunnen uitgebreid worden met extensies. De opdrachtgever van dit onderzoek, Grontmij Geo Informatie, heeft aangegeven dat zij graag zou willen weten wat de mogelijkheden zijn om de nieuwe versie GISMER als uitbreiding te produceren zodat deze gebruikt kan worden in combinatie met alleen een ArcGIS Arcview licentie inclusief Spatial Analyst extensie.

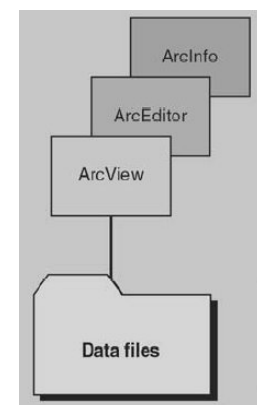


Fig. 1.1
Schaalbaarheid
ArcGIS

1.2 Probleemverkenning

Zoals gezegd, dateert de huidige GISMER applicatie uit 1993. Deze is dan ook gebaseerd op de toenmalige eisen die gesteld werden vanuit de MER-regelgeving en op de wensen die er toen waren met betrekking tot de analysemethode. Het is mogelijk dat deze wensen veranderd zijn. Daarnaast zijn de functionaliteit en de gebruikersinterface van de huidige applicatie gebaseerd op de technische mogelijkheden die in 1993 aanwezig waren. Ook de wensen ten aanzien van de gebruikersinterface kunnen veranderd zijn.

De nieuwe generatie GIS software brengt vele veranderingen met zich mee. Het aantal standaardfuncties voor het bewerken van de data is toegenomen en het is mogelijk zelf modules te ontwikkelen en toe te voegen (ESRI, 2003). Twee belangrijke veranderingen bij de migratie van de oude naar de nieuwe generatie software zijn de introductie van de Geodatabase en het object georiënteerd programmeren (microsoft 2003, 2). Voor gebruikers van alleen standaard functionaliteit is vooral de introductie van de Geodatabase van belang, voor programmeurs is daarnaast ook de introductie van het object georiënteerd ontwikkelen van applicaties van groot belang.

De Geodatabase is een opslagplaats voor geografische informatie in een database management system (DBMS). Door de totaal andere structurering van data, ten opzichte van het op files en directories gebaseerde datamodel van bestaande software brengt de Geodatabase vele extra mogelijkheden met zich mee (ESRI, 2001, 2002 en Meitner, 2003). De Geodatabase kan verschillende rollen vervullen binnen een applicatie. Ten eerste kan hij gebruikt worden als fysieke opslagplaats van geografische data en ten tweede kan hij gebruikt worden als middel om meerdere gebruikers tegelijkertijd toegang te geven tot de data. Tot slot kan er intelligentie in de vorm van gedrag toegekend worden aan ruimtelijke objecten. Het gedrag van ruimtelijke objecten beschrijft onder andere welke relaties deze hebben en hoe ze gewijzigd kunnen worden (ESRI 2001, 2002). Welke rol of rollen de Geodatabase moet gaan spelen in de nieuwe GISMER applicatie is nog niet duidelijk. Ook is er nog de vraag hoe er precies met de Geodatabase omgegaan moet worden bij het programmeren van applicaties.

De nieuwe generatie GIS-software is gebaseerd op het Component Object Model (COM) (Santiago Comella-Dorda, 2001). Dit is geen programmeertaal, maar een protocol dat een software component verbindt met een andere component. Door gebruik te maken van dit protocol kunnen dus losse, opnieuw te gebruiken bouwstenen voor software geproduceerd worden die makkelijk te combineren zijn met de bestaande applicaties (ESRI, 2001, 1 en). Programmeren aan de hand van het COM model wordt ook wel object georiënteerd programmeren genoemd. (microsoft, 2003, 1). Dit maakt het mogelijk dat iedere COM ondersteunende programmeertaal gebruikt kan worden om bestaande applicaties uit te breiden of nieuwe te ontwikkelen. Object georiënteerd programmeren heeft vele voordelen ten opzichte van een commando gerichte programmeertaal als Arc Macro Language (AML) (Fitzgerald, 2002). Welke dit exact zijn voor GISMER en of het object gericht programmeren ook nadelen oplevert voor het ontwikkelen van GISMER is nog niet bekend.

Probleemstelling:

GISMER moet gemigreerd worden van een file directory structuur (ArcInfo – AML) naar een GIS Geodatabase en van een commandoregel gebaseerde applicatie naar een object georiënteerde applicatie. Onduidelijk is nog welke gevolgen het inbrengen van de Geodatabase en het object georiënteerd programmeren hebben voor het ontwikkelen van de applicatie in ArcGIS.

1.3 Doelstelling van het onderzoek

Hoofddoelstelling

Het analyseren van de impact voor gebruikers en programmeurs van de introductie en de implementatie van de Geodatabase en het object georiënteerd programmeren op de migratie van de applicatie GISMER naar ArcGIS.

Subdoelstellingen:

- Het opzetten en uitvoeren van een “survey” ten aanzien van de functionele eisen en wensen die aan de nieuwe GISMER applicatie worden gesteld.
- Het maken van een functioneel ontwerp voor de nieuwe GISMER applicatie.
- Het maken van een technisch ontwerp voor een bouwsteen voor de nieuwe GISMER applicatie.
- Het implementeren van het technisch ontwerp om een bouwsteen voor een prototype van GISMER te leveren.
- Het onderzoek uitvoeren in opdracht van Grontmij Geo Informatie

De onderzoeksvragen die uit deze doelstellingen voortkomen, vormen de basis voor het onderzoek en de opzet daarvan.

Onderzoeksvragen:

- Wat is de impact van de introductie en de implementatie van de Geodatabase en het object georiënteerd programmeren op de migratie van de applicatie GISMER naar ArcGIS?
- Aan welke functionele eisen en wensen moet de nieuwe GISMER applicatie voldoen?
- Hoe ziet een functioneel ontwerp voor de nieuwe GISMER applicatie er uit?
- Hoe ziet een technisch ontwerp voor een bouwsteen van de nieuwe GISMER applicatie er uit?

2 De bestaande GISMER applicatie

GISMER is een applicatie waarmee gebieden met behulp van een multi criteria analyse beoordeeld kunnen worden op hun kwetsbaarheid voor bepaalde typen van ruimtegebruik. Naast het ondersteunen van milieu effect rapportages wordt GISMER ook veel gebruikt bij het uitvoeren van tracé- en locatiekeuzestudies (Commandeur en Kreft, 1998). De bestaande GISMER applicatie is tot stand gekomen in een intern Grontmij project dat is uitgevoerd in 1992 en 1993. De aanleiding voor dit project was dat de mogelijkheden om GIS toe te passen als instrument bij complexe analysevraagstukken, waarbij de ruimtelijke component een bindend gegeven is, nog niet effectief benut werden. Gezien de marktontwikkelingen, onder meer ten aanzien van GIS, en de potentiële mogelijkheden voor Grontmij als adviesbureau om hierop in te springen werd een extra inspanning wenselijk geacht (Grontmij, 1993, 1). Als doel van dit project werd gesteld: “Het, door het ten volle benutten van de mogelijkheden van GIS, krijgen van een groter marktaandeel in projecten waarbij GIS als ondersteunende techniek een essentiële rol speelt” (Grontmij, 1993, 2). Aangezien de toenemende marktvraag voor de inzet van GIS grotendeels uit de MER-markt kwam, is er voor gekozen MER ook als toepassingsgebied voor het verdere project te gebruiken. Milieu effect rapportages moesten steeds goedkoper en vanuit de commissie MER werd gevraagd naar een betere kwantificering van de effecten en een betere inzichtelijkheid en presentatie (Grontmij, 1993, 2). Ook was de druk op standaardisering erg hoog wegens de toename van het aantal verplichte milieu effect rapportages. GISMER is het resultaat van dit project. Het projectteam werd gevormd door mensen uit verschillende Grontmij afdelingen en bestond uit mensen met GIS kennis en mensen met MER kennis.

2.1 Functionele en technische eisen bestaande GISMER applicatie

Bij de ontwikkeling van de bestaande GISMER applicatie was er een analyse gemaakt van de werkwijze zoals deze bij het uitvoeren van een MER wordt gehanteerd. Hieruit kwamen een aantal globale functie eisen naar voren.

- Gegevens dienen met behulp van GIS (ArcINFO) te kunnen worden verwerkt
- Het gehele systeem dient flexibel te zijn
- Basisgegevens voor het systeem is geografische data
- Delen van de gegevens dienen voor verwerking uitgesloten te kunnen worden
- Gegevens dienen gescoord te kunnen worden op een schaal van 0 tot 100
- Gescoorde gegevens moeten met een gewichtsverdeling samengevoegd kunnen worden tot een thema
- Thema's dienen met een gewichtsverdeling samengevoegd kunnen worden tot een visie
- Resultaten moeten tijdens het proces visueel inzichtelijk gemaakt kunnen worden en in de vorm van plots gepresenteerd kunnen worden
- Het deel dat binnen een applicatie wordt gevat, moet eenvoudig te bedienen zijn

Fig. 2.1 Globale functie-eisen bestaande GISMER applicatie (Grontmij. 1993, 3)

Vervolgens is op basis van deze globale functie-eisen het functioneel ontwerp voor de bestaande GISMER applicatie gemaakt.

2.2 Functioneel ontwerp bestaande GISMER applicatie

In figuur 2.2 is het entiteitrelatie diagram van GISMER weergegeven. Hierin staat de invoer onder aan het stroomdiagram en de uitvoer bovenaan. De tussenliggende kaders bevatten allemaal tussentijdse

resultaten. De invoer voor GISMER bestaat uit een aantal basisbestanden en de uitvoer is een visie of integrale kwetsbaarheidkaart. Beoordeelde aspecten en thema's worden ook gebruikt als hulpmiddel bij het verder uitvoeren van de MER of locatiekeuze. Voordat er met de bewerkingen en analyse in GISMER begonnen kan worden, is er een voorbereidingsfase. De applicatie GISMER bestaat in totaal uit een zestal componenten die hieronder een voor een toegelicht worden. De figuren in deze paragraaf zijn afkomstig uit Van Capelleveen *et al.* 1997.

Het databeheer binnen GISMER vindt plaats in een file directory structuur, namelijk in ArcInfo workspaces (ArcInfo help en bijlage 0). Het hele GISMER project wordt beheerd onder een hoofdworkspace, de 'werkdir' met daaronder acht sub-workspaces. De opeenvolging van de sub-workspaces loopt vrijwel synchroon met het entiteitrelatie diagram. In algemene zin bevindt zich tussen iedere component een workspace. Hierop zijn twee uitzonderingen, een binnen de tweede component (Fig. 2.5) en een bij de uitvoer van de derde component (Fig. 2.6). Hieronder zal bij iedere component ook beschreven worden welke workspace zich aan het begin en welke aan het einde bevindt. Alle data is in ArcInfo GRID formaat met uitzondering van de invoer, dit zijn ArcInfo coverages.

2.2.1.1 Voorbereidingsfase

Bij de start van een MER wordt door de opdrachtgever bepaald welke thema's zoals ecologie, landschap, wonen enz. onderzocht moeten worden. Vervolgens wordt voor deze thema's data verzameld. Al deze data moet bewerkt en tot aspecten samengevoegd worden. Een aspect heeft dus vaak meerdere databronnen. Meerdere aspecten vormen samen een thema. De aspecten cultuurhistorische waarden, archeologische waarden enz. kunnen bijvoorbeeld samen het thema landschap vormen. De gegevensvoorbereiding gebeurt volledig handmatig en is dus niet geautomatiseerd in GISMER. (Grontmij, 1993, 3). Het resultaat van deze stap is een set topologisch correcte ArcInfo coverages.

2.2.1.2 Component 1; verrasteren

Als de aspecten eenmaal verzameld zijn, moeten ze verder verbewerkt worden zodat de analyses in GISMER uitgevoerd kunnen worden. Hiervoor moeten alle bestanden in een uniform dataformaat (ArcInfo GRID met eenduidige celgrootte) komen waarin alle bewerkingen gedaan kunnen worden. Deze gegevensvoorbewerking vindt plaats in de eerste component van GISMER, het verrasteren.

De basisbestanden moeten ArcInfo coverages zijn en in de 'cov'-workspace staan. De resultaten van deze component zijn de basisaspecten in ArcInfo GRID formaat met een eenduidige celgrootte. Deze worden opgeslagen in de 'aspct'-workspace (fig 2.3).

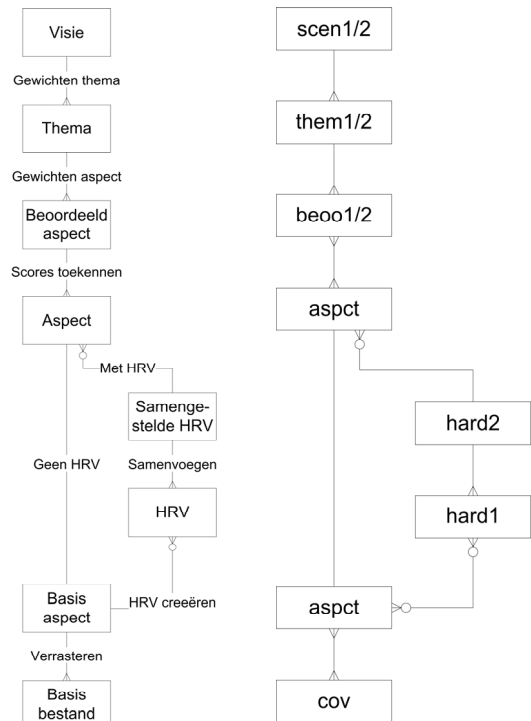


Fig. 2.2 E.R. diagram **Fig. 2.3 Databeheer in GISMER**

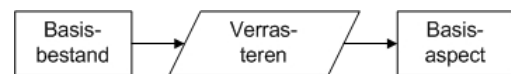


Fig. 2.4 Component 1; verrasteren

2.2.1.3 Component 2; definiëren harde randvoorwaarden

Binnen GISMER kunnen harde randvoorwaarden opgenomen worden. Dit zijn gebieden die bij de verdere beoordeling niet meer mee worden genomen. Dit definiëren van harde randvoorwaarden is de tweede component van GISMER en moet gebeuren voordat met de beoordeling van de aspecten begonnen kan worden. Eerst moeten per aspect de harde randvoorwaarden gedefinieerd worden. De eerste stap hiervoor is het selecteren van de methode, gekozen kan worden tussen absoluut of in klassen. De volgende stap is het opgeven van de selectie. Hierin moet aangegeven worden welke waarden van het aspect niet in aanmerking komen voor de toekomstige ruimtelijke maatregel. Wanneer deze eerste twee stappen voor alle aspecten zijn uitgevoerd kunnen in de derde en laatste stap de harde randvoorwaarden samengevoegd worden. Het eindresultaat is een samengestelde harde randvoorwaarde (SHRV). Er kunnen slechts twee sets samengestelde harde randvoorwaarden gemaakt worden.

In de tweede component bevindt zich de eerste uitzondering met betrekking tot het databeheer. Deze component heeft namelijk met drie workspaces te maken. Een voor de invoer, een binnen de component en een voor de uitvoer. De 'aspct'-workspace (fig 2.3) wordt als invoer gebruikt, in de 'hard1'-workspace worden per aspect de datasets met de harde randvoorwaarden opgeslagen en in de 'hard2'-workspace komen de samengestelde harde randvoorwaarden te staan.



Fig. 2.5 Component 2; definiëren harde randvoorwaarden

2.2.1.4 Component 3; selecteren harde randvoorwaarde

Voordat de aspecten gescoord kunnen worden, moeten ze eerst omgezet worden van basisaspecten in aspecten. Dit gebeurt in de derde component van GISMER. Hierin kan een harde randvoorwaarde geselecteerd worden die dan toegepast wordt op alle basisaspecten. Indien er geen harde randvoorwaarden zijn, moet in deze stap de standaard lege harde randvoorwaarde gekozen worden.

Binnen deze component wordt niets aan een dataset veranderd, er wordt alleen een samengevoegde harde randvoorwaarde toegewezen aan de aspecten.



Fig. 2.6 Component 3; selecteren harde randvoorwaarde

2.2.1.5 Component 4; beoordelen van de aspecten

Wanneer een harde randvoorwaarde geselecteerd is, kunnen de aspecten beoordeeld worden. Alleen de aspecten waaraan een gekozen samengestelde harde randvoorwaarde is toegekend in de derde component, kunnen beoordeeld worden.

De eerste stap is het kiezen van een methode. Binnen GISMER zijn verschillende beoordelingsmethoden toepasbaar. De verschillende methoden worden verderop in dit hoofdstuk toegelicht. De tweede stap in deze component is het toekennen van de score. Alle waarden van het aspect krijgen een score. Het is mogelijk om binnen deze component twee beoordeelde aspecten samen te voegen tot een beoordeeld aspect. Hierbij worden beide aspecten even zwaar gewogen. Deze laatste stap is optioneel en hoeft niet altijd doorlopen te worden.

De invoer voor de vierde component staat in de 'aspct'-workspace. Afhankelijk van welke samengestelde harde randvoorwaarde gekozen is, komen de beoordeelde aspecten in de 'beoo1'- of de 'beoo2'-workspace te staan.

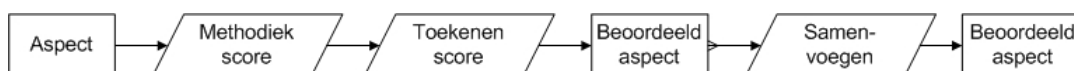


Fig. 2.7 Component 4; beoordelen van de aspecten

2.2.1.6 Component 5; samenstellen thema's

Thema's worden binnen GISMER opgebouwd uit een aantal samengevoegde beoordeelde aspecten.

Het samenvoegen van de aspecten gaat

volgens de multicriteria methode gewogen somming. Deze methode wordt later in dit hoofdstuk verder toegelicht. Eerst moeten de beoordeelde aspecten samengevoegd worden. Vervolgens moeten gewichten aan de aspecten worden toegekend die gebruikt worden bij de gewogen somming. Het resultaat van deze component is per thema een raster die de kwetsbaarheid van dat thema voor de te nemen ruimtelijke maatregel weergeeft. Dit tussenresultaat van het hele GISMER proces wordt ook vaak gebruikt bij het verder uitvoeren van de MER of de locatiekeuze.

Component 5 gebruikt de beoordeelde aspecten uit de 'beoo1'- of 'beoo2'-workspace als invoer, afhankelijk van de gekozen samengestelde harde randvoorwaarde. De uitvoer wordt in de 'them1'- of 'them2'-workspace weggeschreven. De thema's in de 'them 1' workspace zijn opgebouwd met beoordeelde aspecten uit de 'beoo 1' workspace en thema's in de 'them 2' workspace uit beoordeelde aspecten uit de 'beoo 2' workspace.

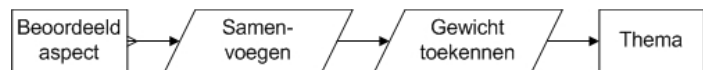


Fig. 2.8 Component 5; samenstellen thema's

2.2.1.7 Component 6; samenstellen visie

De zesde component van GISMER waarin een visie wordt samengesteld uit de thema's is op dezelfde manier opgebouwd als de vijfde component. Nu moeten eerst de relevante thema's

samengevoegd worden en moet vervolgens per thema een gewicht worden toegekend. Het eindresultaat van de analyse in GISMER is een visie of integrale kwetsbaarheidkaart met daarop aangegeven de mate waarin ieder gebied kwetsbaar is voor de te nemen ruimtelijke maatregel.

Het databeheer in de zesde component vindt op dezelfde manier plaats als in de vijfde component. Nu wordt alleen een van de twee 'them?'-workspaces als invoer gebruikt en wordt de uitvoer in een van de twee 'scen?'-workspaces weggeschreven.

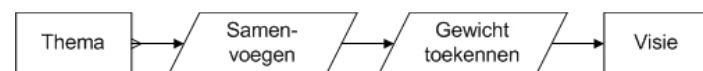


Fig. 2.9 Component 6; samenstellen visie

Bij iedere component is het mogelijk een korte beschrijving van het resultaat op te slaan. Deze wordt weggeschreven in een INFO-tabel (ArcInfo help) en is slecht te raadplegen nadien.

2.3 Benaderingen en methoden in GISMER

Binnen GISMER wordt de kwetsbaarheidbenadering gehanteerd om relatieve kwetsbaarheid of geschiktheid van een gebied in kaart te brengen. Bij het berekenen van de relatieve kwetsbaarheid of geschiktheid wordt gebruik gemaakt van de multicriteria (MCA) methode gewogen somming. Voor het scoren van de aspecten kunnen in GISMER verschillende methodes gekozen worden. Welke methodes dit zijn, wat de kwetsbaarheidbenadering inhoudt, en welke MCA methode gebruikt wordt, wordt hieronder verder toegelicht.

2.3.1 Beoordelingsmethode

De scores die aan de aspectwaarden toegekend worden liggen in principe tussen 1 en 99. Lage scores stellen een lage kwetsbaarheid voor en hoge scores een hoge kwetsbaarheid. De scores 0 en 100 hebben een speciale betekenis. Gebieden die niet relevant zijn om te waarderen vanuit een bepaald aspect krijgen de score 0 voor dit aspect. De score 100 betekent dat de gebieden altijd deze score behouden, ook na de integratie met andere beoordeelde aspecten. Aspecten die niet voor het hele gebied gelden, wanneer er bijvoorbeeld sprake is van punt of lijnobjecten, krijgen op de locaties zonder data de score 0. Dit zijn de zogenaamde niet gebiedsdekkende aspecten. Ook aspecten waar

onvoldoende data voor is, maar die eigenlijk wel voor het hele gebied zouden moeten gelden krijgen op de plaatsen zonder data de score 0. Scores kunnen met verschillende methodes toegekend worden. Deze methodes zijn, absoluut, in klassen, gradueel, logaritmisch of exponentieel.

2.3.2 Kwetsbaarheidbenadering

De impact van een ruimtelijke maatregel op het milieu kan op twee manieren beoordeeld worden. Vanuit de maatregel en vanuit het studiegebied. Wanneer vanuit de maatregel geredeneerd wordt, wordt een effectgerichte benadering gebruikt. Als vanuit het gebied geredeneerd wordt, kan de kwetsbaarheidbenadering gebruikt worden.

In veel gevallen is bij tracé- en locatiekeuzestudies voor het uitvoeren van de daadwerkelijke studies al sprake van een aantal alternatieven. In die gevallen wordt de effectgerichte benadering toegepast. Dit heeft als voordeel dat al vrij snel duidelijk is waarover gepraat wordt. Een groot nadeel is dat door vanaf het begin met slechts een aantal alternatieven te werken vele alternatieven, waaronder misschien wel het beste alternatief, helemaal niet bestudeerd worden (Kreft en Vanhaute, 1997).

Bij het gebruik van GISMER wordt geredeneerd vanuit gebiedskenmerken. Daarbij wordt gekeken naar de huidige geschiktheid en kwetsbaarheid van het studiegebied voor de te nemen ruimtelijke maatregel. Vanuit verschillende aspecten worden thema's opgebouwd. Voor ieder thema worden kwetsbaarheidkaarten gegenereerd. Deze thema kwetsbaarheidkaarten worden gecombineerd tot een integrale kwetsbaarheidkaart. Dit opbouwen van de thema en integrale kwetsbaarheidkaarten gebeurt door middel van een gewogen somming. De opbouw van de integrale kwetsbaarheidkaart is schematisch weergegeven in Fig. 2.10. Door gebruik te maken van de kwetsbaarheidbenadering wordt voor het gehele gebied de relatieve kwetsbaarheid bepaald voor een bepaalde ruimtelijke functie en worden op voorhand geen mogelijkheden uitgesloten. Ook kunnen relatief snel beleidswijzigingen doorberekend worden.

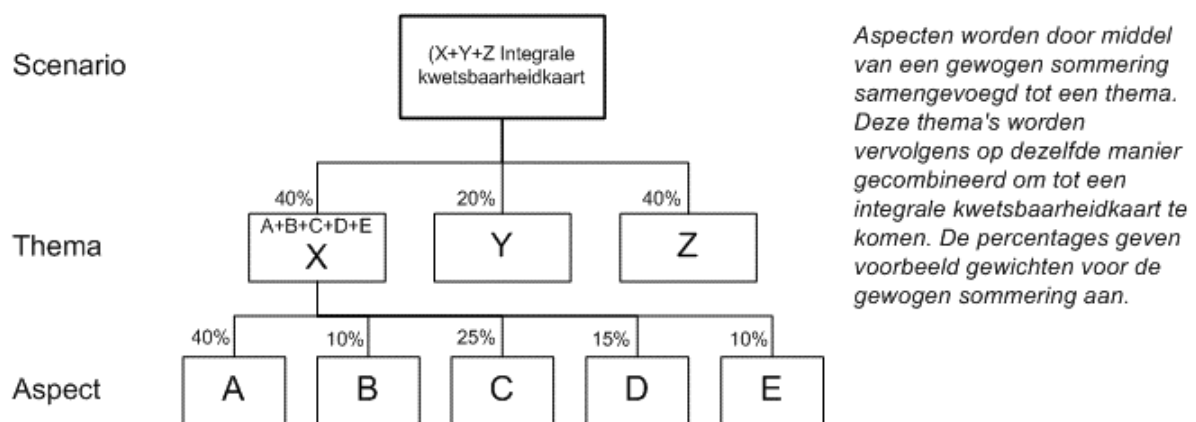


Fig. 2.10 Gewogen somming in de kwetsbaarheidbenadering

2.3.3 MCA-methoden

In de bestaande GISMER applicatie kan alleen gebruik gemaakt worden van de multicriteria analysemethode gewogen somming. Bij deze methode wordt aan ieder aspect een gewicht gekoppeld. Daarna worden de scores van het aspect vermenigvuldigd met het bijbehorende gewicht. Vervolgens worden de aspecten per thema opgeteld. Dezelfde stappen worden nog een keer herhaald, maar nu voor de thema's die gecombineerd worden tot een visie.

Aanvulling MCA-methoden

De functionaliteit van GISMER is door Grontmij verder bestudeerd waarbij onder andere gekeken is naar mogelijkheden om GISMER uit te breiden met een aantal MCA methoden.

Binnen GISMER kan alleen de gewogen somming MCA-methode worden toegepast, een nadeel van deze methode is dat er alleen kwantitatieve scores beoordeeld kunnen worden. Een methode waarin ook kwalitatieve scores meegewogen kunnen worden zou een toegevoegde waarde voor GISMER hebben. De verwachtingswaarde methode kan hiervoor gebruikt worden in de huidige GISMER applicatie. Deze methode maakt ook gebruik van een gewogen somming, maar heeft de mogelijkheid kwalitatieve waarden mee te nemen in de beoordeling. Gewichten worden berekend op basis van het aantal aspecten en op basis van de rangorde van de aspecten. De concordantie analyse, regime analyse en evamix analyse zijn niet geschikt om toegepast te worden in de huidige applicatie (Commandeur en Kreft, 1998).

3 Onderzoeksmethoden

In dit hoofdstuk worden de onderzoeksmethoden die binnen dit onderzoek gebruikt worden besproken. Dit is een verdere uitwerking van de onderzoeksopzet zoals deze in het onderzoeksvoorstel stond en die nu in bijlage 0 is opgenomen. De fasering van het onderzoek wijkt af van die in de figuur in bijlage I, de ‘survey’ is nu een op zichzelf staande fase geworden en er is een fase bijgekomen waarin de bestaande en de nieuwe applicatie met elkaar vergeleken worden. Dit om beter aan het doel van het onderzoek zoals dit in hoofdstuk 1 gesteld is te kunnen voldoen.

Dit onderzoek bestaat uit vier fasen. Ten eerste worden de bestaande GISMER applicatie en het ontwerp hiervan bestudeerd. Dit wordt gedaan door middel van een literatuurstudie en door te werken met de applicatie in een concreet project. De resultaten uit deze fase zijn beschreven in hoofdstuk 2. In de tweede fase wordt een ‘survey’ uitgevoerd naar de wensen en eisen die momenteel aan de functionaliteit en de gebruikersinterface worden gesteld. De opzet van deze ‘survey’ wordt in paragraaf 3.1 besproken en de resultaten in paragraaf 4.1.

De resultaten van de ‘survey’ worden gebruikt om een nieuw ontwerp voor GISMER te maken en een deel te implementeren. De werkwijze die hierbij gevolgd wordt, wordt beschreven in paragraaf 3.2. Het ontwerp en een beschrijving van de implementatie zijn opgenomen in paragraaf 4.2. In de laatste fase worden componenten van de bestaande applicatie vergeleken met componenten van de nieuwe applicatie. Hierbij wordt aandacht geschonken aan drie aspecten. Deze aspecten zijn de functionaliteit en de gebruikersinterface, het databaseer en ten derde de programmeeromgeving. Hoe deze vergelijking uitgevoerd gaat worden wordt beschreven in paragraaf 3.3. De resultaten van de vergelijking worden in paragraaf 4.3 beschreven.

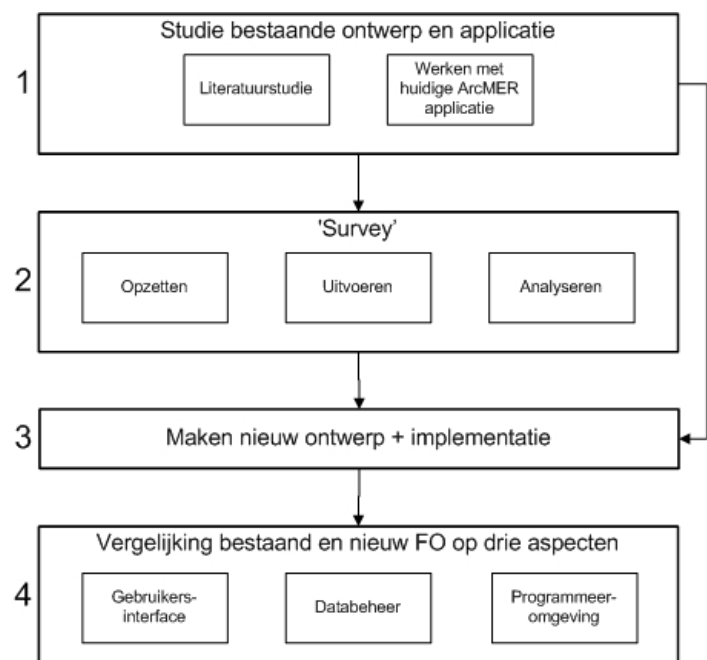


Fig. 3.1 Onderzoeksopzet

In Fig. 3.1 worden de verschillende fasen van dit onderzoek nog eens overzichtelijk weergegeven. De nummers komen overeen met de fasen in Tabel 3.1. In deze tabel is te vinden waar, in dit rapport de methoden en resultaten per fase staan.

Tabel 3.1 Onderzoeksopzet

Fase	Omschrijving	Methode	Resultaten
1	Studie bestaande ontwerp en applicatie	Hoofdstuk 3	Hoofdstuk 2
2	‘Survey’ naar wensen m.b.t. functionaliteit en gebruikersinterface	Paragraaf 3.1	Paragraaf 4.1
3	Maken nieuw ontwerp	Paragraaf 3.2	Paragraaf 4.2
4	Vergelijking bestaande met nieuwe applicatie	Paragraaf 3.3	Hoofdstuk 5

3.1 ‘Survey’ naar functionele eisen en wensen ten aanzien van GISMER

Om inzicht te verschaffen in de wensen en eisen die gesteld worden aan de functionaliteit en gebruikersinterface van GISMER wordt een ‘survey’ opgezet en uitgevoerd. Deze ‘survey’ wordt gehouden onder experts uit het toepassingsgebied. De informatie die dit oplevert wordt ten eerste gebruikt om een vergelijking te maken tussen functionaliteit en de gebruikersinterface van de huidige applicatie en de wensen die hieraan gesteld worden voor de nieuwe applicatie. Vervolgens worden de bevindingen uit de ‘survey’ gebruikt bij het opzetten van een nieuw functioneel ontwerp.

In dit hoofdstuk worden de opzet en de uitvoering van de ‘survey’ beschreven. De resultaten komen in het volgende hoofdstuk. Een ‘survey’ bestaat vanaf de voorbereiding tot en met de afronding uit zeven stappen. Deze zijn in Fig. 3.2 weergegeven.

De eerste vier stappen vinden plaats tijdens het opzetten van de ‘survey’ en worden in subparagraaf 3.1.1 besproken. De vijfde en zesde stap vormen het uitvoeren van de ‘survey’ en komen aan de orde in subparagraaf 3.1.2. De laatste stap, het analyseren van de resultaten in subparagraaf 3.1.3.

1. Vaststellen van het doel van de ‘survey’
2. Bepalen van de doelgroep
3. Kiezen van de interviewmethode
4. Opstellen van de vragenlijst
5. Testen van de vragenlijst
6. Uitvoeren van de ‘survey’
7. Analyseren van de resultaten

Fig. 3.2 Stappen in een ‘survey’ (Bron: Creative Research Systems, 2003)

3.1.1 Opzetten van de ‘survey’

De eerste stap in de ‘survey’ opzet is het bepalen van het doel. Om dit te kunnen bepalen moet er worden gekeken naar de probleem- en doelstelling van dit onderzoek zoals ze in het eerste hoofdstuk beschreven staan. De drie stappen die daarna komen hebben een sterke onderlinge wisselwerking en kunnen ook niet afzonderlijk van elkaar uitgevoerd worden. Er moet antwoord gegeven worden op de vragen wie de doelgroep is, wat er gevraagd gaat worden en welke methode gebruikt gaat worden om de ‘survey’ uit te voeren. Deze vragen hebben ieder voor zich weer invloed op de andere vragen. De grootte van de groep contactpersonen is bijvoorbeeld bepalend voor de te gebruiken methode. Wat er gevraagd moet worden, bepaald vervolgens weer welke personen benaderd moeten worden. De antwoorden op deze vragen zijn voor een deel af te leiden uit het doel van de ‘survey’ en voor een groot deel uit de eerste twee hoofdstukken.

Doel van de ‘survey’

De ‘survey’ die binnen dit onderzoek uitgevoerd wordt, wordt gebruikt om de veranderingen tussen 1993 en 2003/2004 met betrekking tot de eisen en wensen die aan de functionaliteit en de gebruikersinterface van GISMER worden gesteld aan het licht te brengen. De nieuwe functionele wensen en eisen gaan daarnaast ook als basis dienen voor het functioneel ontwerp van de nieuwe GISMER applicatie. Het doel van deze ‘survey’ is dan ook:

Het krijgen van inzicht in eisen en wensen die aan de functionaliteit en gebruikersinterface van de nieuwe GISMER applicatie worden gesteld.

Wie is de doelgroep?

De resultaten van de ‘survey’ moeten inzicht geven in de wensen die gesteld worden aan de functionaliteit en de gebruikersinterface voor de nieuwe GISMER applicatie. Daarnaast dienen de resultaten onder andere als basis voor het functionele ontwerp van de nieuwe applicatie. Het onderzoek wordt uitgevoerd onder potentiële toekomstige gebruikers en opdrachtgevers van de applicatie. Hierbij gaat het dus om mensen met kennis van milieu effect rapportages en locatiekeuze studies. Daarnaast worden een aantal GIS gebruikers en ontwikkelaars aan de groep contactpersonen toegevoegd. Dit om meer inzicht te krijgen in de GIS-technische wensen die aan de nieuwe applicatie worden gesteld.

Opdrachtgevers voor het werken met GISMER zijn binnen en buiten Grontmij te vinden. Binnen Grontmij gaat het meestal om adviseurs die gespecialiseerd zijn op Milieu Effect Rapportage of locatiekeuzestudies die een opdracht gedeeltelijk uitbesteden. Externe opdrachtgevers zijn vaak overheden zoals provincies of een aantal gemeenten in een samenwerkingsverband. Huidige gebruikers van GISMER zijn alleen GIS specialisten van Grontmij Geo Informatie. Potentiële gebruikers van de nieuwe applicatie zijn er ook daarbuiten. Hierbij valt te denken aan adviseurs voor ruimtelijke ordening werkzaam bij advies- of ingenieursbureaus, provincies of grotere gemeenten. In bijlage 0 is een overzicht opgenomen van de contactpersonen die meewerken aan de ‘survey’. In dit overzicht is ook te vinden wat hun functies en expertises zijn.

Wat wordt gevraagd?

Voor de vragen met betrekking tot de functionele eisen en wensen die gesteld worden aan GISMER, wordt gebruik gemaakt van hoofdstuk 2. De vragen hebben betrekking op alle aspecten van de functionaliteit van GISMER. Eerst wordt de toegepaste kwetsbaarheidbenadering ter discussie gesteld. Vervolgens wordt de werking van de applicatie in dezelfde volgorde als de componentbeschrijving uit hoofdstuk twee behandeld. Daarna wordt bekeken of het wenselijk is de applicatie met een aantal analyse methoden uit te breiden. Met betrekking het gebruik van de applicatie door de contactpersonen zelf, worden ook een aantal vragen opgenomen. Deze gaan eerst in op de ArcGIS kennis van de gebruiker en de behoefte aan een GISMER applicatie binnen de organisatie waar de contactpersoon voor werkt. Vervolgens wordt ingegaan op de wensen ten aanzien van de gebruikersinterface.

Welke methode wordt gebruikt om de ‘survey’ uit te voeren?

Het antwoord op de vraag hoe de ‘survey’ uitgevoerd moet worden is grotendeels afhankelijk van het antwoord op de vorige twee vragen. Er zijn namelijk een groot aantal methoden die ieder hun eigen voor- en nadelen hebben (www.surveysystem.com, 2003). Voor de ‘survey’ die in dit onderzoek wordt uitgevoerd, worden persoonlijke interviews gehouden. De contactpersonen krijgen ongeveer een week van te voren per e-mail een vragenlijst met achtergrondinformatie ter voorbereiding toegestuurd. In bijlage 0 is de vragenlijst met achtergrondinformatie opgenomen die de contactpersonen via e-mail toegestuurd hebben gekregen.

3.1.2 Uitvoeren van de ‘survey’

Testen van de ‘survey’

Voordat de interviews gehouden worden onder de contactpersonen moet de vragenlijst eerst getest worden. Hiermee worden de tijdsduur van een interview en de helderheid van de vragenlijst vooraf onderzocht. De ‘survey’ uit dit onderzoek wordt vooraf getest bij Ilya Musters, oud landgebruiksplanning student aan Wageningen Universiteit en werkzaam bij Tauw in Deventer. De resultaten van de test worden meegenomen bij de daadwerkelijke uitvoering van de ‘survey’.

Uitvoeren van de ‘survey’

De vragenlijst inclusief de achtergrondinformatie die in bijlage 0 is opgenomen, wordt ongeveer een week voor de persoonlijke interviews ter voorbereiding naar de contactpersonen gestuurd. De interviews worden vrijwel allemaal gehouden binnen het tijdsbestek van een week. De data waarop de interviews worden gehouden zijn per contactpersoon opgenomen in bijlage 0.

Tijdens de interviews wordt de vragenlijst per deel doorgenomen en niet per vraag. Door het inspelen op de antwoorden van de contactpersonen kan het gesprek een andere wending nemen dan in de richting van de volgende vraag op de vragenlijst. Juist daardoor kan het wel veel nuttige informatie opleveren. Wel moeten alle vragen van de vragenlijst bij ieder interview behandeld worden.

3.1.3 Analyseren resultaten van de ‘survey’

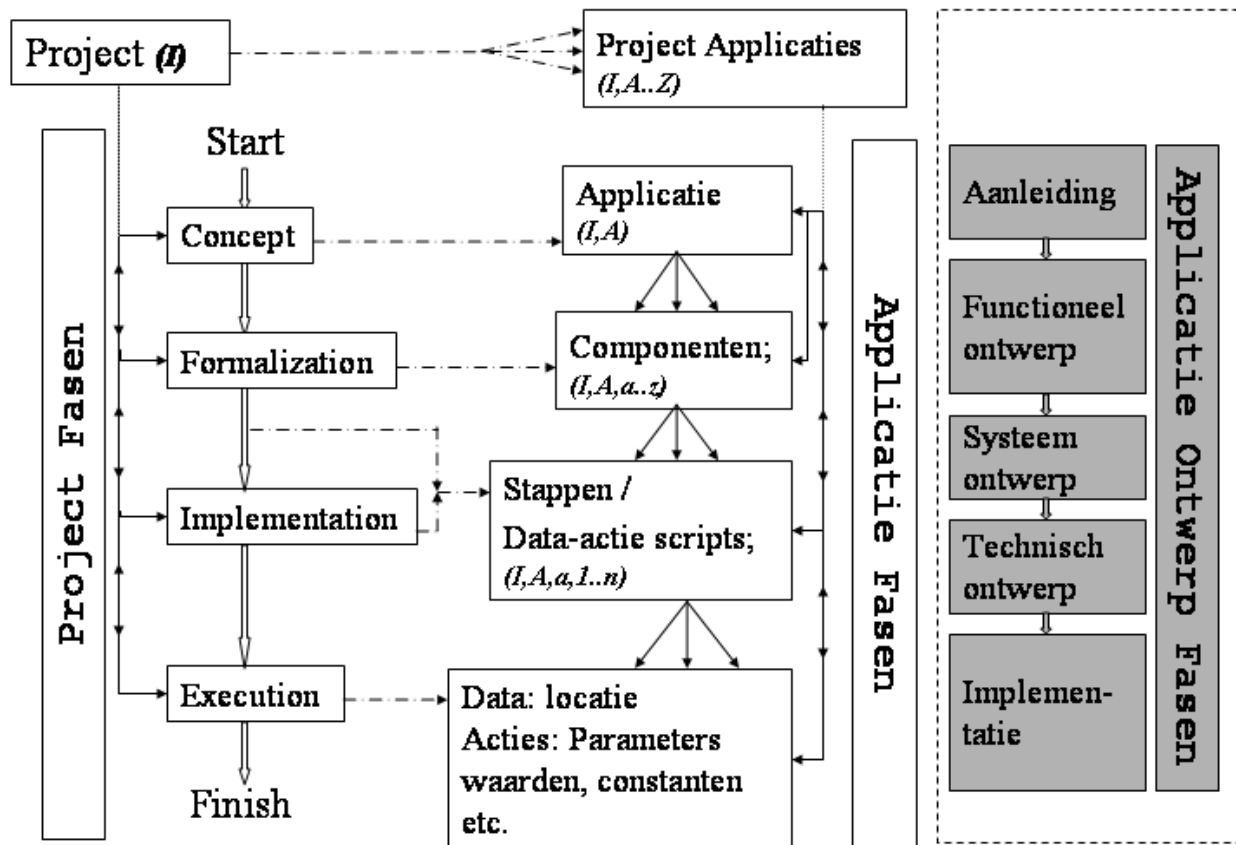
Het analyseren van de antwoorden van de ‘survey’ wordt per deel van de vragenlijst uitgevoerd.

Hierbij wordt gekeken welke antwoorden door de contactpersonen zijn gegeven en welke mening de contactpersonen hebben over de punten die tijdens de interviews onder de aandacht zijn gebracht.

Belangrijk zijn niet de individuele antwoorden of meningen van de contactpersonen, maar juist de antwoorden en meningen die door de contactpersonen gezamenlijk worden gedeeld. In paragraaf 4.1 wordt per deel van de vragenlijst een overzicht gegeven van de wensen en eisen die aan de nieuwe GISMER applicatie worden gesteld.

3.2 Het maken van een ontwerp

Binnen een applicatie zijn een aantal fases te onderscheiden. In hoofdlijnen houdt deze applicatie fasering in dat een applicatie opgebouwd is uit een aantal componenten. Een component is een min of meer op zichzelf staand deel van een applicatie. De losse componenten hebben relaties met elkaar waarbij meestal de uitvoer van een component de invoer is voor een andere. Componenten worden ten opzichte van elkaar onderscheiden door de hoeveelheid interactie met de gebruiker (Stuiver 2003, 2004). De componenten kunnen vervolgens uit een of meerdere stappen bestaan. Binnen een stap kunnen een of meerdere verschillende acties voorkomen. Uiteindelijk worden voor het uitvoeren van de stappen en acties, de benodigde data, parameters, constanten etc. opgegeven. Deze fases komen overeen met een aantal projectfases. Deze overeenkomsten zijn door Stuiver onderscheiden en zijn in Fig. 3.4 weergegeven in de twee linker kolommen.



Project I, bevat de applicaties A t/m Z. Applicatie A bestaat uit de componenten a t/m z en component a bestaat vervolgens uit de stappen 1 t/m n.

Fig. 3.4 Project-, Applicatie- en Applicatie ontwerp fasering (Vrij naar: www.geo-informatie.nl)

In de conceptfase wordt besloten wat de aanleiding en het doel voor een applicatie zijn. In dit onderzoek wordt deze conceptfase gevormd door de inleiding, de analyse van de huidige GISMER applicatie en de 'survey'. Wanneer de conceptfase afgerond is, moet de applicatie ontworpen worden, dit gebeurt in de projectfasen formalisatie en implementatie.

Om tot een praktische invulling van deze projectfasen te komen zijn aan de figuur van Stuiver een aantal applicatie ontwerp fasen toegevoegd. Aan deze applicatie ontwerp fasen wordt in dit onderzoek vorm gegeven door de fontein metafoor (de Champeaux *et al.* 1993 en Booch, 1994). Dit is een variant van de traditionele watervalmethode (de Champeaux *et al.* 1993) van software ontwikkeling. Bij het gebruik van deze methode wordt eerst een functioneel ontwerp, vervolgens een systeem ontwerp en daarna een technisch ontwerp gemaakt. Tenslotte wordt de applicatie geïmplementeerd. Deze opeenvolging van ontwerpen wordt nauwkeurig beschreven door Rumbahugh *et al.* 1991 en in bijlage 0. De drie ontwerpen volgen elkaar niet statisch op, maar er is sprake van een iteratief proces.

Hierdoor kunnen veranderingen die in een later stadium van ontwikkeling aan het licht komen nog wel meegenomen worden. Er is niet gekozen voor een volledig dynamische methode als Dynamic Systems Development Method (DSDM) (DSDM Ltd., 2003). De fontein metafoor is namelijk wel iteratief, het is echter niet nodig om tot een snelle ontwikkeling van een prototype te komen. Een toelichting van de verschillende methoden van applicatie ontwikkeling wordt gegeven in bijlage 0.

In de projectfase formalisatie wordt het functioneel ontwerp opgesteld. Daarin wordt bepaald uit welke componenten een applicatie moet bestaan en welke functionaliteit deze componenten moeten bezitten. Tijdens de projectfase implementatie worden het systeem ontwerp en het technisch ontwerp gemaakt. De implementatie van het technisch ontwerp vindt plaats in de projectfasen implementatie en uitvoering. Hieronder volgt een meer gedetailleerde beschrijving van de verschillende fasen en hun onderlinge relaties zoals ze in Fig. 3.4 te vinden zijn.

3.2.1 Functioneel ontwerp

[functioneel ontwerp](#)

GISMER is een applicatie die deel uit kan maken van een MER- of locatiekeuzeproject en valt in het schema van Fig. 3.4 dus bij de applicatie fasen in het kader Applicatie. Op dit niveau wordt eerst gekeken welke functionaliteit de applicatie moet bevatten. Als bron worden hier de resultaten van de ‘survey’ gebruikt. De gewenste functionaliteit wordt indien mogelijk opgenomen in het functioneel ontwerp. Wanneer functionaliteit die expliciet uit de ‘survey’ als wens naar voren is gekomen wordt opgenomen in het ontwerp, wordt dit vermeld in de tekst. Vervolgens wordt de functionaliteit opgedeeld in een aantal losse componenten. Na het onderscheiden van de verschillende componenten worden per component de relevante objectklassen en hun onderlinge relaties onderscheiden. De relevante objectklassen zijn bijvoorbeeld de geografische datasets die invoer, tussentijds resultaat of uitvoer van de applicatie zijn. Het kunnen ook tabellen zijn waarin alle keuzes die gemaakt worden bij het gebruik van de applicatie op worden geslagen. Bij het ontwerpen van de applicatie moet rekening gehouden worden met deze tabellen en indien nodig moeten daarvoor extra objectklassen en relaties worden toegevoegd.

Wanneer de objectklassen en relaties bekend zijn worden de attributen en mogelijke attribuutwaarden hierbij bepaald. Bij het bepalen van de relevante objectklassen, relaties en attributen worden meteen de interacties met gebruikers en de datastromen binnen de componenten benoemd. Dit om een overzichtelijk beeld te geven van het functioneel ontwerp als geheel. Er wordt dus geen onderscheid gemaakt tussen een object model, dynamisch model en een functioneel model zoals dat wel wordt gedaan bij de methode van Rumbaugh *et al.*

Als al deze stappen doorlopen zijn, is de formalisatie projectfase uit Fig. 3.4 doorlopen. Het functioneel ontwerp wordt besproken in 4.2.1.

[systeem ontwerp](#)

3.2.2 Systeem ontwerp

Bij het systeem ontwerp wordt een keuze gemaakt over het databaseer binnen de applicatie en de besturing van de applicatie. In het systeem ontwerp wordt de manier hoe de data op moet worden geslagen verder toegelicht. Bij het systeem ontwerp wordt daarnaast aandacht besteedt aan de besturing van de applicatie. De resultaten van de ‘survey’ die betrekking hebben op het databaseer en de gebruikersinterface en het functioneel ontwerp worden als uitgangspunt genomen. De keuze van de besturingsmethode van de applicatie hangt af van de functionaliteit. Wanneer een bepaalde stap bijvoorbeeld meerdere malen doorlopen moet worden, vergt dit een andere besturing dan wanneer een stap slechts een keer uitgevoerd moet worden. Het is niet noodzakelijk dat de gehele applicatie op dezelfde manier bestuurd wordt, dit kan per component verschillen. In subparagraaf 4.2.2 is het systeem ontwerp voor de nieuwe applicatie opgenomen.

[technisch ontwerp](#)

3.2.3 Technisch ontwerp

Voor de eerste drie componenten van GISMER wordt een technisch ontwerp gemaakt. In dit technisch ontwerp worden de stappen binnen een component onderscheiden. Binnen de stappen worden de verschillende acties onderscheiden. Voor iedere component worden de bijbehorende data-actie modellen opgesteld. Hierbij wordt niet alleen aandacht geschonken aan de invoer, acties en uitvoer van de componenten, maar ook aan de procesbewaking. Het technisch ontwerp voor de eerste drie componenten is te vinden in subparagraaf 4.2.3.

[implementatie](#)

3.2.4 Implementatie

De laatste applicatie ontwerp fase is de implementatiefase. In deze fase moet eerst de omgeving gekozen worden waarin de applicatie ontwikkeld wordt. Daarna moeten de scripts geschreven worden en moeten de datalocatie, parameters, constanten enz. gedefinieerd worden. De basis voor de implementatie wordt gevormd door het technisch ontwerp. Een beschrijving van de implementatie wordt gegeven in subparagraaf 4.2.4.

3.3 Vergelijking bestaande met nieuwe applicatie

Om inzicht te krijgen in de impact van de Geodatabase en het object georiënteerd programmeren op de migratie van de applicatie GISMER naar ArcGIS worden de bestaande en de nieuwe GISMER applicatie met elkaar vergeleken. Deze vergelijking vindt plaats ten aanzien van de functionaliteit en de gebruikersinterface, het databeheer en de programmeeromgeving. Hieronder wordt per aspect toegelicht hoe de vergelijking plaats gaat vinden.

3.3.1 Vergelijking functionaliteit en gebruikersinterface

Eerst wordt vergeleken welke functionaliteit en vervolgens hoe de gebruikersinterface veranderd is. Om de veranderingen in functionaliteit te achterhalen worden de functionele ontwerpen van de bestaande en de nieuwe applicatie met elkaar vergeleken. Hierbij wordt de functionaliteit per component van de nieuwe applicatie afgezet tegen de functionaliteit van de bestaande applicatie. Het systeem ontwerp van de nieuwe applicatie wordt vergeleken met de huidige gebruikersinterface. Een belangrijk aandachtspunt daarbij is de opbouw van de applicatie. De resultaten zijn in tabelvorm opgenomen in paragraaf 5.1.

3.3.2 Vergelijking databeheer

Om de veranderingen ten aanzien van het databeheer tussen de bestaande en de nieuwe applicatie in beeld te brengen wordt aandacht besteedt aan welke gegevens en hoe deze gegevens opgeslagen moeten worden. Belangrijke aandachtspunten zijn de dataformaten en de opslagstructuren die gebruikt worden bij de applicaties. Achtergronden over het databeheer in ArcInfo en ArcGIS zijn opgenomen in bijlage VI. Daarnaast wordt aandacht besteed aan procesbewaking. Getoetst wordt of de keuzes die gemaakt worden tijdens het gebruik van GISMER vastgelegd worden en of deze reproduceerbaar zijn. De resultaten zijn in tabelvorm opgenomen in paragraaf 5.2.

3.3.3 Vergelijking programmeeromgeving

Bij het vergelijken van de programmeeromgeving wordt voor de nieuwe applicatie gekeken naar het technisch ontwerp en de implementatie. Belangrijke aandachtspunten bij het vergelijken van de programmeeromgeving zijn:

- de beschikbaarheid en keuzevrijheid uit ontwikkelomgevingen
- de gekozen ontwikkelomgevingen
- de mogelijkheid tot het hergebruik van delen applicatiecode
- de mogelijkheid standaardoplossingen toe te passen zonder zelf workarounds (WhatIs.com, 2004) te programmeren
- de mogelijkheid om de applicatie te updaten of wijzigen

Ook de resultaten van deze vergelijking zijn in tabelvorm opgenomen, ze staan in paragraaf 5.3.

4 Resultaten ‘survey’ en applicatieontwikkeling

In dit hoofdstuk worden eerst in paragraaf 4.1 de resultaten van de ‘survey’ besproken, daarna worden in paragraaf 4.2 de resultaten van de applicatieontwikkeling besproken.

4.1 Resultaten ‘survey’

De beschrijving van de resultaten van de ‘survey’ is opgedeeld in zeven onderwerpen, deze onderwerpen komen overeen met de delen waaruit de vragenlijst uit bijlage 0 is opgebouwd.

1. Kwetsbaarheidbenadering
2. Werking GISMER versie 1993; voorbereiding en voorbewerking
3. Werking GISMER versie 1993; bewerking en analyse
4. Beoordelingsmethode
5. MCA-methoden
6. ArcGIS
7. Overig

4.1.1 Kwetsbaarheidbenadering

De kwetsbaarheidbenadering is vooral goed te gebruiken bij een locatie-MER, of bij een inrichtings-MER indien er sprake is van een groot studiegebied. Voor een locatiekeuzestudie is de kwetsbaarheidbenadering ook goed te gebruiken bij het maken van een keuze uit verschillende alternatieven. Bij tracéstudies is dit minder het geval. Hierbij kan de kwetsbaarheidbenadering alleen gebruikt worden om globale voorlopige tracé aan te duiden en beter niet voor het kiezen van een uiteindelijk tracé.

GISMER kan alleen als ondersteunend gereedschap gebruikt worden, zowel bij milieu effect rapportages als bij locatiekeuzestudies. Met de kwetsbaarheidbenadering wordt alleen gekeken naar de minst of meest kwetsbare gebieden. Er wordt niet aangegeven hoe kwetsbaar het minst kwetsbare gebied dan nog is, daarvoor blijft altijd nog een effectgerichte studie nodig.

De effectgerichte benadering kan het beste gebruikt worden bij een inrichtings-MER en in de afsluitende fase van een MER-traject. Dit is dus meestal het geval wanneer al een voorkeurstracé of een voorkeurslocatie, eventueel op basis van de kwetsbaarheidbenadering, gekozen is.

Vaak is de voorkeur voor een te hanteren benadering afhankelijk van de opdrachtgever. Een overheid moet vaak de grond nog verwerven en kan daardoor op zoek gaan naar de meest geschikte locatie. Een particuliere opdrachtgever heeft die vrijheid meestal niet en is daardoor vaak gebonden aan de locatie waarvan hij de grond al in bezit heeft. In het eerste geval kan dus het beste de kwetsbaarheidbenadering toegepast worden om tot een voorkeurslocatie te komen. In het tweede geval kan beter de effectgerichte benadering toegepast worden. Effecten van een ingreep bepalen is zeer divers en er zijn vele modellen voor nodig om deze te berekenen. Daarom is het moeilijk om dit volledig te automatiseren.

4.1.2 Werking GISMER versie 1993; voorbereiding en voorbewerking

Standaardisatie in de voorbereidingsfase van GISMER is enerzijds niet gewenst en anderzijds wel. Volledige standaardisatie vanuit het oogpunt welke criteria meegenomen en beoordeeld moeten worden is niet mogelijk. Iedere MER of locatiekeuzestudie heeft namelijk eigen aspecten die alleen spelen in het studiegebied. Aangeraden wordt wel om een groot aantal locatiekeuzestudies en milieu effect rapportages te bestuderen om tot een lijst met veel voorkomende criteria te komen. Voor deze

criteria is het dan wel gewenst om tot een gestandaardiseerde technische voorbereiding van de data te komen. Deze standaardprocedures moeten dan op worden genomen in de nieuwe GISMER applicatie. Vanuit de gebruikers is in ieder geval een standaardprocedure opgemerkt die zeker opgenomen moet worden. Dit is het voorbereiden van vector- of rasterdata door middel van een buffering binnen een rasterformaat. Het is daarmee mogelijk om een raster te maken met daarin de afstanden tot een ruimtelijk object, bijvoorbeeld een ontsluitingspunt tot een snelweg.

De standaard datastructuur die gehanteerd moet worden in GISMER is GRID (rasterdata in ArcInfo formaat). Met deze datastructuur is het makkelijker MCA berekeningen toepassen dan in een vectordata georiënteerde omgeving. Ook de einduitvoer van de applicatie is in GRID.

Schijnnaauwkeurigheid die zou kunnen ontstaan bij het terugconverteren naar een vectorstructuur wordt voorkomen. Als invoer moeten in ieder geval de meest gangbare dataformaten gebruikt kunnen worden, waarbij de ESRI-shapefile de voorkeur heeft. De invoer van data moet eenmalig plaatsvinden en niet twee keer zoals in de huidige applicatie het geval is. Voor Grontmij is het belangrijk dat de herkomst van en gegevens over basisdata nauwkeurig gedocumenteerd kan worden. Daarnaast moet de data, zowel invoer als (tussen)resultaten, overzichtelijk beheerd kunnen worden.

4.1.3 Werking GISMER versie 1993; bewerking en analyse

Een deel van de ondervraagden geeft de voorkeur aan het vooraf inbrengen van harde randvoorwaarden en een deel geeft de voorkeur aan het achteraf inbrengen. Vooraf heeft als voordeel dat er meteen duidelijkheid gecreëerd wordt in een projectteam en richting burgers. In de praktijk blijkt dat de harde randvoorwaarden regelmatig aangepast worden gedurende het locatiekeuze- of MER-project. Door het achteraf inbrengen van de harde randvoorwaarden worden onverwachte alternatieven niet bij voorbaat uitgesloten. Ook kan het uitsluiten van bepaalde locaties dan juist gerechtvaardigd worden.

Over het algemeen is zo veel mogelijk flexibiliteit gewenst zodat de opdrachtgever zelf kan beslissen of en zo ja, wanneer harde randvoorwaarden ingebracht moeten worden.

De opbouw van een integrale kwetsbaarheidkaart via drie niveaus is logisch en inzichtelijk. Dit vinden de contactpersonen belangrijk. Een belangrijk niveau is dat van de thema's. De thema kwetsbaarheidkaarten worden vaak gebruikt voor presentatiedoeleinden. Een aantal contactpersonen heeft aangegeven dat het wenselijk is om een vierde niveau op te nemen. Deze kan dan gebruikt worden wanneer het nodig is. Dit vierde niveau moet dan onder het aspect niveau komen te staan zodat een aspect uit meerdere criteria opgebouwd kan worden. Dit vierde niveau wordt door de huidige gebruikers niet als wenselijk gezien. Andere opdrachtgevers geven aan dat bij een vierde niveau de applicatie minder doorzichtig wordt.

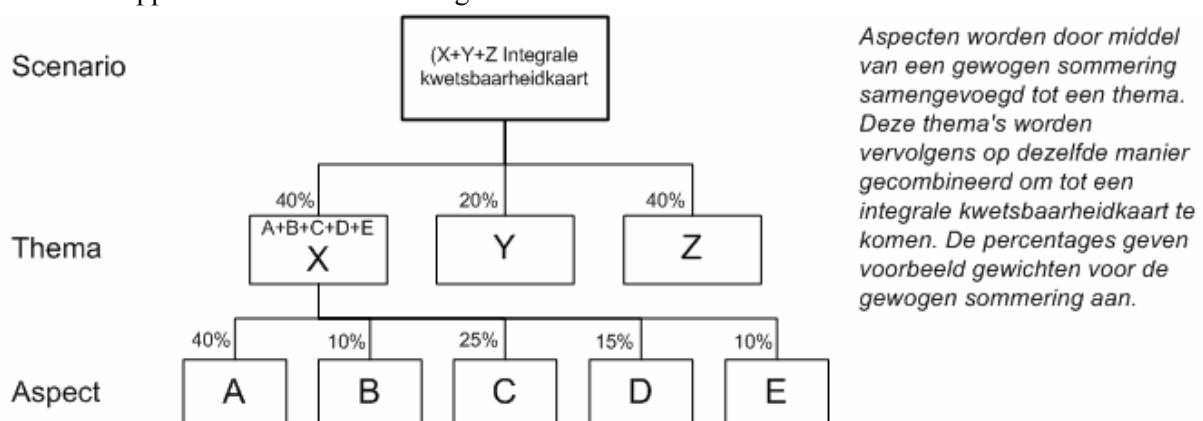


Fig. 4.10 Opbouw integrale kwetsbaarheidkaart volgens drie niveaus

De wegingsfactoren om tot een thema te komen moeten bepaald worden door inhoudelijke deskundigen en de wegingsfactoren om tot een visie te komen door bestuurders.

De term visie is geen goede term voor het eindproduct van GISMER. Ook de term scenario is niet goed. Een scenario bestaat namelijk uit ontwikkelingen, bijvoorbeeld economisch of demografisch, die van invloed zijn op de locatiekeuze of milieu effect rapportage. De term die wel gebruikt kan worden is integrale kwetsbaarheidkaart of integraal kwetsbaarheidraster. De naam ‘integrale kwetsbaarheidkaart’ wordt gebruikt wanneer over analoge producten wordt gesproken en ‘integraal kwetsbaarheidraster’ wanneer het over digitale producten gaat.

Alle keuzes die gemaakt worden tijdens het werken met GISMER zoals het toekennen van gewichten of het toepassen van een samengestelde harde randvoorwaarde moeten opgeslagen worden en reproduceerbaar zijn. Procesbewaking binnen de applicatie is voor zowel opdrachtgevers als gebruikers belangrijk en mag daarom niet ontbreken.

4.1.4 Beoordelingsmethode

De scores 0 en 100 moeten dezelfde status krijgen als de scores 1 t/m 99. Wanneer iets niet relevant is voor een bepaald aspect, is het ook niet kwetsbaar. Het huidige gebruik van de score 0 is dus niet rechtvaardig. Het gebruik van 100 als absolute score geeft aanleiding tot onjuistheden bij het samenvoegen van aspecten tot thema's. De score 100 moet dus dezelfde status krijgen als 1 t/m 99. Wanneer de scores 0 en 100 beide deze status krijgen, zijn er in totaal 101 verschillende mogelijkheden. Dit moet teruggebracht worden naar 100.

Niet gebiedsdekkende aspecten en aspecten met onvolledige data moeten wel mee worden genomen in de beoordeling. De contactpersonen hebben geen oplossing voor de omgang met niet gebiedsdekkende aspecten. De beoordeling moet uitgevoerd worden door inhoudelijke deskundigen. Voor aspecten met onvolledige data moet er een vakinhoudelijke inschatting gemaakt worden waarbij aannames gedaan worden op basis van deskundigheid.

Alle beoordelingsmethoden die in de bestaand applicatie bestaan, moeten toepasbaar blijven. De meeste aandacht moet uitgaan naar absoluut scores, scores in klassen en gradueel scores.

4.1.5 MCA-methoden

Het belangrijkste bij het kiezen van een methode is dat deze simpel en doorzichtig is voor gebruiker en burger. De analysemethode zoals deze in de applicatie gebruikt wordt, moet uit te leggen zijn aan opdrachtgevers en burgers. Door Carsjens wordt aangegeven dat de gewogen sommering de enige redelijk uit te leggen multi-criteria analyse methode is. Hij geeft ook aan dat het gebruik van meerdere methoden meestal geen andere resultaten geeft. Uitbreiding met extra MCA-methoden is vanuit deze twee invalshoeken dus niet gewenst.

Een effectgerichte benadering gaat uit van kwalitatieve aspecten. Indien een effectgerichte benadering wordt gebruikt bij MER of locatiekeuzestudies, moeten deze ook meegenomen worden. Het automatiseren van een effectgerichte benadering wordt echter niet als haalbaar gezien.

4.1.6 ArcGIS

Bij opdrachtgevers en uitvoerders van locatiekeuzestudies en milieu effect rapportages is er nauwelijks ArcGIS kennis aanwezig. De behoefte aan kennis is er wel, maar dan voornamelijk voor het visualiseren van hun producten. In het geval van GISMER zouden ze zelf wel de gegevensvoorbereiding willen doen, maar daar hebben ze de kennis niet voor. Het werken met applicaties zoals GISMER wordt overgelaten aan geo-informatiekundige bedrijven. Belangstelling voor het gebruik van GISMER in het onderwijs is er wel vanuit het laboratorium voor ruimtelijke planvorming van Wageningen Universiteit. Hier wordt echter ook nog geen ArcGIS gebruikt, de behoefte daaraan is er wel, maar de bereidheid om te migreren nog niet.

Wensen met betrekking tot de gebruikersinterface werden door de meeste contactpersonen niet gegeven aangezien ze het gebruik van een applicatie zoals GISMER uitbesteden. Vanuit de huidige gebruikers worden een aantal punten ter verbetering van de gebruikersinterface aangedragen. Data moet momenteel twee keer ingevoerd worden, eerst door het verrasteren en daarna moet het verrasterde basisbestand nog als aspect gedefinieerd worden. Dit moet in een keer gedaan kunnen worden. Bij het beoordelen van aspecten is niet te zien welke waarden allemaal al gescoord zijn en welke niet. Achtergrondinformatie bij aspecten, thema's of visies die opgeslagen moeten worden, moet in twee stappen gegeven worden. De mogelijkheid van documentatie van de bestanden moet verbeterd worden, zowel functioneel als in de gebruikersinterface. In algemene zin is de gebruikersinterface niet gebruiksvriendelijk en heeft deze een sterk verouderd uiterlijk. Vanuit Grontmij is het gewenst de mogelijkheid open te houden dat de applicatie verkocht kan worden. Daarvoor moet de gebruikersinterface wel veel eenvoudiger en overzichtelijker worden dan die van de bestaande applicatie. De wens is om deze te integreren met die van ArcGIS als een nieuwe extensie.

4.1.7 Overig

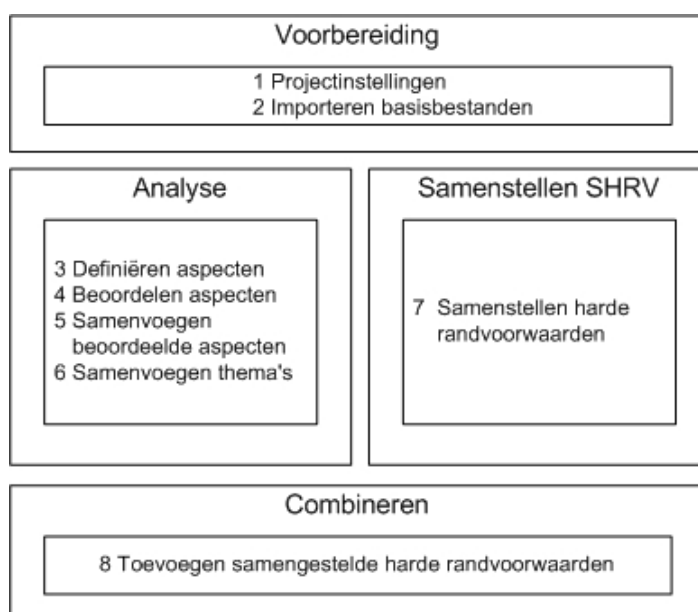
Bij alle studies met een ruimtelijke component waarbij geschiktheden, kwetsbaarheden of kansen bepaald moeten worden kan GISMER ingezet worden. GISMER moet niet alleen gebruikt worden om kwetsbaarheden (negatief) of geschiktheden (neutraal) in kaart te brengen. Ook de doorslag naar de positieve kant, waarbij gekeken wordt naar kansrijke gebieden, moet gemaakt worden.

4.2 Een ontwerp voor de nieuwe applicatie

In deze paragraaf wordt het ontwerp van de nieuwe applicatie beschreven. Hierbij wordt de methode gehanteerd zoals deze beschreven staat in 3.2. Eerst wordt in subparagraaf 4.2.1 het functioneel ontwerp besproken, daarna wordt in subparagraaf 4.2.2 het systeem ontwerp besproken en vervolgens in subparagraaf 4.2.3 het technisch ontwerp. In subparagraaf 4.2.4 wordt uiteindelijk de implementatie besproken. Bij het opstellen van de ontwerpen en bij de implementatie is gewerkt zoals beschreven in 3.2.

[werkwijze](#)

4.2.1 Functioneel ontwerp



Eerst is de applicatie op basis van de informatie uit hoofdstuk 2 en uit paragraaf 4.1 opgedeeld in een achttal componenten. Deze acht componenten zijn onder te verdelen in lagen en partities. Lagen verdelen een systeem horizontaal in stukken en partities verticaal. De lagen en partities hebben een cliënt-server relatie, net zoals de componenten binnen de eerste partitie, de analyse. De server verleent hierbij een dienst voor de client (Watson, 1999). In Fig. 4.2 zijn de lagen en partities en de componenten waaruit ze bestaan weergegeven.

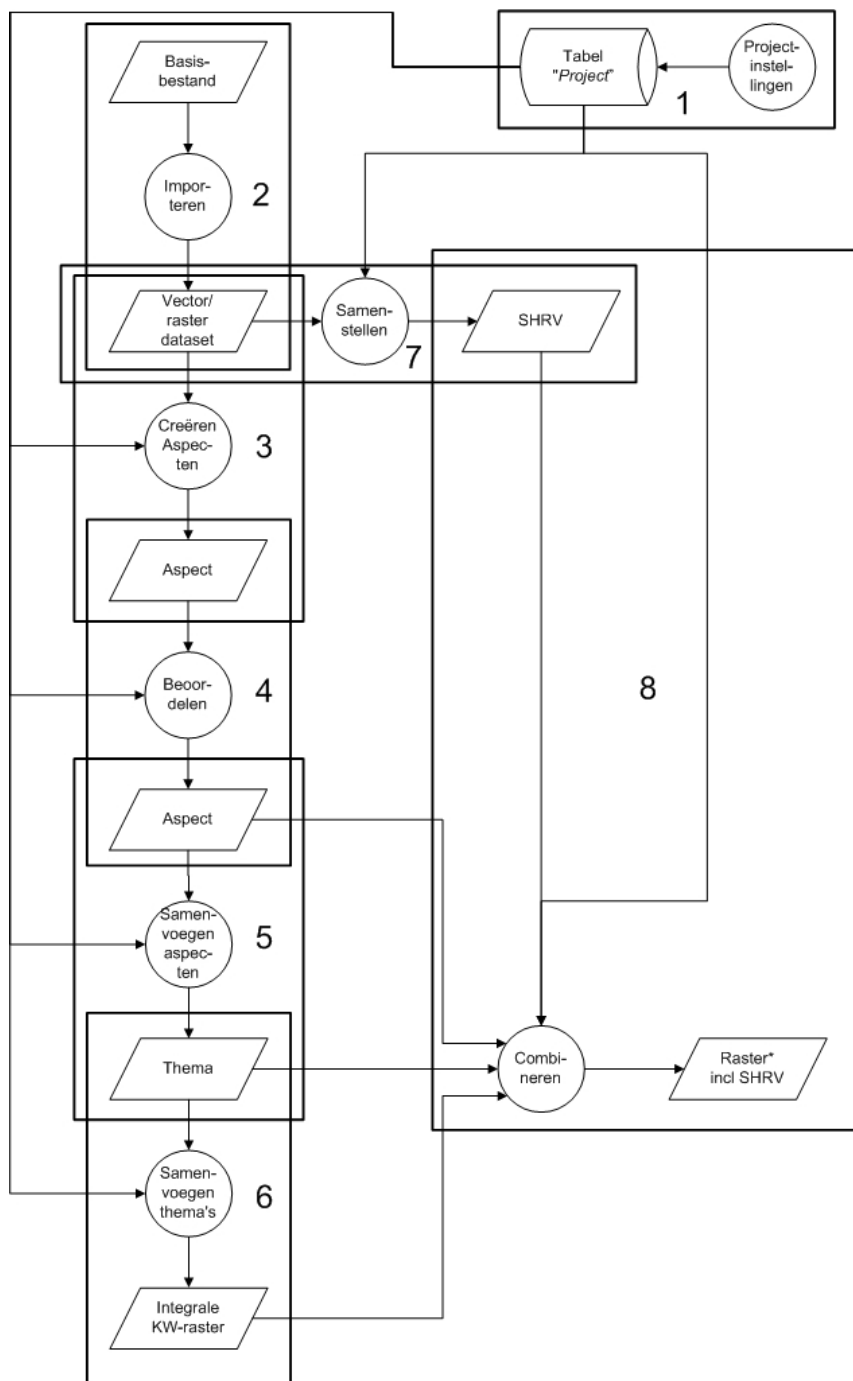
In de eerste laag vindt de voorbereiding plaats. Deze bestaat uit het instellen van een project en het importeren van

Fig. 4.2 Lagen en partities van GISMER

basisbestanden. De eerste partitie, de analyse, wordt gevormd door de derde tot en met de zesde component, het definiëren van aspecten, het beoordelen en twee keer samenvoegen. De tweede partitie bestaat uit een component, namelijk het samenstellen van harde randvoorwaarden. In de onderste laag worden de aspecten, thema's en integrale kwetsbaarheidsraster gecombineerd met de samengestelde harde randvoorwaarden.

In figuur Fig. 4.2 is per laag of partitie ook aangegeven welke componenten er toe behoren. Van de componenten is een meer nauwkeurige weergave te zien in figuur Fig. 4.3. Daarin is ook aangegeven hoe ze met elkaar samenhangen door middel van hun in- en uitvoerdata.

De eerste component is het kiezen van een bestaand project of het aanmaken van een nieuw GISMER



* = Aspect, Thema of Integrale Kwetsbaarheden raster

Fig. 4.3 Componenten van GISMER

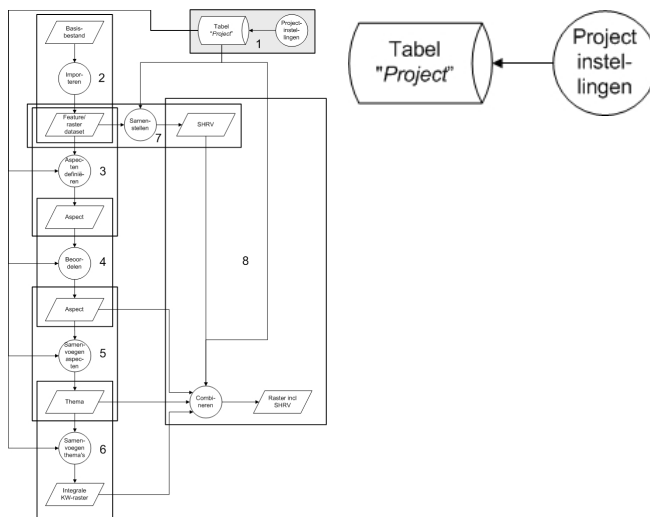
project. De tweede component bestaat uit het importeren van basisbestanden. Daarna worden in de derde component aspecten gecreëerd van de geïmporteerde vector of raster datasets. Na de voorbewerking moeten de aspecten beoordeeld worden. Het resultaat van het beoordelen is een set beoordeelde aspecten. Deze beoordeelde aspecten worden in de vijfde component via een gewogen somming samengevoegd tot een thema. De thema's zijn vervolgens de invoer voor de zesde component waarin ze samengevoegd worden tot een integraal kwetsbaarheidsraster. In de zevende component worden uit de basisbestanden, de gebieden die als harde randvoorwaarden moeten gelden geselecteerd. Dit zijn gebieden die voor verdere analyse uitgesloten moeten worden. Vervolgens worden deze in dezelfde component nog samengevoegd tot een samengestelde harde randvoorwaarde. In de achtste component kunnen tenslotte de samengestelde harde randvoorwaarden gecombineerd worden met beoordeelde aspecten, thema's of integrale

kwetsbaarheidsrasters. Zoals door een aantal contactpersonen in de ‘survey’ is aangegeven kunnen de SHRV’s op drie niveaus worden toegevoegd. Het eindresultaat is dan een beoordeeld aspect, thema of integraal kwetsbaarheidsraster waar de gebieden die niet voor verdere studie mee moeten worden genomen, zijn uitgesloten.

Hier onder wordt aan de hand van een stroomdiagram iedere component van de nieuwe GISMER applicatie besproken. In de stroomdiagrammen zijn alle relevante objectklassen (bijlage 0) en hun relaties en verbindingen opgenomen. De relevante objectklassen worden gevormd door de geografische datasets en de tabellen. In bijlage 0 is een beschrijving van de relevante objectklassen en hun attributen opgenomen.

4.2.1.1 Component 1; “projectinstellingen” technisch ontwerp

In de eerste component vindt geen bewerking van geografische data plaats. Een project moet aangemaakt worden, waarbij alleen een naam opgegeven moet worden. Deze naam wordt weggeschreven in de tabel “Project”. Wanneer in een bestaand project verder moet worden gewerkt, kan een project uit deze tabel worden gekozen.



* = Aspect, Thema of Integraal Kwetsbaarheidsraster

Fig. 4.4 Stroomdiagram component 1; “projectinstellingen”

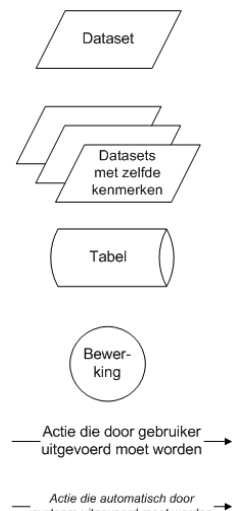


Fig. 4.5 Legenda stroomdiagrammen Functioneel ontwerp

4.2.1.2 Component 2; “importeren basisbestanden” technisch ontwerp

In de tweede component worden basisbestanden geïmporteerd in de applicatie. Hiervoor bestaan twee keuzeropties. Deze zijn afhankelijk van het dataformaat, vector of raster, van de invoer van de component. Het dataformaat van de uitvoer is gelijk aan dat van de invoer. Bij het importeren moet de metadata (University of Edinburgh and Association for Geographic Information, 1996) ingevuld worden. Deze metadata wordt weggeschreven als een XML-bestand. (Walsh, 1998) Hierbij valt te denken aan de herkomst van het basisbestand en de datum wanneer het is aangemaakt. Zoals gewenst kunnen basisbestanden van verschillende dataformaten geïmporteerd worden.

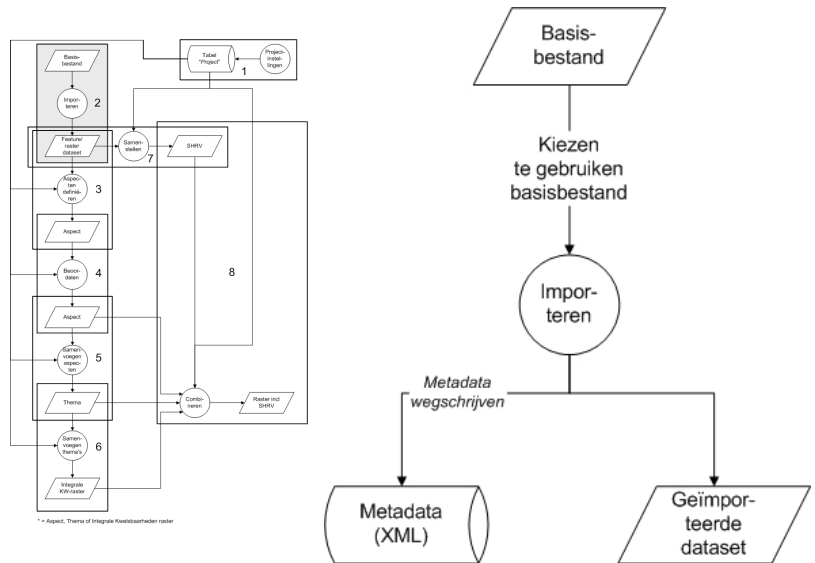


Fig. 4.6 Stroomdiagram component 2; importeren basisbestanden

4.2.1.3 Component 3; “creëren aspecten” technisch ontwerp

In deze component worden aspecten gecreëerd. De invoer voor deze component wordt gevormd door de in de tweede component geïmporteerde vector of raster datasets. De uitvoer van deze component is een aspect, dat klaar is om te beoordelen. Een aspect heeft een raster datastructuur, waarmee de wens voor een dergelijke structuur die als resultaat uit de ‘survey’ is gebleken is vervuld. Voor vector datasets zijn er twee keuzeropties om een aspect te creëren, namelijk een standaard conversie naar rasterformaat of het aanmaken van een buffer in rasterformaat. Bij een standaardconversie moet het verrasterattribuut gekozen worden. Voor beide methodes moeten ook een plangrens en celgrootte gekozen worden.

Wanneer de invoer in rasterformaat is, kan ook een buffer in rasterformaat aangemaakt worden. Daarnaast is er de keuzeroptie ‘aanpassen beoordelingsattribuut’. Hierbij is het mogelijk om voor een raster dataset een attribuut te kiezen op basis waarvan de beoordeling in de volgende component plaats moet gaan vinden. Door de keuzeroptie ‘buffer in rasterformaat’ op te nemen wordt voldaan aan de wens om deze standaard voorbewerking aan de applicatie toe te voegen.

De keuzes die gemaakt worden in deze component worden opgeslagen in de tabel “Aspect”. De gebruikersinteractie vindt per keuzeoptie plaats door middel van een batch bewerking (University of Edinburgh and Association for Geographic Information, 1996). Eerst worden alle keuzes gemaakt voor meerdere basisbestanden, vervolgens worden de bewerkingen in een keer uitgevoerd.

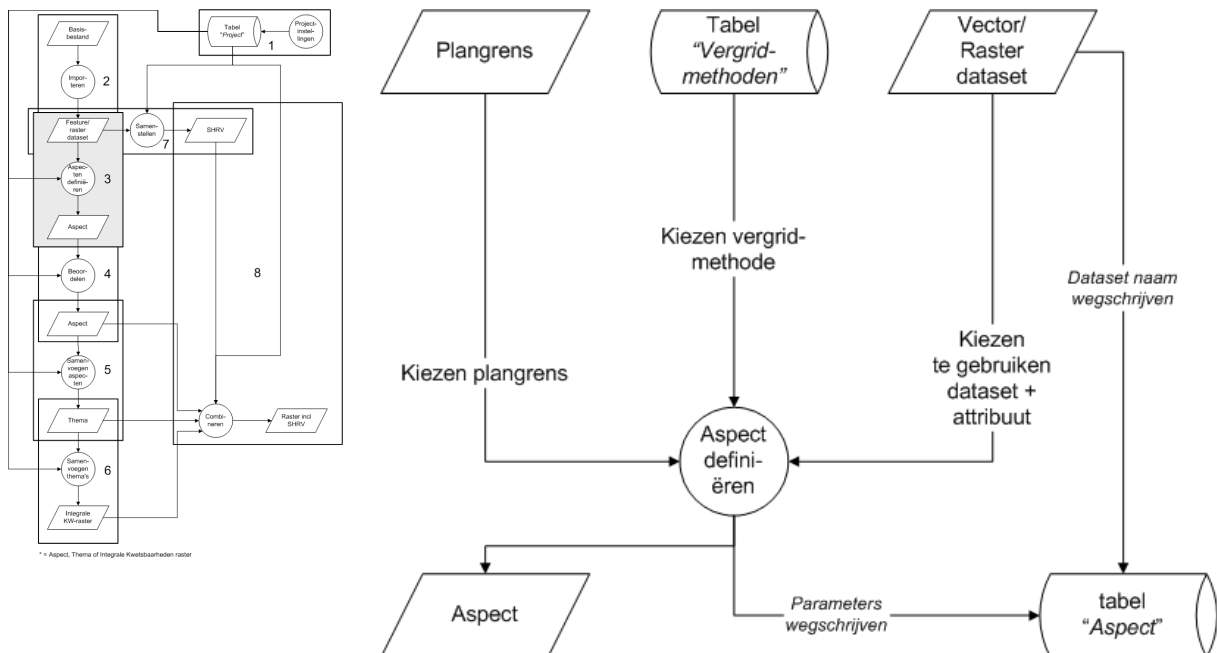


Fig. 4.7 Stroomdiagram Component 3; creëren aspecten

4.2.1.4 Component 4; “beoordelen aspecten”

Binnen de component “beoordelen aspecten” worden de aspecten beoordeeld op hun kwetsbaarheid. De gebruiker moet eerst een beoordelingsmethode kiezen en volgens deze methode scores toekennen aan de waarden van het aspect. De keuze van de beoordelingsmethode wordt weggeschreven in de tabel “Aspect”. De oorspronkelijke waarden en de scores worden weggeschreven in de tabel “Beoordeling”. In de tabel “Beoordeeld aspect”, wordt de naam van ieder beoordeeld aspect eenmaal weggeschreven. De actie die wordt uitgevoerd is een herclassificatie op basis van de opgegeven

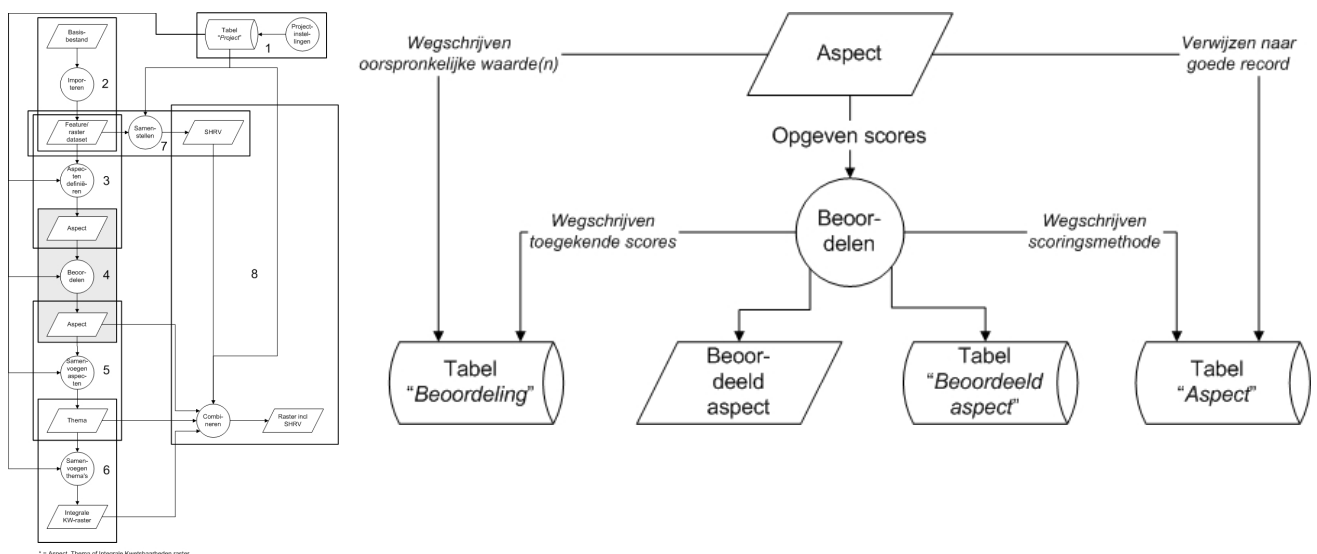


Fig. 4.8 Stroomdiagram component 4; “beoordelen aspecten”

scores.

4.2.1.5 Component 5; “samenvoegen beoordeelde aspecten”

In de vijfde component worden verschillende beoordeelde aspecten samengevoegd door middel van een gewogen somming (MCA-analyse). Eerst moet door de gebruiker bepaald worden welke aspecten samengevoegd moeten worden en vervolgens welke gewichten aan ieder samen te voegen aspect gegeven moet worden. Daarna wordt door middel van het vermenigvuldigen en optellen van de beoordeelde aspecten een thema berekend. In de tabel “Thema” wordt weggeschreven welke aspecten tot hetzelfde thema behoren en welke gewichten aan deze aspecten zijn gegeven. Bij de gewichtsverdeling moeten 100 punten verdeeld worden over de beoordeelde aspecten. De punten kunnen verdeeld worden met stappen van 1.

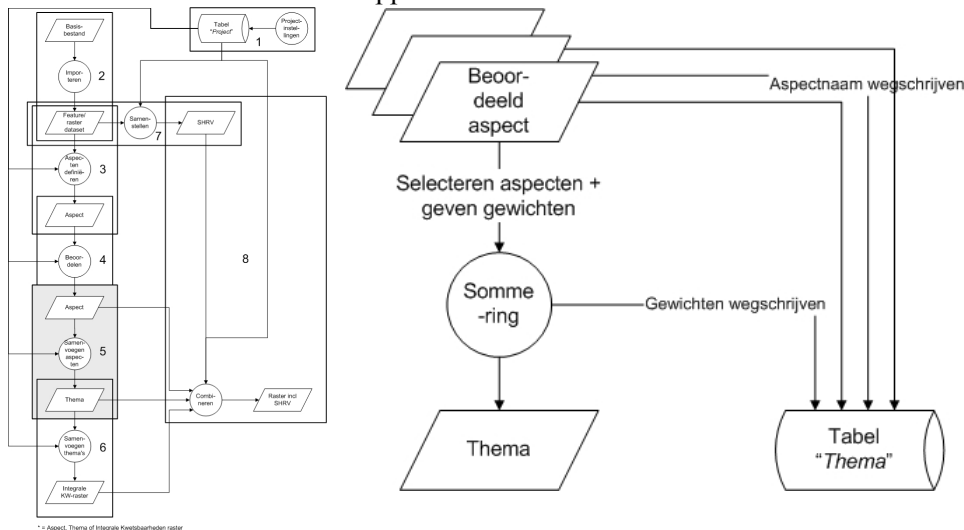


Fig. 4.9 Stroomdiagram component 5; “samenvoegen beoordeelde aspecten”

4.2.1.6 Component 6; “samenvoegen thema’s ”

De zesde component heeft veel overeenkomsten met de vijfde. Nu worden verschillende thema’s samengevoegd door middel van een gewogen somming. De gebruiker moet eerst bepalen welke thema’s samengevoegd moeten worden en vervolgens welke gewichten er aan ieder samen te voegen thema gegeven wordt. De berekening die uitgevoerd wordt is dezelfde als bij het samenvoegen van beoordeelde aspecten. In de tabel “Int_kw” wordt weggeschreven welke thema’s samengevoegd zijn en welke gewichten aan deze thema’s zijn gegeven.

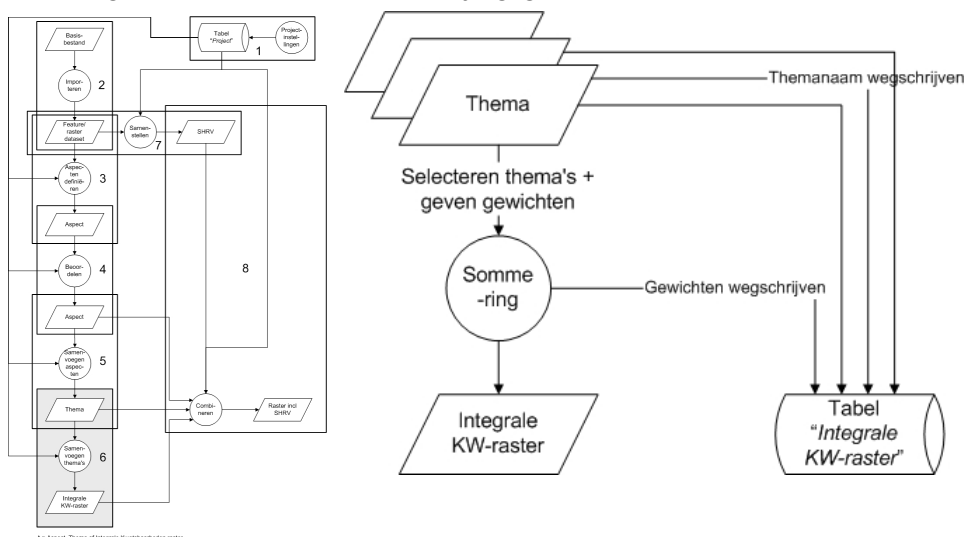


Fig. 4.10 Stroomdiagram component 6; “samenvoegen thema’s”

4.2.1.7 Component 7; “samenstellen harde randvoorwaarden”

In de zevende component worden de samengestelde harde randvoorwaarden (SHRV's) aangemaakt. De invoer voor deze component zijn de vector of raster datasets die in de tweede component zijn geïmporteerd. Middels selecties op basis van attribuutwaarden worden de delen geselecteerd die als harde randvoorwaarde opgenomen moeten worden. De gebruiker moet de selectie opgeven die vervolgens per dataset wordt opgeslagen in de tabel “SHRV”. Alle selecties worden in een batchbewerking gezet. Bij het uitvoeren van deze batch worden eerst alle selecties geconverteerd naar tijdelijke raster datasets waarbij de vooraf opgegeven plangrens en celgrootte gehanteerd worden. Daarna worden in dezelfde batch bewerking de tijdelijke raster datasets samengevoegd tot een SHRV. Per batch wordt dus een SHRV gemaakt. Voor iedere gebruikte dataset wordt in de tabel “SHRV” weggeschreven tot welke SHRV ze behoren en welke plangrens en celgrootte gehanteerd zijn. De SHRV kan hierna gebruikt worden in de zevende component.

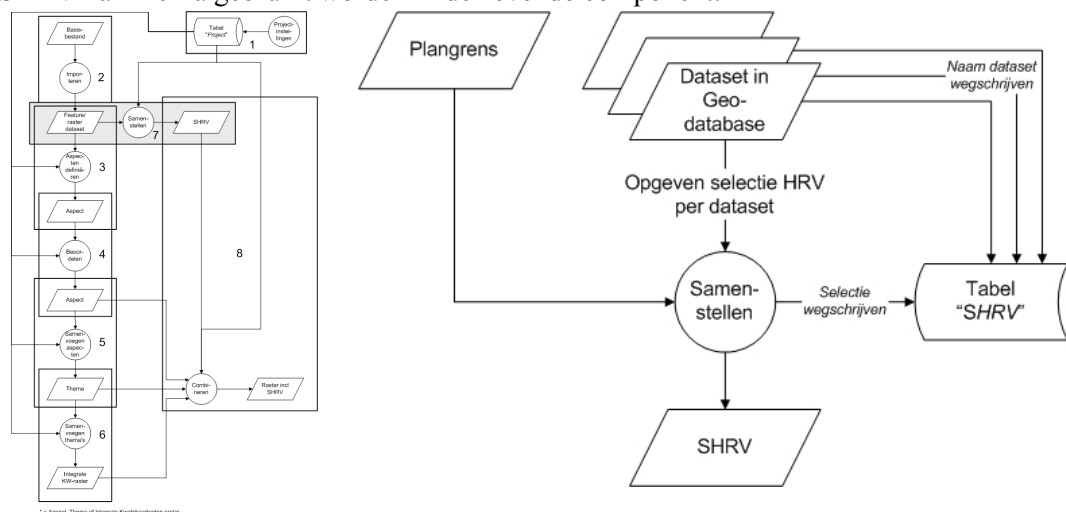


Fig. 4.11 Stroomdiagram component 7; “samenstellen harde randvoorwaarden”

4.2.1.8 Component 8; “toevoegen samengestelde harde randvoorwaarden”

In de zevende en laatste component van de applicatie kunnen de tussenproducten, dit zijn de beoordeelde aspecten, de thema's en de integrale kwetsbaarheidrasters gecombineerd worden met de in de zevende component gemaakte samengestelde harde randvoorwaarden. Hiervoor moet eerst een tussenproduct en een SHRV gekozen worden. Deze keuzes worden opgeslagen in tabellen. Voor een aspect in de tabel “Beoo aspct incl. SHRV”, voor een thema “Thema incl. SHRV” en voor een integrale kwetsbaarheidraster “Int_kw incl. SHRV”. Vervolgens wordt een herclassificatie uitgevoerd waarmee het resultaat van deze component de waarden van het tussenproduct krijgt op de plaatsen waar geen SHRV voor geldt en de waarde Nodata waar de SHRV geldt.

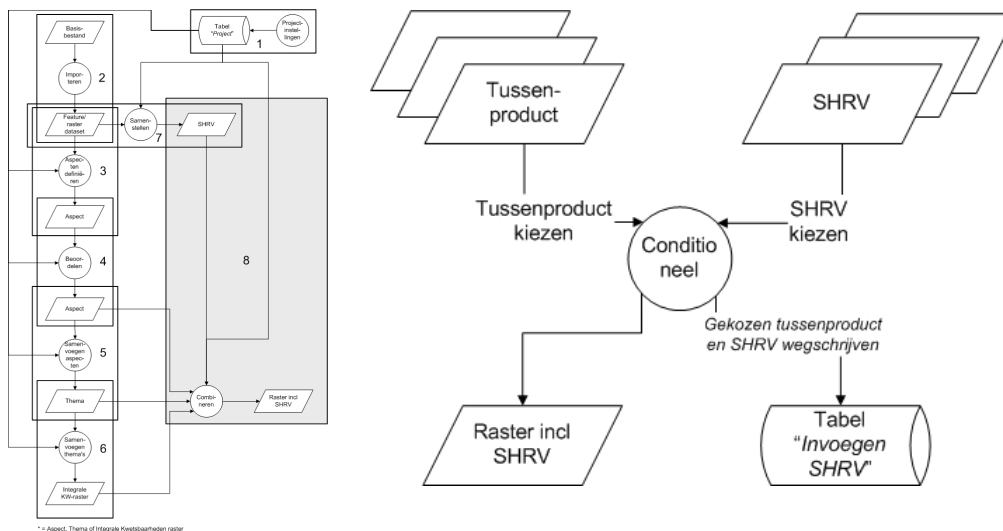


Fig. 4.12 Stroomdiagram component8; "toevoegen samengestelde harde randvoorwaarden"

[werkwijze](#)

4.2.2 Systeem ontwerp

In deze subparagraaf wordt het systeem ontwerp van GISMER besproken, eerst wordt aandacht geschonken aan het databeheer binnen de applicatie en daarna aan de besturing van de applicatie. De verschillen met betrekking tot het databeheer en de gebruikersinterface ten opzichte van de bestaande applicatie worden beschreven in hoofdstuk 5.

4.2.2.1 Databeheer binnen de applicatie [technisch ontwerp](#)

Het databeheer binnen de applicatie vindt plaats in een (Geodatabase). Het voordeel daarvan is dat geografische datasets en tabellen daar centraal in op kunnen worden geslagen. Daarnaast is het mogelijk alle keuzes, scores, wegingsfactoren en beslissingen in de database vast te leggen. Hierdoor kan procesbewaking plaatsvinden en wordt aan een duidelijke wens die uit de 'survey' voort is gekomen en door de opdrachtgever, Grontmij Geo Informatie, is gegeven voldaan. Achteraf kan bekeken worden welke wijzigingen plaats hebben gevonden en welke versies van een project, aspect, thema of integraal kwetsbaarheid raster er zijn. Ook zijn aan de hand van de database eerdere versies reproduceerbaar.

In Fig. 4.13 is een entiteit-relatie diagram opgenomen van de tabellen uit de database die voor de procesbewaking binnen GISMER gebruikt worden. Hierin staan per tabel alleen de sleutelattributen. Aan de hand van deze figuur wordt per component een toelichting gegeven op de tabellen. In bijlage 0 wordt een volledig overzicht gegeven van de database. Hierin worden alle tabellen, attributen en relaties beschreven.

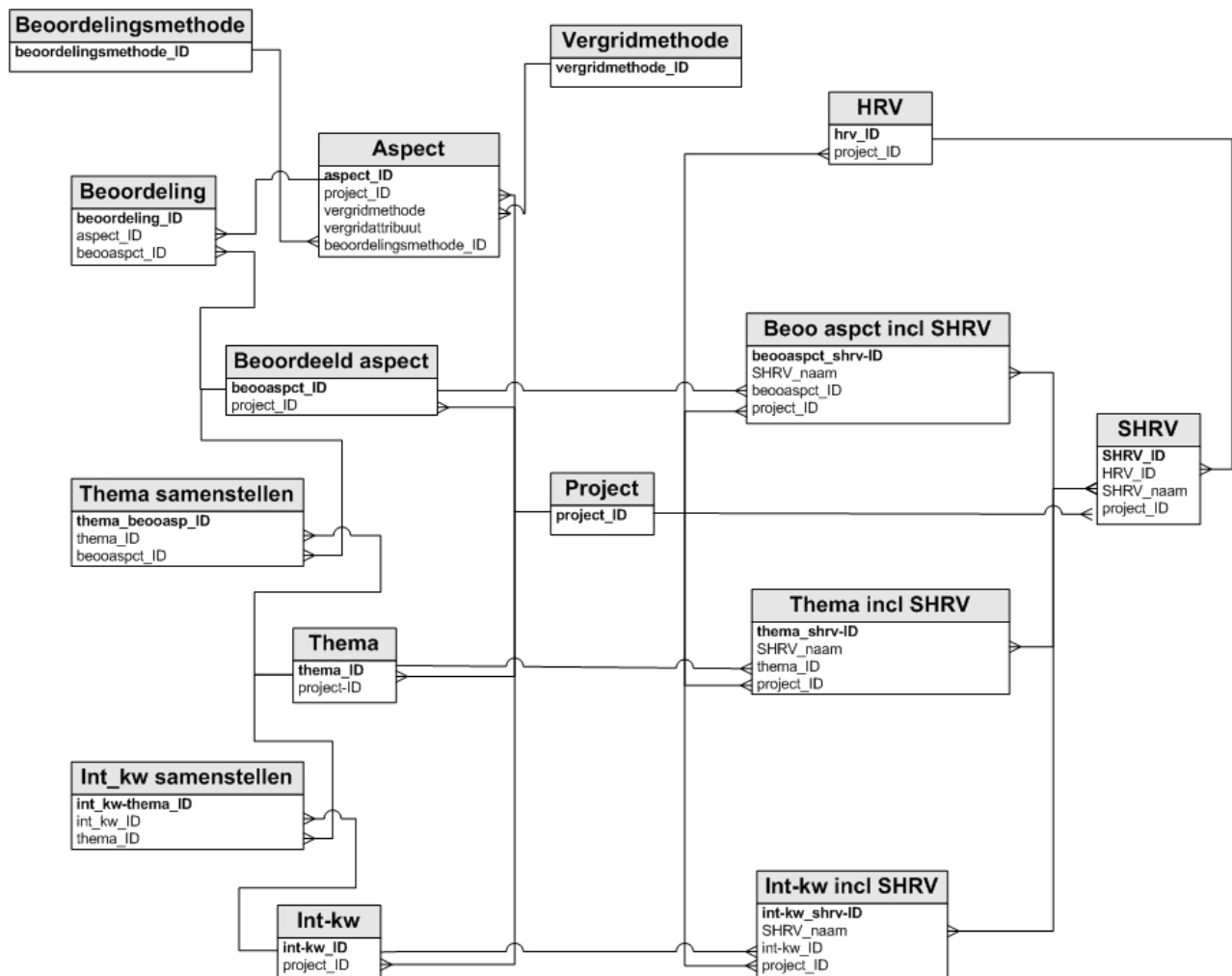


Fig. 4.13 E.R.-diagram procesbewaking database

In de eerste component wordt de tabel 'Project' uitgelezen of gevuld. Een bestaand project kan gekozen worden uit de tabel. Wanneer een nieuw project wordt begonnen, wordt dit weggeschreven in de tabel.

Daarnaast moet er ook een attribuutdomein *project_ID* gedefinieerd worden en toegepast worden op alle velden *project_ID* die voorkomen in de database. De waarde van dit domein is afhankelijk van het project waarin gewerkt wordt.

Bij het gebruik van de tweede component, het importeren van de basisbestanden worden geen tabellen gevuld die gebruikt worden voor de procesbewaking. Wel wordt metadata voor de basisbestanden ingevuld zodat de herkomst en totstandkoming van deze gegevens gedocumenteerd en achteraf geraadpleegd kunnen worden. Hiermee wordt voldaan aan de wens van de opdrachtgever van dit onderzoek, Grontmij Geo Informatie.

In de derde component worden de aspecten aangemaakt. In de tabel 'Aspect' worden alle keuzes vastgelegd die gemaakt worden om tot een aspect te komen. Om het attribuut 'vergridmethode' te kunnen vullen in de 'Aspect' tabel is de look-up tabel 'Vergridmethoden' opgenomen in de database. In deze tabel staan de beschikbare methoden die gebruikt kunnen worden om een aspect om te zetten van een vector dataset in een raster dataset. Om vast te kunnen leggen wanneer een methode met betrekking tot het creëren van aspecten toe is gevoegd aan de applicatie is een datumveld in de tabel 'Vergridmethoden' opgenomen.

Bij de vierde component, het beoordelen, worden de scores per aspect vastgelegd in de tabel *'Beoordeling'*. De methode die bij het beoordelen wordt gebruikt wordt vastgelegd in de tabel *'Aspect'*, want dat is een eigenschap van het aspect. De eigenschappen van een beoordeeld aspect worden allemaal vastgelegd in de tabel *'Beoordeeld aspect'*.

De keuzes die gemaakt worden bij het samenvoegen van beoordeelde aspecten tot een thema worden vastgelegd in de tabel *'Thema samenstellen'*. Hierbij gaat het om welke beoordeelde aspecten samen een thema vormen en welke gewichten de beoordeelde aspecten krijgen bij de gewogen somming. In de tabel *'Thema'* worden alle eigenschappen van de thema's opgeslagen.

De zesde component is nagenoeg gelijk aan de derde, dit geldt ook voor de tabellen die hiervoor in de database zijn opgenomen. In de tabel *'Int_kw samenstellen'* wordt vastgelegd welke thema's samengevoegd worden en welke gewichten daarbij gehanteerd worden. In de tabel *'Int_kw'* komen de eigenschappen van een integraal kwetsbaarheidraster.

Bij het definiëren en samenstellen van harde randvoorwaarden wordt de tabel *'SHRV'* uit de database gevuld. In deze tabel wordt vastgelegd uit welke datasets een SHRV is opgebouwd en welke selecties hierbij zijn uitgevoerd.

Bij het toevoegen van harde randvoorwaarden in de achtste component kunnen drie verschillende tabellen gevuld worden. In deze tabellen wordt vastgelegd welke combinaties gemaakt zijn tussen een beoordeeld aspect, thema of integraal kwetsbaarheidraster en een SHRV. Voor de beoordeelde aspecten is er de tabel *'Beooaspct incl. SHRV'*, voor thema's de tabel *'Thema incl. SHRV'* en voor integrale kwetsbaarheidrasters de tabel *'Int_kw incl. SHRV'*.

4.2.2.2 Besturing van de applicatie [technisch ontwerp](#)

Aan de besturing van de applicatie en de gebruikersinterface worden alleen wensen gesteld die rechtstreeks voortkomen uit functionaliteit of die zeer algemeen van aard zijn. Bij het systeem ontwerp wordt alleen de structuur van de gebruikersinterface besproken en worden niet alle menu's en dialoogschermen tot in detail besproken. De wensen met betrekking tot de gebruikersinterface die voortkomen uit functionaliteit zijn bij het systeem ontwerp dus niet verder van invloed. Aan de algemene wens dat de gebruikersinterface meer gebruiksvriendelijk en overzichtelijk moet zijn wordt wel voldaan.

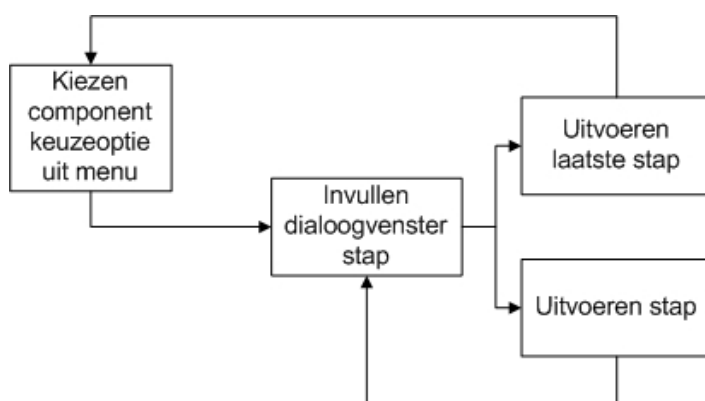


Fig. 4.14 Systeem ontwerp gebruikersinterface

De applicatie GISMER bestaat uit acht componenten. Binnen deze componenten komen een aantal keuzeopties voor. De opbouw van het menu volgt de indeling in componenten en bestaat dus uit acht keuzes op het eerste niveau. De keuzeopties komen op het tweede en derde niveau in de menustructuur. Bij component 3 moet bijvoorbeeld eerst bepaald worden of de invoer een vector- of raster dataset is, deze keuze wordt gemaakt op het tweede niveau. Daarna moet op het derde niveau nog de methode gekozen worden.

Een component bestaat uit een of meerdere stappen. Deze opbouw is ook terug te vinden bij de opbouw van de gebruikersinterface. Wanneer een component keuzeoptie uit de menustructuur is gekozen, wordt een dialoogvenster geopend voor de eerste stap binnen de keuzeoptie. In dit dialoogvenster moeten de parameters, waarden en constanten voor alle acties binnen die stap ingevuld

worden. Vervolgens wordt de eerste stap uitgevoerd waarna een dialoogvenster geopend wordt voor de volgende stap binnen dezelfde component. Deze cyclus blijft zich herhalen totdat alle stappen binnen de component zijn doorlopen. Deze opbouw van de menustructuur volgens de componenten en het per stap invullen van een dialoogscherf zorgt voor een overzichtelijk gebruik van de applicatie.

De communicatie met gebruikers voor de uitvoering van de stappen bestaan een aantal typen interfaces. Twee hiervan worden in GISMER toegepast, dit zijn de batch transformatie en de interactieve interface. Een beschrijving van deze twee is te vinden in bijlage 0. De batch transformatie wordt gebruikt bij stappen die vele malen herhaald moeten worden zoals het importeren van basisbestanden of het creëren van een aspect. Bij de stappen waarbij veel interactie met de gebruiker plaatsvindt wordt de interactieve interface toegepast.

[werkwijze](#)

4.2.3 Technisch ontwerp

Het technisch ontwerp van de nieuwe GISMER applicatie wordt in deze subparagraaf beschreven en bestaat uit drie delen. Ten eerste de componenten van de nieuwe GISMER applicatie. Alleen voor de eerste drie componenten wordt een technisch ontwerp gemaakt. Vervolgens de database waarbij alleen aandacht wordt geschonken aan de tabellen die nodig zijn voor de procesbewaking en tenslotte de gebruikersinterface van de applicatie.

Uit de 'survey' zijn, met uitzondering van de rasterstructuur voor het uitvoeren van de Multi Criteria Analyse, geen expliciete wensen naar voren gekomen ten aanzien van het technisch ontwerp. De wensen die betrekking hebben op de functionaliteit zijn meegenomen in het functioneel ontwerp en worden van daaruit ook meegenomen in het technisch ontwerp.

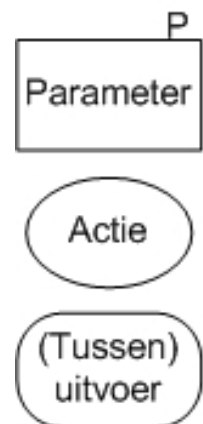


Fig. 4.15 Legenda Data-actiemodellen Technisch ontwerp

4.2.3.1 Componenten

4.2.3.2 Component 1; "projectinstellingen"

[functioneel ontwerp](#)

[implementatie](#)

In de eerste component zijn er twee keuzeopties, namelijk *nieuw project* en *bestaand project*. Er moet een bestaand project gekozen worden of een nieuw project aangemaakt worden. Voor beide opties wordt hieronder een data-actie model gegeven en wordt daar een beschrijving bij gegeven.

Keuzeoptie "Nieuw project"

De keuzeoptie *nieuw project* bestaat uit een stap die vervolgens weer uit een actie bestaat. De gebruiker moet een nieuwe projectnaam opgeven. Deze moet in de eerste actie weggeschreven worden als nieuw record in de tabel 'Project'. Hierbij moet aan de projectnaam een project_ID worden toegekend. Vervolgens moet in de tweede actie de project_ID uit worden gelezen als domeinwaarde voor het attribuutdomein 'project_ID' en als projectvariabele. Tijdens het verdere gebruik van de GISMER sessie kunnen alleen datasets behorende tot het betreffende project aangemaakt en uitgelezen worden.

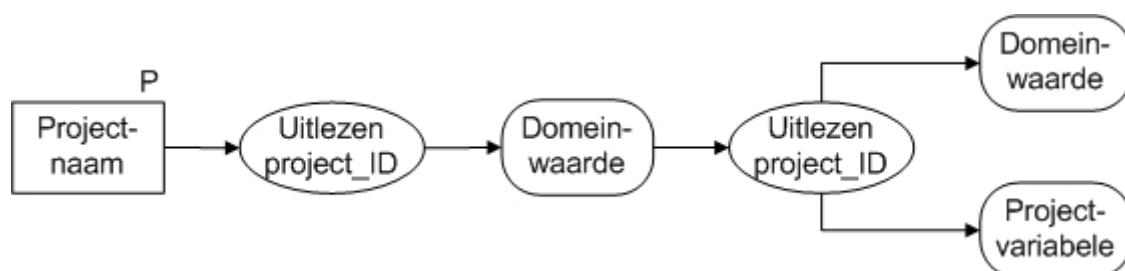


Fig. 4.16 Data-actiemodel component 1; "projectinstellingen", keuzeoptie; "nieuw project"

Keuzeoptie “Bestaand project”

De keuzeoptie *bestaand project* bestaat uit een stap. Alleen de `project_ID` van het gekozen project moet uitgelezen worden uit de tabel “*project*” als domeinwaarde voor het attribuutdomein ‘*project_ID*’ en als projectvariabele.

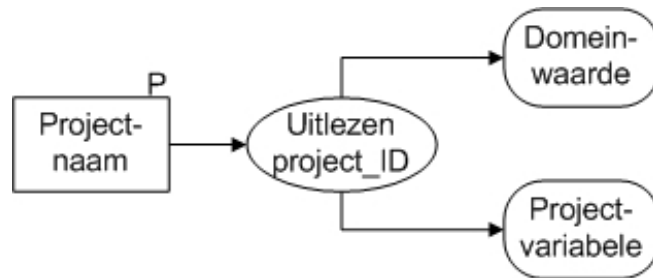


Fig. 4.17 Data-actiemodel component 1; “projectinstellingen”, keuzeoptie; “bestaand project”

4.2.3.3 Component 2; “importeren basisbestanden”

functioneel ontwerp

implementatie

Voor het importeren van de basisbestanden bestaan twee keuzeopties, voor beide keuzeopties wordt hieronder een technisch ontwerp gegeven.

Keuzeoptie “Vector basisdata”

De component keuzeoptie *vector basisbestand importeren* bestaat uit twee stappen waarbij in de eerste stap het feitelijke importeren plaatsvindt en in de tweede stap de metadata ingevuld wordt. De eerste stap bestaat uit vier acties. Deze vier acties, hun bijbehorende data en de relaties tussen de acties zijn weergegeven in figuur Fig. 4.18 aan de linkerkant van de stippellijn.

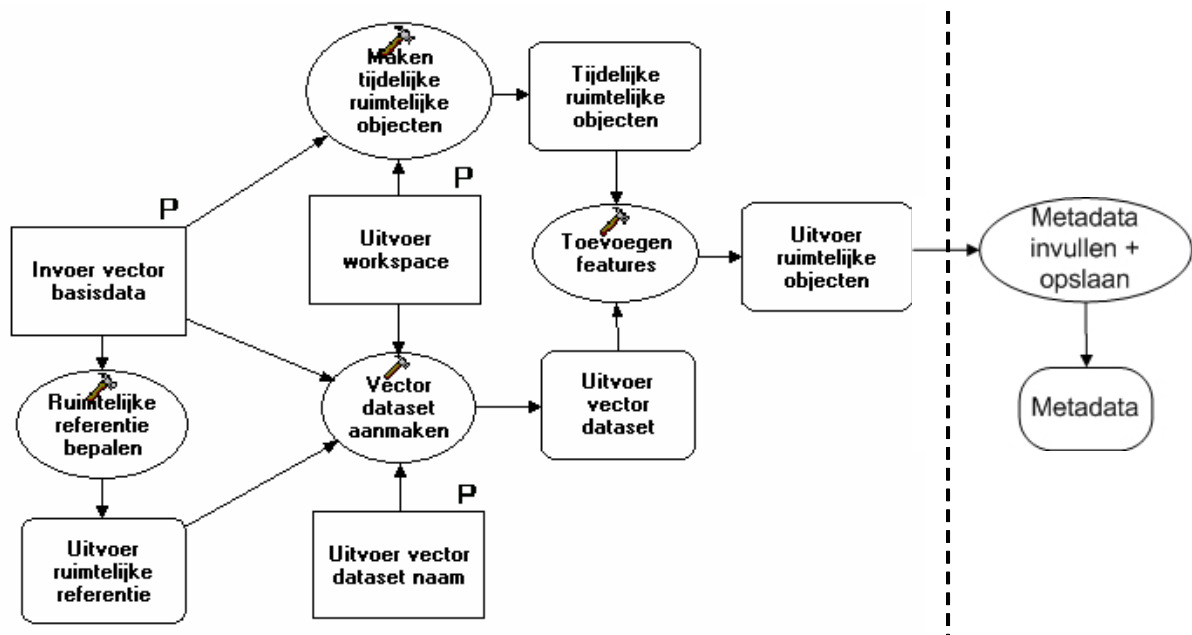


Fig. 4.18 Data-actiemodel component 2; “importeren basisbestand”, keuzeoptie; “vector basisdata”

Van een basisbestand moet eerst de ruimtelijke referentie uitgelezen worden, zodat deze toegepast kan worden op de vector dataset die uiteindelijk in de ruimtelijke database wordt opgenomen. Vervolgens moeten de tijdelijke ruimtelijke objecten (punt, lijn of vlak) gemaakt worden en moet een vector dataset aangemaakt worden waarop de ruimtelijke referentie wordt toegepast die in de eerste actie is uitgelezen. Bij de laatste actie in deze stap worden de tijdelijke ruimtelijke objecten toegevoegd aan de

gecreëerde vector dataset. Door de gebruiker moeten drie verplichte parameters opgegeven worden, namelijk de invoer basisbestanden, de uitvoer workspace en de naam van de uitvoer vector dataset. Aan de rechterkant van de stippellijn in Fig. 4.18 is de tweede stap weergegeven. Deze bestaat uit een stap waarin de metadata, die door de gebruiker ingevuld moet worden, wordt weggeschreven. Voor het invullen van de metadata moet een verbinding gelegd worden met een metadata editor. Het invullen van de metadata moet zo geïmplementeerd worden dat het onmogelijk is datasets te importeren zonder de metadata in te vullen.

Keuzeoptie “Raster basisdata”

Bij het importeren van een raster basisbestand wordt deze in feite alleen gekopieerd naar de ruimtelijke database waar alle datasets die voor het gebruik van GISMER nodig zijn in komen te staan. Dit kopiëren vindt plaats in de eerste stap van deze component keuzeoptie en bestaat uit een actie. Na deze eerste stap moet in de tweede stap de metadata voor de raster dataset in gevuld kunnen worden, net zoals bij de keuzeoptie “Vector basisdata”. In figuur Fig. 4.19 zijn de twee stappen gescheiden door de stippellijn. Aan de linkerkant vindt het importeren plaats en aan de rechterkant het invullen en wegschrijven van de metadata.

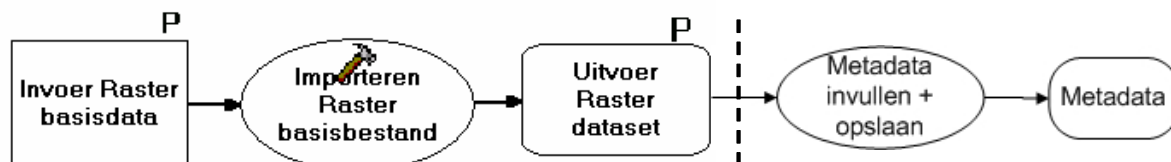


Fig. 4.19 Data-actiemodel component 2; “importeren basisbestand”, keuzeoptie; “vector basisdata”

4.2.3.4 Component 3; “creëren aspecten”

functioneel ontwerp

implementatie

De derde component is het creëren van aspecten. Hiervoor zijn vier keuzeopties, twee voor wanneer de invoer in vectorformaat is en twee voor wanneer de invoer in rasterformaat is. Voor alle keuzeopties wordt hieronder een technisch ontwerp gegeven. De keuzeoptie ‘bufferen in rasterformaat’ met als invoer vectordata is gelijk aan bufferen in rasterformaat met rasterdata als invoer. Deze twee keuzeopties worden daarom samen besproken.

Uit de ‘survey’ is gebleken dat de datastructuur waarin de analyse uitgevoerd moet worden een raster datastructuur moet zijn, is de uitvoer van deze component een raster dataset, ongeacht de keuzeoptie.

Keuzeoptie “Conversie” (invoer vector dataset)

De eerste optie voor het creëren van aspecten van een vector dataset is de conversie. Deze bestaat uit twee stappen, namelijk de conversie zelf en de procesbewaking. De eerste stap, het converteren bestaat uit een actie en is in figuur Fig. 4.20 aan linkerkant van de stippellijn weergegeven. Door de gebruiker moeten voor de bewerking vijf parameters opgegeven kunnen worden, namelijk de vector dataset, het vergridattribuut, de te hanteren celgrootte, de plangrens die als map extent (University of Edinburgh and Association for Geographic Information, 1996) functioneert en de naam en locatie van het resultaat. In de tweede stap, moeten de parameters die in de eerste stap zijn opgegeven ten behoeve van de procesbewaking weggeschreven kunnen worden in de tabel “Aspect”. Daarnaast moeten door de gebruiker de aspectnaam en versie opgegeven kunnen worden, deze moeten ook weg worden geschreven. De stap van de procesbewaking is in Fig. 4.20 aan de rechterkant van de stippellijn weergegeven.

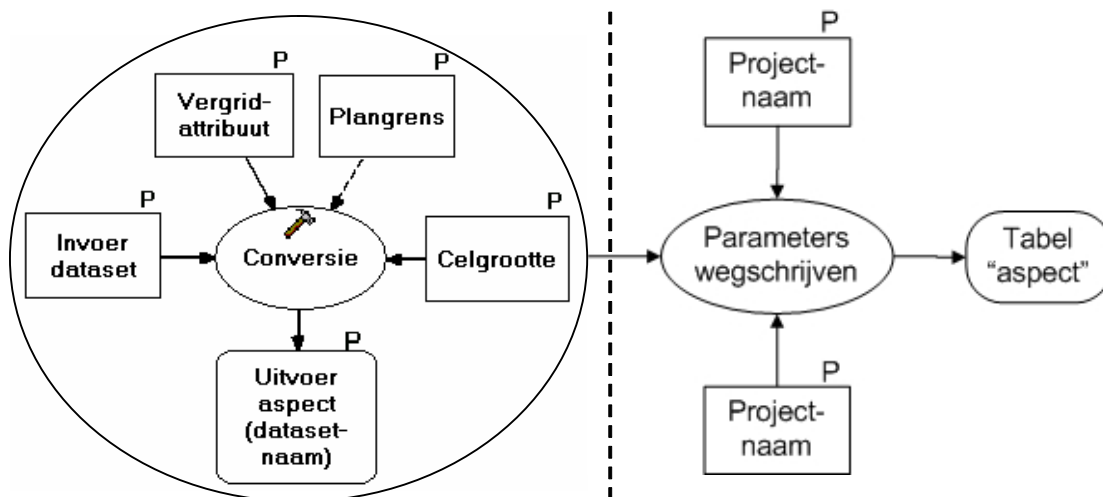


Fig. 4.20 Data-actiemodel component 3; "creëren aspecten", keuzeoptie; "conversie"

Keuzeoptie "buffer in rasterformaat" (invoer vector of raster dataset)

De tweede keuzeoptie bij het creëren van aspecten is het aanmaken van een buffer in rasterformaat. De invoer kan zowel in vector- als in rasterformaat zijn. Deze keuzeoptie bestaat uit twee stappen, het creëren van het aspect zelf en het wegschrijven van de keuzes ten behoeve van de procesbewaking. In Fig. 4.21 is aan de linkerkant van de stippellijn de stap weergegeven waarin het aspect wordt gecreëerd en aan de rechterkant de stap waarin de procesbewaking plaatsvindt. Beide stappen bestaan uit een actie. De eerste stap moet zo geïmplementeerd worden dat door de gebruiker vier parameters opgegeven kunnen worden. Deze parameters moeten allemaal in de tabel "Aspect" worden wegeschreven in de tweede stap. Ook bij deze keuzeoptie moeten door de gebruiker een aspectnaam en versie opgegeven kunnen worden in de tweede stap.

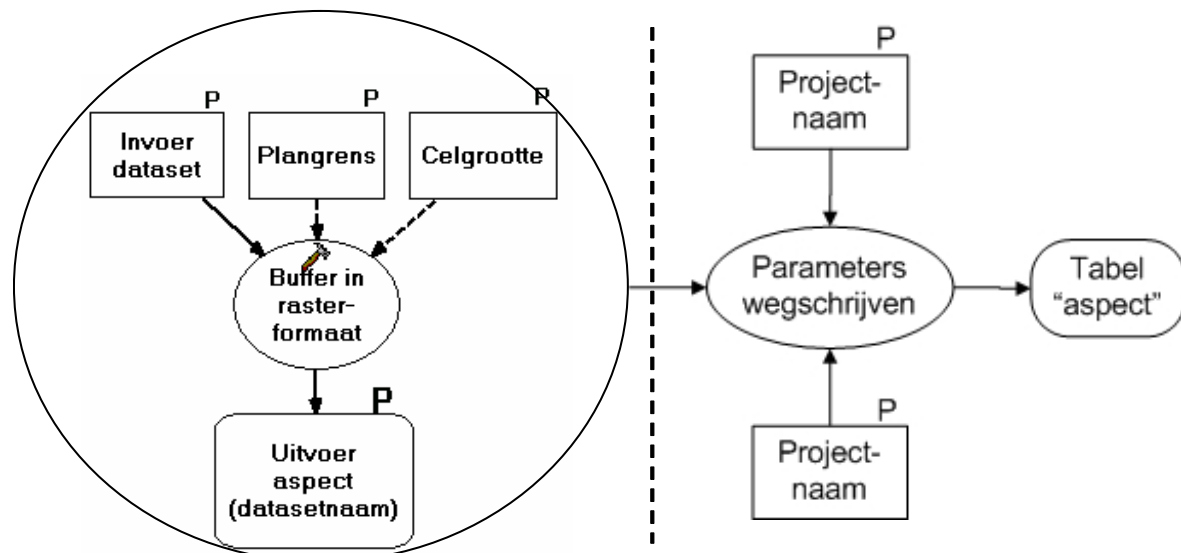


Fig. 4.21 Data-actiemodel component 3; "creëren aspecten", keuzeoptie; "Bufferen in rasterformaat"

Keuzeoptie “Beoordelingsattribuut aanpassen” (invoer raster dataset)

Bij de keuzeoptie ‘beoordelingsattribuut aanpassen’ wordt het waardeveld van de invoer raster dataset aangepast op basis van een ander attribuut uit de attribuuftabel. Deze keuzeoptie bestaat uit twee stappen. Een is de bewerking zelf. Hierin moeten door de gebruiker vijf parameters op gegeven kunnen worden. Dit zijn de invoer raster dataset, het beoordelingsattribuut, de plangrens, de celgrootte en de uitvoer aspect datasetnaam. De tweede stap is de procesbewaking waarin alle keuzes en beslissingen die gemaakt zijn bij het uitvoeren van de eerste stap in de tabel “*Aspect*” worden weggeschreven. In deze stap moet de gebruiker nog de aspectnaam en aspectversie op kunnen geven. Het data-actiemodel van deze keuzeoptie is weergegeven in Fig. 4.22 met aan de linkerkant van de stippellijn de eerste stap en aan de rechterkant de tweede stap.

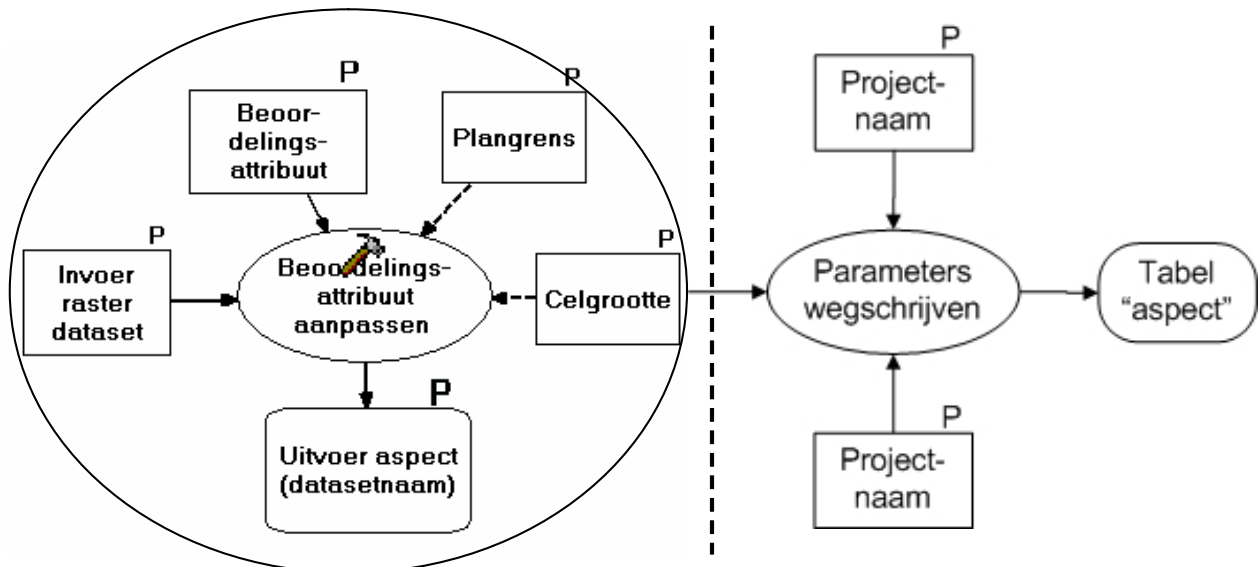


Fig. 4.22 Data-actiemodel component 3; “creëren aspecten”, keuzeoptie; “Beoordelingsattribuut aanpassen”

4.2.3.5 Databeheer (procesbewaking)

[systeem ontwerp](#)

[implementatie](#)

Om alle functionaliteit die beschreven staat in subparagraaf 4.2.2 in de database te kunnen implementeren moet data integriteit tussen de verschillende tabellen gewaarborgd worden. Wanneer een record aan de ‘een kant’ van een relatie verwijderd wordt, moeten ook alle gerelateerde records aan de ‘veel kant’ verwijderd worden. Daarnaast moeten ook de geografische datasets waarnaar verwezen wordt vanuit het verwijderde record uit de database verwijderd worden. Alle velden met een relatie moeten een numeriek veld zijn om de rekensnelheid bij bevestigingen van de database te versnellen.

4.2.3.6 Besturing van de applicatie

[systeem ontwerp](#)

[implementatie](#)

Het hoofdmenu moet geïmplementeerd worden zoals het in het systeem ontwerp is beschreven. De component keuzeopties bij de component “creëren aspecten” moeten uitgelezen worden uit de tabel “*vergridmethoden*” en de keuzeopties bij “beoordelen aspecten” uit de tabel “*beoordelingsmethoden*”. Wanneer een keuzeoptie gekozen is, moeten de bijbehorende dialoogschermen geopend worden.

4.2.4 Implementatie

[werkwijze](#)

Alleen de delen van de applicatie waarvoor een technisch ontwerp is gemaakt worden geïmplementeerd. Begonnen wordt met de eerste drie componenten van de applicatie, daarna wordt de ruimtelijke database geïmplementeerd en tot slot de gebruikersinterface.

De implementatie wordt uitgevoerd in ArcGIS Desktop 9.0 Pre Release. Daarbij wordt zoveel mogelijk gebruik gemaakt van standaardfunctionaliteit om te bekijken in hoeverre deze producten gebruikt kunnen worden voor applicatieontwikkeling.

Indien geen standaardfunctionaliteit van ArcGIS Desktop 9.0 Pre Release gebruikt kan worden, moet geprogrammeerd worden. Hiervoor bestaan twee mogelijkheden, scripting en programmeren met behulp van ArcObjects. Dit programmeren wordt niet uitgevoerd in dit project. In bijlage 0 is achtergrondinformatie over het ontwikkelen in ArcGIS Desktop 9.0 Pre Release opgenomen.

De database wordt geïmplementeerd als Geodatabase. Geodatabases bestaan bij ESRI in twee varianten, “Personal” en “Enterprise”. Bij het gebruik van GISMER worden basisgegevens niet gewijzigd, er worden alleen analyses mee uitgevoerd. Het is dus niet nodig dat meerdere mensen tegelijk de gegevens kunnen wijzigen zoals dat bij een Enterprise Geodatabase mogelijk is. Daarom is gekozen voor een Personal Geodatabase (PGDB). In bijlage 0 is meer achtergrondinformatie over databeheer in ArcGIS en in het bijzonder in de Geodatabase opgenomen.

4.2.4.1 Componenten

Bij het implementeren van de componenten is veel gebruikt gemaakt van de “Model Builder”. De meeste acties die binnen GISMER uitgevoerd moeten worden bestaan als standaard tools binnen ArcGIS 9. Met behulp van de Model Builder kunnen variabelen en parameters van deze functies aangepast worden zodat ze te gebruiken zijn voor GISMER. De modellen zijn te exporteren naar een scriptingtaal. Van ieder model dat is aangemaakt is in bijlage 0 het script opgenomen. Voor de extra functionaliteit van GISMER die buiten de standaardfunctionaliteit van ArcGIS valt zijn opties voor de implementatie gegeven door Voets, Support Engineer van ESRI Nederland. Deze opties zijn opgenomen bij de betreffende component.

4.2.4.2 Component 1; “projectinstellingen”

[technisch ontwerp](#)

Het aanmaken of kiezen van een project kan niet geïmplementeerd worden met behulp van standaardfunctionaliteit van ArcGIS. Hiervoor moet geprogrammeerd worden, dit is niet uitgevoerd in dit onderzoek.

4.2.4.3 Component 2; “importeren basisbestanden”

[technisch ontwerp](#)

Deze component heeft twee keuzeopties, namelijk vector en rasterdata importeren. De implementatie van beide keuzeopties wordt hier onder besproken.

4.2.4.4 Keuzeoptie Vector

Voor het importeren van een vector basisbestand in de Geodatabase is het model “feature class to feature class”, dat tot de standaardfunctionaliteit van ArcGIS Desktop 9.0 Pre Release behoort, aangepast. Variabelen die niet van belang zijn voor het gebruik van deze functie in GISMER zijn verwijderd en parameters die wel van belang zijn voor GISMER, zijn variabel gemaakt. Daarnaast zijn parameters, voor zover mogelijk, al vooraf gedefinieerd. Een voorbeeld daar van is de “uitvoer workspace”, hier is de Geodatabase die ook gebruikt wordt voor de procesbewaking toegekend. De naamgeving van de variabelen is aangepast naar GISMER termen, “Input features” is bijvoorbeeld aangepast naar “Invoer basisbestand”.

Voor de tweede stap, het invullen van de metadata bestaat geen standaardfunctie die gebruikt kan worden. Na het uitvoeren van de eerste actie in deze component moet een metadata editor geopend worden. Dit moet geprogrammeerd worden en is dus niet verder geïmplementeerd.

4.2.4.5 Keuzeoptie Raster

Voor de keuzeoptie importeren raster bestand bestaat geen standaard functie. Rasters kunnen niet in een personal Geodatabase gekopieerd of geïmporteerd worden. Wel kan een raster wanneer deze het resultaat is van een bewerking weg worden geschreven in een personal Geodatabase. Deze keuzeoptie moet dus geprogrammeerd worden. Het invullen van de metadata moet op dezelfde manier geïmplementeerd worden als bij de eerste keuzeoptie.

Om meerdere datasets tegelijk te importeren moet per keuzeoptie een batch gecreëerd worden. Dit is niet mogelijk met de standaardfunctionaliteit van ArcGIS 9.0. Er moet een script gemaakt worden dat het model meerdere malen aanroept of er moet een compleet nieuwe interface gebouwd worden (Voets, 2004).

4.2.4.6 Component 3; “creëren aspecten” technisch ontwerp

De keuzeopties voor het creëren van de aspecten zijn allemaal geïmplementeerd. De keuzeoptie buffer in rasterformaat is voor vector en raster hetzelfde en wordt op dezelfde manier geïmplementeerd. Deze wordt hieronder daarom maar een keer beschreven. De tweede stap, de procesbewaking is volgens het technisch ontwerp voor alle keuzeopties gelijk. Deze wordt daarom maar een keer beschreven, na de beschrijving van de laatste keuzeoptie.

4.2.4.7 Keuzeoptie Conversie (invoer vector dataset)

De eerste keuzeoptie voor vector data is het converteren van een feature class. Hiervoor is de standaard tool “feature to raster conversion” als basis gebruikt. De functie is in de model builder geplaatst en de model- en omgevingsparameters zijn variabel gemaakt.

4.2.4.8 Keuzeoptie Buffer in rasterformaat (invoer vector of raster dataset)

De keuzeoptie buffer in rasterformaat is beschikbaar voor zowel raster als vector datasets als invoer. Hier is standaard de tool “euclidean distance” (Tomlin, 1990) aangepast met behulp van de model builder. De model- en omgevingsparameters die in het technisch ontwerp staan zijn variabel gemaakt.

4.2.4.9 Aanpassen beoordelingsattribuut (invoer raster dataset)

Voor het aanpassen van het beoordelingsattribuut, de tweede keuzeoptie indien de invoerdata in rasterformaat is, is geen standaardfunctionaliteit beschikbaar die aangepast kan worden. Het is mogelijk om hiervoor een reclassify te gebruiken. Hiermee kan echter niet een-op-een een ander attribuut als “value” veld aangemerkt worden. Handmatig moeten dan de nieuwe waarden ingevuld worden, dit is niet gewenst vanwege de grote kans op fouten.

4.2.4.10 Procesbewaking in de keuzeopties

De parameters die per dataset zijn opgegeven moeten weg worden geschreven in de tabel “*aspect*”. Dit is niet uit te voeren met behulp van standaardfunctionaliteit. Een mogelijke oplossing hiervoor is, het model aan te roepen vanuit een zelf geprogrammeerd script. Het model kan vervolgens ook weer scripts aanroepen. Binnen die scripts kunnen de tabellen dan gewijzigd worden (Voets, 2004). Het is daarnaast ook niet mogelijk om extra variabelen toe te voegen aan een model, bijvoorbeeld ten behoeve van de procesbewaking. Hiervoor moet door de programmeur zelf een interface ontwikkeld worden.

4.2.4.11 Batchbewerking per keuzeoptie

Om meerdere aspecten tegelijk te creëren moet per keuzeoptie een batch gecreëerd worden. Net als bij de tweede component is dit niet mogelijk met de standaardfunctionaliteit van ArcGIS Desktop 9.0.

Hiervoor moet dus ook een script gemaakt worden dat het model meerdere malen aanroept of er moet een compleet nieuwe interface gebouwd worden.

4.2.4.12 Procesbewaking database [technisch ontwerp](#)

De database die gebruikt wordt voor de procesbewaking is aangemaakt en ingericht in ArcCatalog. Eerst is een nieuwe Personal Geodatabase aangemaakt. Deze Geodatabase wordt bij het gebruik van GISMER niet alleen gebruikt voor de procesbewaking, maar ook als opslagplaats voor de geografische data. De tabellen die nodig zijn voor de procesbewaking zijn aangemaakt volgens de beschrijving uit bijlage 0. Alle een-op-veel relaties die in de database voorkomen en in Fig. 4.13 zijn aangegeven, zouden als composite relationship class gedefinieerd moeten worden. Dit bleek echter niet mogelijk. Een tabel kan in de Personal Geodatabase slechts met een attribuut aan de “veel-kant” van een composite relatie zitten. Het is wel mogelijk om een attribuut aan de “veel-kant” van een composite relationship class te laten zitten en een of meerdere andere attributen aan de “veel-kant” van een simple relationship class.

Naast het aanmaken van alle benodigde tabellen en relationship classes is ook het attribute domain “*project-ID*” aangemaakt. Dit attribute domain is toegepast op alle velden “*project_ID*” die voorkomen in de tabellen van de database met uitzondering van de tabel “*project*”. De geldige waarde voor het attribute domain wordt pas bij het gebruik van de applicatie opgegeven. Deze functionaliteit moet geïmplementeerd worden bij de eerste component.

4.2.4.13 Menustructuur [technisch ontwerp](#)

De menustructuur van GISMER zoals deze in Fig. 4.23 te zien is, is aangemaakt met behulp van de menu editor uit Visual Basic 6. Hierbij is afgeweken van het functioneel en technisch ontwerp. De keuzeopties voor het creëren en beoordelen van aspecten zijn namelijk niet uitgelezen uit de tabellen uit de database. Bij de uiteindelijke implementatie moet dit wel geprogrammeerd worden. Wanneer een model of het daarvan afgeleide script geopend wordt, wordt meteen een gebruikersinterface in de vorm van een dialoogschermdisplay gegenereerd. De dialoogschermen zijn opgenomen in bijlage 0.

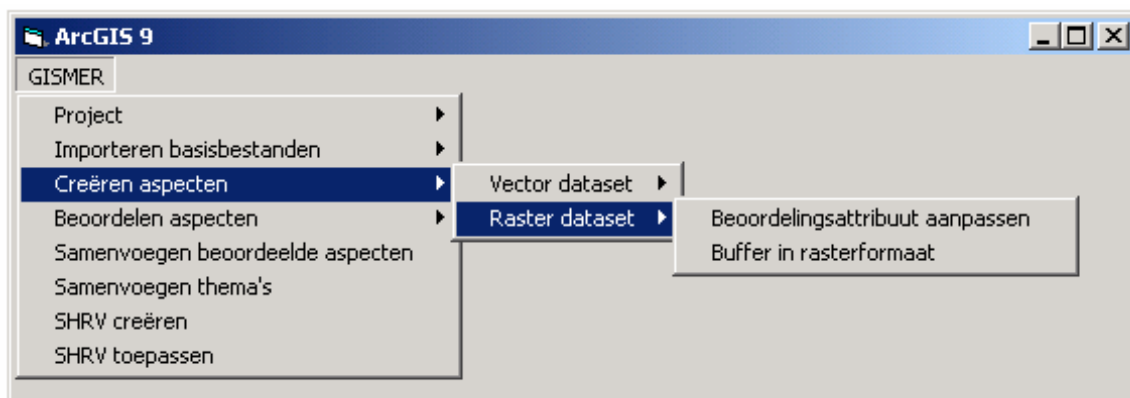


Fig. 4.23 Menustructuur GISMER

5 Vergelijking bestaande en nieuwe applicatie

In deze paragraaf worden de bestaande en de nieuwe applicatie met elkaar vergeleken op drie aspecten. Deze drie aspecten zijn de functionaliteit en de gebruikersinterface, het databasebeheer en tot slot de programmeeromgeving.

5.1 Veranderingen met betrekking tot de functionaliteit en de gebruikersinterface

Hieronder wordt eerst per component de veranderde functionaliteit beschreven. Extra functionaliteit die in de gehele applicatie is toegevoegd is de procesbewaking. Deze verandering is in paragraaf 4.2 al besproken en wordt daarom hier niet nogmaals beschreven. Na de veranderingen ten aanzien van de functionaliteit worden de veranderingen met betrekking tot de gebruikersinterface besproken.

Componenten

Tabel 5.1 Veranderingen met betrekking tot de functionaliteit

Component nieuwe applicatie	Functionaliteit bestaande applicatie	Functionaliteit nieuwe applicatie
1	Per project moet een nieuwe datastructuur geïnstalleerd worden.	Meerdere projecten kunnen naast elkaar in de database voorkomen.
2	- Importeren van basisbestanden moet buiten de applicatie om. Basisbestanden moeten in de cov-workspace gezet worden. - Alleen coverages kunnen als basisbestand gebruikt worden. - N.v.t. - Metadata wordt niet ingevuld.	- Basisbestanden importeren is een aparte component geworden binnen de applicatie. - Naast coverages kunnen ook andere formaten zoals shape, feature class en raster gebruikt worden. - Meerdere basisbestanden kunnen tegelijk geïmporteerd worden d.m.v. een batch bewerking. - Metadata moet ingevuld worden.
3	- Een keuzeoptie voor het verrasteren (creëren van aspecten) beschikbaar. - Alleen vector data als invoer mogelijk. - Aspecten moeten een voor een gecreëerd worden.	- Vier keuzeopties voor het creëren van aspecten beschikbaar. - Vector en Rasterdata als invoer mogelijk. - Meerdere aspecten kunnen tegelijk gecreëerd worden d.m.v. een batch bewerking
4	Keuze SHRV moet voor het beoordelen plaatsvinden	Keuze SHRV kan na het beoordelen plaatsvinden.
5	- SHRV is al toegepast bij het beoordelen. - Gewichtsverdeling in stappen van 5.	- SHRV kan na het samenvoegen plaatsvinden. - Gewichtsverdeling in stappen van 1.
6	Veranderingen gelijk aan component 5.	Veranderingen gelijk aan component 5.
7	SHRV moet vooraf gedefinieerd worden.	SHRV kan op ieder willekeurig moment na het importeren van de basisbestanden plaatsvinden.

	• Samenstellen vindt plaats in 2 stappen, eerst HRV en vervolgens SHR.V. • Alleen raster datasets kunnen als invoer gebruikt worden.	• Samenstellen vindt plaats in 1 stap. • Vector en raster datasets kunnen als invoer gebruikt worden.
8	• SHR.V wordt al voor het beoordelen toegepast, deze component is dus niet aanwezig.	• SHR.V kan op drie niveaus worden toegepast.

Gebruikersinterface

Tabel 5.2 Veranderingen met betrekking tot de gebruikersinterface

Bestaande applicatie	Nieuwe applicatie
• Opbouw van de menustructuur volgt niet volledig de indeling in componenten. • Niet voor iedere stap is er exact één dialoogscher.m.	• Opbouw menustructuur volgt volledig de opbouw in componenten. • Per stap is er een dialoogscher.m.

5.2 Veranderingen met betrekking tot het databeheer

In deze subparagraaf worden de veranderingen met betrekking tot het databeheer besproken. Besproken wordt welke data op moet worden geslagen, hoe dit op moet worden geslagen en welke gevolgen dit heeft.

Tabel 5.3 Veranderingen met betrekking tot het databeheer

Bestaande applicatie	Nieuwe applicatie
• Zes verschillende typen geografische datasets moeten op worden geslagen: (basisbestanden, aspecten, SHR.V's, beoordeelde aspecten, thema's en visies.) • Eén dataformaat (coverage) mogelijk als invoer voor de applicatie. • Datasets en gegevens worden opgeslagen in een file directorystructuur. • Gemaakte keuzes tijdens het gebruik van GISMER worden niet vastgelegd. • Gemaakte keuzes zijn niet reproduceerbaar.	• Negen verschillende typen geografische datasets moeten op worden geslagen: (geïmporteerde basisbestanden, aspecten, beoordeelde aspecten, thema's, integrale kwetsbaarheidrasters, SHR.V's, beoordeelde aspecten incl. SHR.V, thema's incl. SHR.V en integrale kwetsbaarheidrasters incl. SHR.V) • Meerdere dataformaten (o.a. coverage, shape, feature dataset en raster dataset) zijn mogelijk als invoer voor de applicatie. • Datasets en gegevens worden opgeslagen in een Geodatabase. • Gemaakte keuzes tijdens het gebruik van GISMER worden vastgelegd. • Gemaakte keuzes zijn reproduceerbaar.

5.3 Veranderingen met betrekking tot de programmeeromgeving

Tabel 5.4 Veranderingen met betrekking tot de programmeeromgeving

Bestaande applicatie	Nieuwe applicatie
Keuze ontwikkelomgeving beperkt tot ArcInfo Macro Language (AML). Commando gestuurde taal AML gebruikt voor de implementatie.	Vrije keuze ontwikkelomgeving mits deze object georiënteerd is. ArcGIS Desktop 9.0 PreRelease als ontwikkelomgeving gebruikt. Model builder, scripting en programmeren in Visual Basic m.b.v. Arcobjects.
Geen mogelijkheden tot integratie met andere softwarepakketten.	Goede mogelijkheden tot integratie met andere softwarepakketten.
Hergebruik van stukken applicatiecode d.m.v. opnieuw aanroepen van een complete macro.	Hergebruik van stukke applicatiecode door inkapseling. Bijvoorbeeld het plaatsen van een model of script binnen een nieuw model.
Gebruikersinterface en bewerking zijn in applicatiecode gescheiden van elkaar.	Gebruikersinterface en bewerking geïntegreerd in applicatiecode.

6 Evaluatie

In dit hoofdstuk wordt het onderzoek geëvalueerd. Het hoofdstuk is opgebouwd uit drie paragrafen. Eerst komen in paragraaf 5.1 de conclusies aan de orde waarmee antwoorden worden gegeven op de vier onderzoeksvragen uit paragraaf 1.3. Vervolgens worden in paragraaf 5.2 enkele kanttekeningen bij het onderzoek gemaakt en tenslotte worden in paragraaf 5.3 nog aanbevelingen gedaan voor verder onderzoek.

6.1 Conclusies

Aan het begin van dit onderzoek is het probleem gesteld dat het nog onduidelijk is welke gevolgen het inbrengen van de Geodatabase en het object georiënteerd programmeren hebben voor het ontwikkelen van de applicatie GISMER in ArcGIS. Hieruit is het doel van dit onderzoek voortgekomen, namelijk: “Het analyseren van de impact voor gebruikers en programmeurs van de introductie van de Geodatabase en het object georiënteerd programmeren op de migratie van de applicatie GISMER naar ArcGIS.” Om aan deze doelstelling te kunnen voldoen zijn in paragraaf 1.3 vier onderzoeksvragen opgesteld. In deze paragraaf wordt antwoord gegeven op deze onderzoeksvragen.

1. Wat is de impact van de introductie en de implementatie van de Geodatabase en het object georiënteerd programmeren op de migratie van GISMER naar ArcGIS?

Geodatabase

Aangezien in subparagraaf 4.2.4 gekozen is om het onderzoek verder te richten op de Personal Geodatabase (PGDB), zijn de conclusies die hieronder beschreven staan ook alleen van toepassing op de PGDB.

Gebleken is dat de PGDB goed dienst kan doen als centrale opslagplaats voor data. Geografische datasets in verschillende dataformaten en structuren kunnen opgeslagen worden in een PGDB. Daarnaast kunnen losse tabellen met verdere informatie bijvoorbeeld over de geografische datasets opgenomen worden.

De PGDB kan, wanneer alleen gebruik wordt gemaakt van standaardfunctionaliteit, niet aan alle wensen en eisen die gesteld worden in het functioneel en technisch ontwerp voor GISMER voldoen. In het bijzonder kan de gewenste procesbewaking en het reproduceren van eerder gemaakte keuzes niet plaatsvinden, doordat data integriteit niet gewaarborgd kan worden en queries op basis van meerdere tabellen niet uitgevoerd kunnen worden.

Wanneer van standaardfunctionaliteit van de PGDB wordt afgeweken en een stuk gedrag van tabellen geprogrammeerd gaat worden met behulp van ArcObjects is wel integriteit te waarborgen. Ook is het mogelijk om zo toegang te krijgen tot meerdere tabellen in een bevraging.

De Geodatabase is vooral geschikt om binnen feature datasets (vector) ongeldige features uit te sluiten. Hiervoor zijn een aantal handige tools beschikbaar, zoals subtypes en attribuut domeinen. Het uitsluiten van ongeldige waarden in tabellen of complete datasets op basis van informatie uit tabellen is niet mogelijk.

Object georiënteerd programmeren

Aangezien ArcGIS een COM structuur heeft, is voor het uitbreiden van de applicatie de keuze van de programmeertaal vrij. De enige vereiste is dat de programmeertaal COM-ondersteunend is. Door de COM-structuur van ArcGIS, kan bij het ontwikkelen van nieuwe applicaties ook gebruik gemaakt worden van functionaliteit van andere softwarepakketten met een COM-structuur zoals Excel.

Doordat alle technieken (model builder, scripting, programmeren m.b.v. ArcObjects) voor het ontwikkelen van tools en applicaties binnen ArcGIS Desktop 9.0 Pre Release uitwisselbaar zijn wordt dit veel eenvoudiger en zijn applicaties makkelijker te onderhouden. De ontwikkelaar kan de techniek kiezen die het meest geschikt is voor het ontwikkelen van die specifieke tool of applicatie. Het resultaat daarvan is in een later stadium te combineren met andere tools of applicaties die met behulp van een andere techniek zijn ontwikkeld.

Veel bewerkingen zijn te implementeren met behulp van standaardfunctionaliteit van ArcGIS Desktop 9.0 Pre Release. Om een complete applicatie te maken van deze losse bewerkingen en om complexe tools of functies te ontwikkelen die afwijken van standaardfunctionaliteit moet geprogrammeerd worden met behulp van ArcObjects. Een ontwikkelaar van complete applicaties of complexe tools en functies moet dus altijd kennis hebben van ArcObjects en een COM-ondersteunende programmeertaal. Dit geldt ook voor Grontmij Geo Informatie.

Bij het ontwikkelen van complexe tools of functies met behulp van ArcObjects moet veel geprogrammeerd worden. Ontwikkelde tools of functies kunnen echter wel opnieuw gebruikt worden binnen dezelfde applicatie of zelfs binnen een andere applicatie. Dit hergebruik vindt plaats door middel van inkapseling. Ook is het mogelijk modellen en scripts in te kapselen. Dit kan door een model of script in een ander model op te nemen. De meerwaarde van object oriëntatie zit dus in de inkapseling en het hergebruik van stukken van een applicatie (objecten). Een voorbeeld hiervan is het terugkeren van een batchbewerking in verschillende componenten. De batch hoeft maar een keer geprogrammeerd te worden waarna telkens andere bewerkingen binnen deze batch geplaatst kunnen worden.

2. Aan welke functionele eisen en wensen moet de nieuwe GISMER applicatie voldoen?

Deze onderzoeksvraag is volledig beantwoordt in paragraaf 4.1. Om niet in herhaling te vallen wordt daarom verwezen naar die paragraaf. Hier worden nog even kort de belangrijkste functionele eisen en wensen genoemd.

- De functionaliteit van de bestaande applicatie moet behouden blijven. Belangrijk hierbij zijn de kwetsbaarheidbenadering en de multi-criteria methode.
- Gemaakte keuzes moeten op worden geslagen en vervolgens reproduceerbaar zijn. Procesbewaking moet een belangrijk aandachtspunt zijn in de nieuwe GISMER applicatie.
- In plaats van het vooraf uitsluiten van gebieden moeten de samengestelde harde randvoorwaarden achteraf en op meerdere niveaus toegepast kunnen worden.
- De applicatie moet helder en overzichtelijk zijn en de gevolgde werkwijze moet uit te leggen zijn.

3. Hoe ziet een functioneel ontwerp voor de nieuwe GISMER applicatie er uit?

Het functioneel ontwerp voor GISMER is opgebouwd volgens de methodiek die beschreven is in subparagraaf 3.2.1 en gevisualiseerd is in Fig. 3.4. Hierbij is gebleken dat applicatie ontwerp fasen parallel lopen met project fasen en applicatie fasen.

Het opstellen van het functioneel ontwerp vindt plaats in de projectfase formalisatie. De wensen ten aanzien van de functionaliteit van de applicatie worden in deze fase opgesteld en de applicatie wordt verdeeld in een aantal losse componenten. Voor GISMER is de gewenste functionaliteit besproken in paragraaf 4.1. Een functioneel ontwerp voor de nieuwe GISMER applicatie dat volgens deze methode tot stand is gekomen is bij de resultaten in subparagraaf 4.2.1 besproken.

4. Hoe ziet een technisch ontwerp voor een bouwsteen van de nieuwe GISMER applicatie er uit?

Ook bij het opstellen van het technisch ontwerp is gebleken dat applicatie ontwerp fasen parallel lopen met project fasen en applicatie fasen. Het technisch ontwerp loopt parallel met de projectfase implementatie. De componenten worden verder opgedeeld in stappen. Voor alle stappen worden data-

actie modellen opgesteld. Voor de eerste drie componenten van GISMER is een technisch ontwerp opgesteld in subparagraaf 4.2.2.

6.2 Kanttekeningen

De aanleiding voor dit onderzoek was de introductie van de nieuwe softwarelijn ArcGIS van ESRI. Dit onderzoek is echter niet alleen gericht op deze software. Het gehele ontwerp voor GISMER is onafhankelijk van een bepaalde omgeving gemaakt. Pas bij de implementatie is de keuze gemaakt ArcGIS te gebruiken als implementatieomgeving. Het zou ook heel goed mogelijk zijn de applicatie te implementeren met behulp van Intergraph en Oracle producten.

De “survey” is uitgevoerd onder dertien contactpersonen. Dit aantal is niet statistisch getoetst. In dit onderzoek is na het uitvoeren van deze dertien interviews aangenomen in overleg met de opdrachtgever van dit onderzoek, Grontmij Geo Informatie, dat het niet meer nodig was meerdere interviews te houden. De resultaten van de interviews waren namelijk vrijwel allemaal gelijk, terwijl de contactpersonen bij een aantal verschillende organisaties werkzaam zijn en hierdoor een heterogene groep vormen.

De resultaten en conclusies van dit onderzoek zijn alleen van toepassing op de Personal Geodatabase van ESRI. Resultaten en conclusies zijn niet direct toepasbaar op andere formaten GIS Geodatabase, zoals een enterprise Geodatabase of een andere (niet ESRI) GIS Geodatabase.

Het onderzoek met betrekking tot object oriëntatie is veelal gebaseerd op theorie en beweringen van experts (Stuiver 2003, 2004, Voets, 2004). In dit onderzoek is alleen standaardfunctionaliteit van ArcGIS Desktop 9.0 Pre Release gebruikt voor de implementatie. Conclusies met betrekking tot object oriëntatie zijn subjectief van aard.

6.3 Aanbevelingen

De bestaande en nieuw GISMER applicaties geen beide uit van de kwetsbaarheidbenadering. Dit is een methode waarbij het studiegebied het uitgangspunt is. Daarnaast zijn er ook methoden waarbij gekeken wordt vanuit maatregelen, de zogenaamde effectgerichte methoden. Uit de ‘survey’ is gebleken dat het moeilijk is deze effectgerichte methoden te automatiseren en in een applicatie te implementeren. In verder onderzoek moet onderzocht worden hoe deze methoden wel te automatiseren en te implementeren zijn.

Uit de ‘survey’ is onder andere naar voren gekomen dat het gewenst is om voor veel voorkomende criteria in een milieu effect rapportage (MER) tot een gestandaardiseerde technische voorbereiding van de data te komen. Deze technische voorbereiding zou dan opgenomen moeten worden in de functionaliteit van GISMER. Welke criteria dit zijn is niet onderzocht in dit onderzoek, hier moet verder onderzoek naar gedaan worden, bijvoorbeeld door middel van het monitoren van een groot aantal milieu effect rapportages. Vervolgens kunnen ook de resultaten daarvan omgezet worden in extra functionaliteit van GISMER.

De bestaande GISMER applicatie beschikt over een presentatie of lay-out component. Bij het ontwerpen van de nieuwe applicatie is hier geen aandacht aan besteedt. Bij het verder ontwikkelen van de nieuwe GISMER applicatie moet een dergelijke component toe worden gevoegd. In het bijzonder moet rekening gehouden worden met presentaties via internet en koppelingen met een elektronisch vergader systeem (crealogic, 1999).

Zoals bij de kanttekeningen gebleken is, zijn de resultaten en conclusies van dit onderzoek alleen van toepassing op de Personal Geodatabase van ESRI. Om een goed beeld te geven van de migratie van

een file-directory structuur naar een GIS- Geodatabase structuur voor de applicatie GISMER moet de implementatie ook plaats vinden in een andere omgeving, en in een enterprise Geodatabase. Hiervoor zou bijvoorbeeld Oracle Spatial ontsloten via ArcSDE (ESRI, 2003) of via een ander pakket, wanneer de applicatie in een andere omgeving geïmplementeerd wordt, gebruikt kunnen worden. Niet alleen de bevindingen uit dit onderzoek kunnen dan aangescherpt worden, ook kan een vergelijking gemaakt worden tussen de verschillende GIS Geodatabase formaten.

Het onderzoek naar de impact van object georiënteerd ontwikkelen is vooral theoretisch uitgevoerd. De bevindingen zouden verder onderbouwd moeten worden met verder praktisch onderzoek. Daarbij moet gelet worden op de interoperabiliteit van modellen, scripts en code geschreven met behulp van ArcObjects. Daarnaast moet in dit vervolgonderzoek ook aandacht geschonken worden aan ArcGIS Engine, het nieuwe ontwikkelproduct van ESRI.

In het vervolgonderzoek moet niet alleen ESRI programmatuur gebruikt worden, het zou het onderzoek ten goed komen om ook andere ontwikkelomgevingen te betrekken in het onderzoek.

Referenties

Geschreven literatuur:

Booch, G. 1994. *Object-oriented analysis and design with applications*. The Benjamin/Cummings Publishing Company Inc. Redwood City USA.

Burgerhart, N. Ten Napel D.J. en Postema J. 2002. *Programmeerervaringen met de nieuwe software-lijn van ESRI, FytoGIS in ArcInfo 8 bij Directie IJsselmeergebied*. Geonieuws 1

Van Capelleveen P. Kreft M. en Miltenburg J. 1997. *Handleiding ArcMER, Systeemeisen, - ontwerp en functionele specificaties*. De Bilt. Grontmij Ruimtelijke Inrichting.

de Champeaux D. Douglas, L. en Faure, P. 1993. *Object-Oriented System Development*. Addison-Wesley Publishing Company. Reading USA.

Commandeur, T. en Kreft M. 1998. *Taak4, Implementatie van multicriteria analyse methoden in ArcMER*. Grontmij Advies & Techniek bv, adviesgroep Ruimtelijke ontwikkeling . De Bilt.

ESRI, ArcGIS Desktop help

ESRI, ArcInfo help

ESRI, 1990, *Understanding GIS, The ARC/INFO Method*. ESRI Press. Redlands. USA

ESRI, 1994, *ARC/INFO Data Management, Concepts, data models, database design, and storage*. ESRI Press. Redlands. USA.

ESRI, 2001, 1 White paper, *ArcObjects: The Open Customization Environment Within ArcGIS*. ESRI Press. Redlands. USA.

ESRI, 2001, 2 *Exploring ArcObjects Vol.1-Applications and Cartography* ESRI Press. Redlands. USA.

ESRI, 2001, 2002. White paper, *What is ArcGIS?* ESRI Press. Redlands. USA.

ESRI, 2003, White paper, *ArcGIS8: The Complete Geographic Information System*. ESRI Press. Redlands. USA.

Fitzgerald, T. 2002. *Making the switch from AML to VBA*. ArcUser April-juni 2002. ESRI Press. Redlands. USA.

Knippel, R. 2002 *GIS101: Why migrate to ArcGIS?* GIS News, fall 2002 Dakota County. USA.

Kreft, M. en Vanhaute, L. 1997. *Vlaamse proefprojecten demonstreren waarde van 'kwetsbaarheidsbenadering' Sneller besluiten over wegtracés*. KenMERken September 1997

Grontmij, 1993, 1. *Produktontwikkelingsproject "GIS als analyse-instrument", Hoofdrapport*. Grontmij Advies & Techniek bv Project Bureau Milieu. Zeist.

Grontmij, 1993, 2. *Produktontwikkelingsproject "GIS als analyse-instrument", Bijlage 1 Verzoek om een PMC-budget*. Grontmij Advies & Techniek bv Project Bureau Milieu. Zeist.

Grontmij, 1993, 3. *Produktontwikkelingsproject "GIS als analyse-instrument", Bijlage 3 Resultaten schaduwproject GIS-Haarlemmermeer*. Grontmij Advies & Techniek bv Project Bureau Milieu. Zeist.

Loomis, M. E. S., 1995. *Object databases, The Essentials*. Addison-Wesley Publishing Company. Reading USA.

Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., en Lorensen, W. 1991, *Object-Oriented Modeling and Design*. Prentice-Hall International, Englewood Cliffs, New Jersey.

Tomlin, C.D. 1990, *Geographic systems and cartographic modelling*. Prentice-Hall International, Englewood Cliffs, New Jersey

Watson R.T., 1999, *Data Management (Database and Organizations)*. John Wiley & Sons

Zeiler, M. 1999, *Modeling Our World, The ESRI Guide to Geodatabase Design*. ESRI Press. Redlands. USA.

Internetpagina's:

Crealogic 1999. *Wat is een Electronisch Vergader Systeem (EVS)* www.crealogic.nl/site/page1.html

DSDM Ltd., 2003 *The DSDM tour* www.dsdm.org oktober 2003

Meitner, M.J. 2003, *Models of spatial information* www.ideal.forestry.ubc.ca september 2003

Microsoft 2003, 1 *The Component Object Model* <http://msdn.microsoft.com> september 2003

Microsoft 2003, 2 *Dr. GUI on Components, COM, and ATL*. <http://msdn.microsoft.com/> september 2003

NSIDC, 2003. *Esri Vector Options* http://nsidc.org/noaa/iicwg/pdf/esri_vector_options.pdf September 2003

Santiago Comella-Dorda. 2001, *Component Object Model (COM), DCOM and Related Capabilities*. www.sei.cmu.edu april 2004

Creative Research Systems, 2003 *The Survey System's Tutorial, (chapter the Survey Design)* www.surveysystem.com oktober 2003

Hathaway, R.J. *Object FAQ* versie 2.0, 2003 <http://www.objectfaq.com/oofaq2/> oktober 2003

University of Edinburgh and Association for Geographic Information, 1996 *GIS dictionary* www.geo.ed.ac.uk/agidict/welcome.html april 2004

Walsh, N. 2004, *A Technical introduction to XML*. www.xml.com april 2004

WhatIs.com, 2003 <http://whatis.techtarget.com> april 2004

Gesproken / e-mail:

Eijkelenboom, E. 2003 ESRI Nederland, GIS conferentie 2003, september 2003.

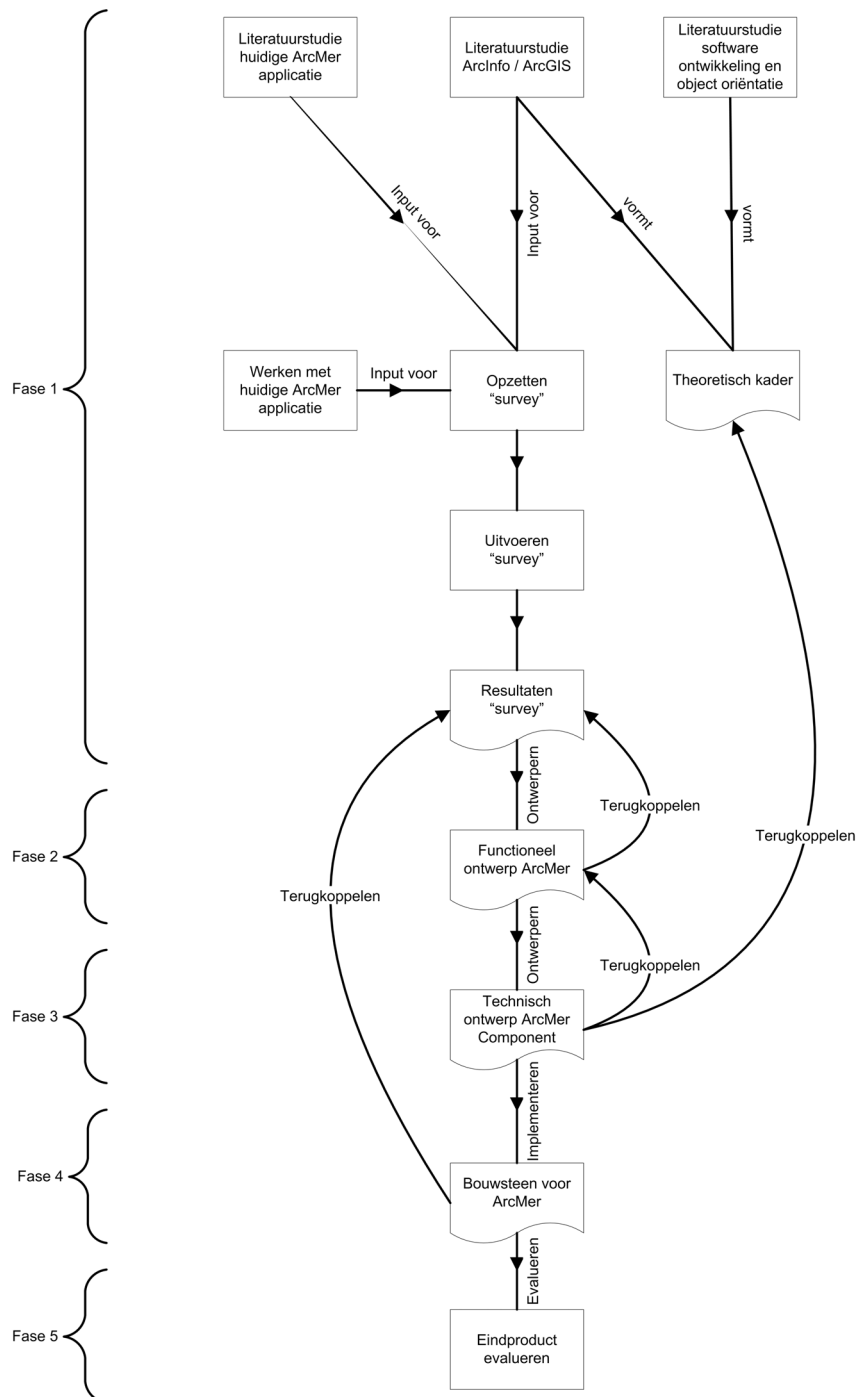
Lanen, R. Van. 2003, 2004 Grontmij Geo-informatie Houten, september 2003 – mei 200

Stuiver, H.J. 2003, 2004 Wageningen Universiteit, september 2003 – mei 2004.

Voets, D. 2004 ESRI Nederland, februari 2004

Bijlage I: Onderzoeksopzet (oorspronkelijk onderzoeksvoorstel)

Het onderzoek bestaat uit een aantal stappen die onder te verdelen zijn in 5 fasen. De stappen, hun onderlinge relaties en de verdeling over de fasen zijn schematisch weergegeven in onderstaande figuur. De eerste 4 fasen komen voort uit een van de hierboven gestelde onderzoeksvragen. Het hele schema moet doorlopen worden om uiteindelijk aan de hoofddoelstelling van dit onderzoek te voldoen en om antwoord te kunnen geven op alle onderzoeksvragen.



Fase 1:

De eerste fase van het onderzoek heeft betrekking op de tweede onderzoeksvraag. Hiervoor wordt begonnen met een literatuurstudie naar het databasebeheer en de programmeermethoden in ArcInfo workstation en ArcGIS. Daarnaast wordt een literatuurstudie uitgevoerd naar de huidige GISMER applicatie. Daarin wordt gekeken naar de functionele en technische eisen die in 1993 aan GISMER werden gesteld en naar het functioneel ontwerp van de huidige GISMER applicatie.

Ook wordt een literatuurstudie uitgevoerd naar een aantal benaderingen van software ontwikkeling en naar object oriëntatie. Deze laatste literatuurstudie vormt samen met de studie naar het databasebeheer en de programmeerwijzen in ArcInfo en ArcGIS het kader waarbinnen dit onderzoek verder plaatsvindt. Vervolgens wordt een “survey” opgezet. Hiervoor wordt om een beter inzicht te krijgen in de mogelijkheden van de huidige GISMER applicatie in een concreet project met de applicatie gewerkt. Dit moet samen met de genoemde literatuurstudies voldoende relevante vragen en aandachtspunten opleveren om in de “survey” aandacht aan te besteden. De “survey” wordt, om een zo goed mogelijk beeld te krijgen van de wensen, gehouden onder zowel GIS specialisten als niet GIS specialisten. Hierbij valt te denken aan huidige en potentiële gebruikers van de applicatie en opdrachtgevers voor het gebruik van de applicatie. Daarna wordt de opgezette “survey” uitgevoerd onder bovengenoemde groepen mensen. Als laatste stap worden in deze fase de resultaten van de “survey” gebundeld om tot een pakket van technische en functionele eisen en wensen voor de nieuwe GISMER applicatie te komen. Het literatuuronderzoek levert ook aandachtspunten op waarop gelet moet worden tijdens fase 4, zodat uiteindelijk ook aan de doelstelling van dit onderzoek voldaan kan worden.

Fase 2:

In figuur 1 is fase 2 weergegeven als het functioneel ontwerp van de applicatie. Het pakket van wensen dat aan het einde van fase 1 is ontstaan wordt vertaald in functies die GISMER moet vervullen. Hoe deze functies onderling samenhangen, gaat visueel weergegeven worden in een stroomdiagram. Bij dit stroomdiagram wordt een nauwkeurige beschrijving gegeven van wat de functies inhouden en wat de samenhang tussen de verschillende functies is. Na het maken van het functioneel ontwerp vindt terugkoppeling plaats naar de resultaten van de “survey” waaraan het ontwerp getoetst wordt.

Fase 3:

Fase 3 is het maken van het technisch ontwerp van een bouwsteen van de nieuwe applicatie. In deze fase wordt bekeken welke processen er tussen de samenhangende functies plaatsvinden. Ook wordt bekeken hoe deze processen en functies technisch geïmplementeerd kunnen worden in ArcObjects met behulp van VB. Ook hierbij wordt een stroomdiagram gemaakt, nu van de technische stappen die genomen moeten worden om uitvoering te geven aan het functioneel ontwerp uit fase 2. Na het maken van het technisch ontwerp vindt terugkoppeling plaats naar het functioneel ontwerp om te kijken of alle functies en relaties opgenomen zijn in het technisch ontwerp. Ook wordt in deze fase bekeken wat nu de impact van de introductie van de Geodatabase en het object georiënteerd denken in applicatieontwikkeling is voor de migratie van GISMER naar ArcGIS.

Fase 4:

Fase 4 bestaat uit de implementatie van het technisch ontwerp. Ook hier wordt verder de impact van de introductie van de Geodatabase en het object georiënteerd denken in applicatieontwikkeling bekeken, maar nu meer vanuit de implementatie van een technisch ontwerp.

Fase 5:

In de laatste fase van het onderzoek wordt het eindproduct geëvalueerd aan de hand van de resultaten van de “survey”. Hierbij wordt bekeken of de applicatie wel aan de gestelde eisen en wensen voldoet. Ook worden er kanttekeningen gemaakt, bijvoorbeeld door de gemaakte aannames te belichten, en worden er aanbevelingen gedaan.

Bijlage II: Contactpersonen 'survey'

Naam	Cor Blanken
Organisatie	Gemeente Zwolle
Functie	Adviseur Milieu specialisme MER / Externe veiligheid
Expertise(s)	MER Locatiekeuzestudies Tracéstudies
Telefoon	038-4984181
E-mail	c.blanken@zwolle.nl
Interview	7 november 2003

Naam	Gerrit-Jan Carsjens
Organisatie	Wageningen Universiteit, Laboratorium voor ruimtelijke planvorming
Functie	Universitair docent
Expertise(s)	Landinrichting MER Locatiekeuzestudies GIS gebruik
Telefoon	0317-484840
E-mail	Gerrit-jan.carsjens@wur.nl
Interview	4 november 2003

Naam	Albert Driesprong
Organisatie	Rijkswaterstaat Adviesdienst Geo-Informatie en ICT, Delft
Functie	Adviseur
Expertise(s)	Geo-informatie (gebruiker en ontwikkelaar)
Telefoon	015 – 2757485 / 0651450544
E-mail	a.driesprong@agi.rws.minvenw.nl
Interview	13 november 2003

Naam	Marja van Eck
Organisatie	Commissie MER
Functie	Werkgroepsecretaris
Expertise(s)	MER Locatiekeuzestudies Tracéstudies MCA
Telefoon	030-2347623
E-mail	Meck@eia.nl
Interview	10 november 2003

Naam	Mark Groen
Organisatie	Grontmij Overijssel, Zwolle
Functie	Teamleider Plan & Proces, Adviseur MER
Expertise(s)	MER Locatiekeuzestudies Tracéstudies
Telefoon	038-4991739 (intern: 1739)

E-mail	Mark.groen@grontmij.nl
Interview	7 november 2003

Naam	Merijn Hansler
Organisatie	Rijkswaterstaat Adviesdienst Geo-Informatie en ICT, Delft
Functie	Junior adviseur
Expertise(s)	Geo-informatie (advies en onderzoek)
Telefoon	015 – 2757061
E-mail	m.b.hansler@agi.rws.minvenw.nl
Interview	13 november 2003

Naam	Robert-Jan Jonker
Organisatie	Grontmij Utrecht, Houten
Functie	Hoofd afdeling Ruimte
Expertise(s)	MER Locatiestudies Tracéstudies
Telefoon	030-6344655 (intern: 4655)
E-mail	Robertjan.jonker@grontmij.nl
Interview	11 november 2003

Naam	Marc Kreft
Organisatie	Grontmij Zuid-Holland, Waddinxveen
Functie	Teamleider / projectmanager
Expertise(s)	MER GIS
Telefoon	0182-625519 (intern: 5519)
E-mail	Marc.kreft@grontmij.nl
Interview	12 november 2003

Naam	Ronald van Lanen
Organisatie	Grontmij Geo Informatie, Houten
Functie	Projectleider GIS-toepassingen
Expertise(s)	GIS-gebruiker GIS-ontwikkelaar
Telefoon	030-6344643 (intern: 4643)
E-mail	Ronald.vanlanen@grontmij.nl
Interview	Gedurende hele traject thesis

Naam	Piet Mossel
Organisatie	Provincie Overijssel, Zwolle
Functie	Beleidsmedewerker ruimtelijk beleid (streekplan, doorwerking rijksbeleid etc.)
Expertise(s)	MER Locatiekeuzestudies Tracéstudies
Telefoon	038-4252525
E-mail	Pcw.mossel@prv-overijssel.nl
Interview	7 november 2003
Naam	Pauline van Veen
Organisatie	Grontmij Utrecht, Houten
Functie	Projectleider MER, coördinator netwerk MER
Expertise(s)	MER Locatiekeuzestudies
Telefoon	030-6394934 (intern: 4934)
E-mail	Pauline.vanveen@grontmij.nl
Interview	3 november 2003

Naam	Marjan Vervoort
Organisatie	Grontmij Overijssel, Zwolle
Functie	Adviseur
Expertise(s)	MER Locatiekeuzestudies Tracéstudies EVS (Elektronisch Vergader Systeem)
Telefoon	038-4991731 (intern: 1731)
E-mail	Marjan.vervoort@grontmij.nl
Interview	7 november 2003

Naam	Marieke Willekens
Organisatie	Grontmij Noord-Brabant, Eindhoven
Functie	Teamleider Groene Ruimte en Water + projectleider MER
Expertise(s)	MER Tracéstudies Locatiestudies
Telefoon	040-2651273 (intern: 1273)
E-mail	Marieke.willekens@grontmij.nl
Interview	5 november 2003

Bijlage III: Vragenlijst m.b.t. de functionele en technische eisen en wensen ten aanzien van GISMER

Contactpersoon: Wouter Rozendaal (stagiair bij Grontmij Geo Informatie)
Tel.: 030-6344674
E-mail: wouter.rozendaal@grontmij.nl
Postadres: Grontmij Geo Informatie Houten
Postbus 119
3990 DC Houten

Doel vragenlijst

GISMER is een uit 1993 daterende gisapplicatie die binnen Grontmij ontwikkeld is. Deze applicatie wordt gebruikt om Milieu Effect Rapportages (MER) en tracé- en locatiekeuzestudies mee te kunnen uitvoeren. De huidige applicatie is gebouwd met ArcInfo AML. In het kader van een afstudeeronderzoek in opdracht van Grontmij Geo Informatie wordt deze applicatie opnieuw ontworpen en ontwikkeld binnen ArcGIS. Deze vragenlijst is bedoeld om inzicht te krijgen in de technische en functionele eisen en wensen die aan de nieuwe GISMER applicatie worden gesteld. Deze gaan dan als basis dienen voor een functioneel en technisch ontwerp.

Opbouw vragenlijst

De vragenlijst bestaat uit 28 vragen verdeeld over 7 delen. De totale tijdsduur die nodig is om alle vragen te kunnen beantwoorden is ongeveer een uur. De meeste delen worden voorafgegaan door een stuk achtergrondinformatie. De meeste vragen bestaan uit de keuze JA / NEE of EENS / ONEENS. De EENS / ONEENS vragen zijn te beantwoorden op een schaal van 1 t/m 5. Juist de toelichtingen bij de antwoorden kunnen veel bruikbare informatie opleveren. Het wordt dan ook erg op prijs gesteld als deze toelichtingen gegeven worden. Indien een vraag niet aansluit bij uw vakkennis, kan dit aan worden gegeven en hoeft de vraag niet beantwoordt te worden. De vragenlijst kan digitaal ingevuld worden, of uitgeprint en vervolgens ingevuld worden. Gelieve de ingevulde vragenlijst voor woensdag 19 november terug sturen via de post of e-mail.

Met alle informatie afkomstig uit deze vragenlijst wordt vertrouwelijk omgegaan. De informatie wordt alleen voor dit afstudeeronderzoek gebruikt. Bij voorbaat dank voor uw tijd en inspanning om deze vragenlijst te beantwoorden en terug te sturen.

*1 doorhalen of weghalen wat niet van toepassing is

*2 omcirkelen of markeren op de schaal 1 t/m 5

Start vragenlijst

Kwetsbaarheidbenadering

De impact van een ruimtelijke maatregel op het milieu kan op twee manieren beoordeeld worden; vanuit de maatregel of vanuit het studiegebied. Wanneer vanuit de maatregel geredeneerd wordt, wordt een effectgerichte benadering gebruikt, als vanuit het gebied geredeneerd wordt, kan de kwetsbaarheidbenadering gebruikt worden.

Bij het gebruik van GISMER wordt geredeneerd vanuit gebiedskenmerken. Daarbij wordt gekeken naar de huidige geschiktheid en kwetsbaarheid van het studiegebied voor de geplande ruimtelijke maatregel. Voor verschillende thema's zoals natuur en landschap, verkeer en vervoer, economie enz. worden kwetsbaarheidanalyses uitgevoerd en kwetsbaarheidkaarten gegenereerd. Door deze kwetsbaarheidkaarten te combineren ontstaat er een integrale kwetsbaarheidkaart. Daarop staan de meest kwetsbare gebieden voor de te nemen ruimtelijke maatregel. Door het toepassen van de kwetsbaarheidbenadering worden op voorhand dus geen mogelijkheden uitgesloten.

De kwetsbaarheidbenadering vind ik GOED / MATIG / SLECHT ^{*1} geschikt voor het uitvoeren van een milieu effect rapportage.

Toelichting (a.u.b. ook toelichten welk type MER wel/niet):

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

De kwetsbaarheidbenadering vind ik GOED / MATIG / SLECHT ^{*1} geschikt voor het uitvoeren van een locatiekeuzestudie.

Toelichting:

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

De kwetsbaarheidbenadering vind ik GOED / MATIG / SLECHT ^{*1} geschikt voor het uitvoeren van een tracéstudie.

Toelichting:

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

De effectgerichte benadering vind ik GOED / MATIG / SLECHT ^{*1} geschikt voor het uitvoeren van een milieu effect rapportage.

Toelichting (a.u.b. ook toelichten welk type MER wel/niet):

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

De effectgerichte benadering vind ik GOED / MATIG / SLECHT ^{*1} geschikt voor het uitvoeren van een locatiekeuzestudie.

Toelichting:

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

De effectgerichte benadering vind ik GOED / MATIG / SLECHT ^{*1} geschikt voor het uitvoeren van een tracéstudie.

Toelichting:

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

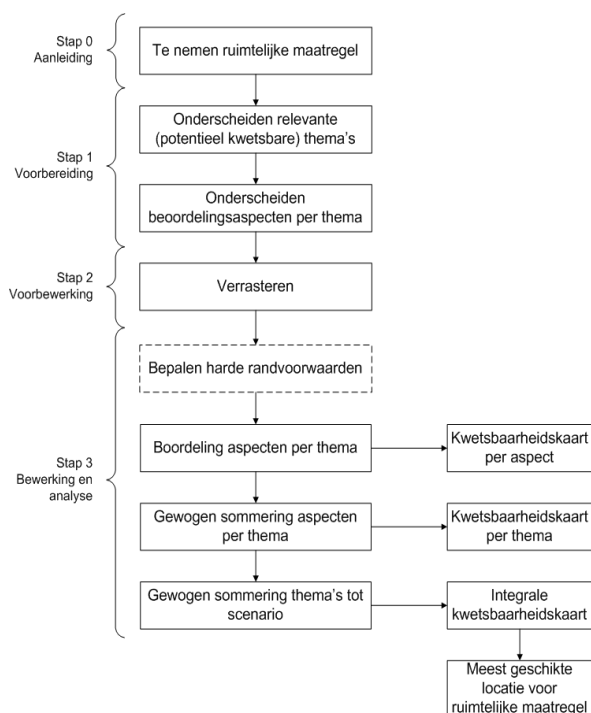
Als alle drie bovengenoemde toepassingen binnen een methode gevangen moeten worden, dan heeft de EFFECTGERICHTE / KWETSBAARHEIDS ^{*1} benadering de voorkeur.

Toelichting:

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

Werking GISMER versie 1993

Met de bestaande applicatie GISMER kunnen gebieden op eenduidige wijze beoordeeld worden op hun geschiktheid en kwetsbaarheid voor bepaalde typen van ruimtegebruik. Naast het ondersteunen van milieu effect rapportages wordt GISMER ook veel gebruikt bij het uitvoeren van tracé- en locatiekeuzestudies. Door gebruik te maken van de kwetsbaarheidbenadering wordt voor het gehele gebied de relatieve kwetsbaarheid bepaald voor een bepaalde ruimtelijke functie. De methode die gevolgd moet worden om het hele GISMER proces te doorlopen bestaat uit drie stappen die zijn weergegeven in de figuur hiernaast.



Stap 1: Voorbereiding

Bij de start van een MER wordt door de opdrachtgever bepaald welke thema's zoals ecologie, landschap, wonen enz. er onderzocht moeten worden.

Vervolgens wordt voor deze thema's data verzameld. Al deze data moet bewerkt en tot aspecten samengevoegd worden. Een aspect heeft dus vaak meerdere databronnen. Meerdere aspecten vormen samen een thema. Voor het thema landschap zouden er bijvoorbeeld de aspecten cultuurhistorische waarden, archeologische waarden enz. kunnen zijn. De gegevensvoorbereiding gebeurt volledig handmatig en is dus niet geautomatiseerd in GISMER.

Stappen in een GISMER proces

	EENS	ONEENS ^{*2}				
De datavoorbereiding moet zoveel mogelijk gestandaardiseerd zijn.	1	2	3	4	5	
Toelichting:						

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

De datavoorbereiding moet binnen GISMER plaatsvinden.

EENS	ONEENS ^{*2}				
	1	2	3	4	5

Toelichting:

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

Stap 2: Voorbewerking

Als de aspecten eenmaal verzameld zijn, moeten ze verder voorbewerkt worden zodat de analyses in GISMER uitgevoerd kunnen worden. Hiervoor moeten alle bestanden in een uniform dataformaat (ArcInfo Grid met eenduidige celgrootte) komen waarin alle bewerkingen gedaan kunnen worden. Deze gegevensvoorbewerking is in GISMER geautomatiseerd en wordt verrasteren genoemd.

De standaard datastructuur in GISMER moet een GRID / VECTOR ^{*1} datastructuur zijn.

Toelichting:

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

Stap 3: Bewerking en analyse

Harde randvoorwaarden

Binnen GISMER is er de extra optie om harde randvoorwaarden op te nemen. Dit zijn gebieden die bij de verdere beoordeling niet meer mee worden genomen en uitgesloten worden als potentiële locatie voor de geplande vorm van ruimtegebruik. Dit definiëren van de harde randvoorwaarden moet gebeuren voordat er met de beoordeling van de aspecten begonnen kan worden. Dit moet eerst per aspect uitgevoerd worden, waarna ze allemaal gecombineerd worden.

Gezien vanuit de gehanteerde methodiek. Harde randvoorwaarden moeten VOOR HET BEOORDELEN / NA HET BEOORDELEN / MAAKT NIETS UIT bepaald worden. ^{*1}

Toelichting:

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

Gezien vanuit GIS-technisch oogpunt. Harde randvoorwaarden moeten VOOR HET BEOORDELEN / NA HET BEOORDELEN / MAAKT NIETS UIT bepaald worden. ^{*1}

Toelichting:

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

Aspecten

Wanneer de harde randvoorwaarden gedefinieerd zijn kunnen de aspecten beoordeeld worden. Binnen GISMER zijn verschillende beoordelingsmethoden toepasbaar⁽¹⁾. Alle waarden van het aspect, bijvoorbeeld de afstand tot bebouwd gebied, krijgen een score. Als resultaat wordt er per aspect een kwetsbaarheidkaart gemaakt.

Thema's

Bij het definiëren van thema's worden alle aspecten die tot een thema behoren samengevoegd. Een thema is bijvoorbeeld natuur en landschap, of wonen en bedrijvigheid. Hiervoor worden de waarden van de aspecten d.m.v. de MCA-methode gewogen sommering⁽¹¹⁾ bij elkaar opgeteld. Er kunnen dus gewichten aan de aspecten worden gekoppeld. Als resultaat worden per thema kwetsbaarheidkaarten gemaakt.

Visie

Een visie wordt op een gelijke manier opgebouwd als de thema kwetsbaarheidkaarten. Nu worden de thema's via een gewogen sommering samengevoegd. Op deze wijze kan een bepaalde visie in de integrale kwetsbaarheid opgenomen worden. Het eindresultaat van de analyse in GISMER is de integrale kwetsbaarheidkaart met daarop aangegeven de mate waarin ieder gebied kwetsbaar is voor de te nemen ruimtelijke maatregel.

De opbouw van een integrale kwetsbaarheidkaart, via de drie niveaus, aspecten, thema's, visie is een goede methode. EENS / ONEENS ^{*1} want,

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

De term visie kan beter vervangen worden door scenario. EENS / ONEENS ^{*1} want,

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

⁽¹⁾ Beoordelingsmethode

De scores die aan de aspectwaarden toegekend worden liggen in principe tussen 1 en 99. Lage scores stellen een lage kwetsbaarheid voor en hoge scores een hoge kwetsbaarheid. De scores 0 en 100 hebben een speciale betekenis. Gebieden die niet relevant zijn om te waarderen vanuit een bepaald aspect krijgen de score 0 voor dit aspect. De score 100 betekent dat de gebieden altijd deze score behouden, ook na de integratie met andere beoordeelde aspecten. Aspecten die niet voor het hele gebied gelden, wanneer er bijvoorbeeld sprake is van punt of lijnobjecten, krijgen op de locaties zonder data de score 0. Dit zijn de zogenaamde niet gebiedsdekkende aspecten. Ook aspecten waar onvoldoende data voor is, maar die eigenlijk wel voor het hele gebied zouden moeten gelden, krijgen op de plaatsen zonder data de score 0. Scores kunnen met verschillende methodes toegekend worden, absoluut, in klassen, gradueel, logaritmisch of exponentieel

	EENS	ONEENS ^{*2}			
De scores 0 en 100 worden op een praktische manier toegepast	1	2	3	4	5
Toelichting:					

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

Niet gebiedsdekkende aspecten moeten buiten beschouwing gelaten worden.
EENS / ONEENS ^{*1} want,

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

Aspecten met onvolledige data moeten buiten beschouwing gelaten worden.
EENS / ONEENS want,

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

Welke van de volgende beoordelingsmethodes moeten toepasbaar zijn? ^{*1}

Absoluut	JA / NEE
In klassen	JA / NEE
Gradueel	JA / NEE
Logaritmisch	JA / NEE
Exponentieel	JA / NEE
Anders nl.	

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

(II) MCA-methoden

In GISMER versie 1993 kan alleen gebruik gemaakt worden van de Multi Criteria Analyse methode gewogen somming. Bij deze methode wordt aan ieder aspect een gewicht gekoppeld. Daarna worden de scores van het aspect vermenigvuldigd met het bijbehorende gewicht. Vervolgens worden de aspecten per thema opgeteld. Dezelfde stappen worden nog een keer herhaald, maar nu voor de thema's die gecombineerd worden tot een visie. Een mogelijke tweede MCA-methode is de verwachtingswaarde methode. Deze methode maakt ook gebruik van een gewogen somming, maar heeft de mogelijkheid kwalitatieve waarden mee te nemen in de beoordeling. Gewichten worden berekend op basis van het aantal aspecten en op basis van de rangorde van de aspecten.

- | |
|--|
| 1. De gewogen somming methode biedt voldoende functionaliteit om GISMER als een betrouwbaar hulpmiddel te gebruiken bij MER, locatiekeuze en tracé keuzestudies. |
|--|

EENS / ONEENS want,

*Deze vraag sluit WEL / NIET ^{*I} aan bij mijn vakkennis.*

- | |
|--|
| 2. De verwachtingswaarde methode biedt VEEL / WEINIG / GEEN extra functionaliteit. |
|--|

Toelichting:

*Deze vraag sluit WEL / NIET ^{*I} aan bij mijn vakkennis.*

ArcGIS

ArcGIS 8 is de nieuwe generatie GIS software van ESRI. ArcGIS bestaat uit een aantal losse componenten die onderling een hiërarchische structuur hebben. Gebruikers kiezen het pakket dat de functionaliteit biedt die zij nodig hebben. Wanneer de eisen van de gebruiker toenemen, kan er altijd een upgrade plaatsvinden. De nieuwe GISMER applicatie gaat ontwikkeld worden binnen ArcGIS. Het is mogelijk om nieuwe software componenten aan ArcGIS toe te voegen waarbij gebruik wordt gemaakt van standaardfunctionaliteit van ArcGIS.

	WEINIG	VEEL ^{*2}			
	1	2	3	4	5
Wat is de ArcGIS kennis van potentiële gebruikers binnen uw organisatie?					
Toelichting:					

*Deze vraag sluit WEL / NIET^{*1} aan bij mijn vakkennis.*

	LAAG			HOOG ^{*2}	
	1	2	3	4	5
Hoe is de bereidheid over te schakelen van Arcview 3.x of een ander pakket naar ArcGIS?					
Toelichting:					

*Deze vraag sluit WEL / NIET^{*1} aan bij mijn vakkennis.*

3. Is er behoefte bij de organisatie waar u voor werkt aan een applicatie als GISMER?

JA / NEE^{*1} want,

4. Aan welke eisen moet gebruikersinterface van de nieuwe GISMER applicatie voldoen?

Overig

5. Welke andere mogelijke toepassingen, zoals een waterkansenkaart, van een applicatie als GISMER zijn er?

*Deze vraag sluit WEL / NIET ^{*1} aan bij mijn vakkennis.*

6. Verdere opmerkingen en suggesties

Persoonlijke vragen

7. Wat is uw naam en voor welke organisatie werkt u?

Naam:

Organisatie:

8. Wat is uw huidige functie binnen de organisatie waar u voor werkt?

Wat zijn uw expertises met betrekking tot het onderwerp van deze vragenlijst? ^{*1}

Milieu Effect Rapportage JA / NEE

Locatiekeuzestudies JA / NEE

Tracéstudies JA / NEE

GIS gebruiker JA / NEE

GIS ontwikkelaar JA / NEE

Anders nl.

Hartelijk dank voor uw tijd en inspanning bij het invullen en terugsturen van deze vragenlijst. Indien u nog verdere informatie wilt of vragen heeft over het onderwerp van deze vragenlijst, neem dan alstublieft contact op. Mogelijkheden om in contact te komen zijn aan het begin van deze vragenlijst te vinden.

Bijlage IV: Object oriëntatie en software ontwikkeling

Object oriëntatie

Object oriëntatie (OO) is een manier van denken waarbij gebruik wordt gemaakt van modellen die opgebouwd zijn rondom gebeurtenissen. In het geval van software ontwikkeling betekent object oriëntatie dat software georganiseerd is als discrete objecten die zowel datastructuur als gedrag omvatten. Dit in tegenstelling tot traditionele software ontwikkeling waarbij de datastructuur los van de operaties staat. Object oriëntatie heeft in hoofdzaak vier aspecten, identiteit, classificatie, polymorfisme en erfelijkheid (Rumbaugh *et. al.*, 1991). Deze, en het gebruik van het begrip object in dit rapport worden hieronder kort toegelicht.

Basisprincipes object oriëntatie

Identiteit

Identiteit houdt in dat objecten worden onderscheiden door hun eigen bestaan en niet door de beschrijvende eigenschappen die ze eventueel hebben. Twee objecten zijn dus toch anders ook al hebben ze exact dezelfde attribuutwaarden.

Classificatie

Classificatie betekent dat objecten met dezelfde datastructuur (attributen) en gedrag (operaties) gegroepeerd zijn in klassen. Een klasse beschrijft de eigenschappen die belangrijk zijn voor een applicatie en negeert de overige eigenschappen. Een object is een specifieke toestand van een klasse.

Polymorfisme

Polymorfisme is het anders reageren van een zelfde operatie op verschillende klassen. Een specifieke implementatie van een operatie door een bepaalde klasse wordt een methode genoemd.

Overerven

Overerven is het delen van attributen en operaties door verschillende klassen op basis van een hiërarchische relatie. Klassen kunnen ruw gedefinieerd worden en verder verfijnd worden in subklassen. Iedere subklasse erft de eigenschappen van zijn superklasse en voegt daar zijn eigen karakteristieken aan toe.

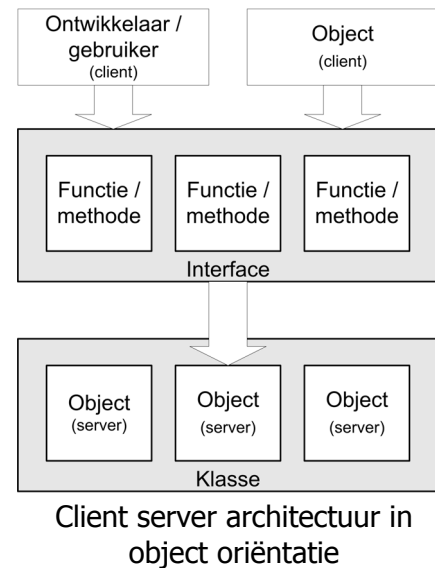
Binnen de objectbenadering is het object een belangrijk begrip, het is dan ook van groot belang het gebruik van dit begrip in dit rapport toe te lichten. Binnen software ontwikkeling is onderscheid te maken tussen software objecten en data objecten. Een software object is een stuk applicatiecode inclusief datastructuur die een service verleend aan de rest van de applicatie (<http://msdn.microsoft.com/>, 2003, 1; Loomis 1995). Data objecten vertegenwoordigen entiteiten en een data objectklasse is een verzameling entiteiten met dezelfde beschrijvende eigenschappen.

Het Component Object Model

Het Component Object Model (COM) is een platform onafhankelijk object georiënteerd systeem om software componenten die met elkaar samen kunnen werken te creëren (http://msdn.microsoft.com). COM bepaald de programmeerbenodigheden om objecten te creëren die met elkaar in interactie kunnen (http://msdn.microsoft.com, 1993, 2). COM laat de programmeertaal, de structuur van een applicatie en de verdere details over aan de programmeur. De enige eis die gesteld wordt is dat de programmeertaal die gebruikt wordt, COM ondersteund.

COM heeft een client/server architectuur, de server levert bepaalde functionaliteit die de client gebruikt. (ESRI, 2001, 2). De client kan een ontwikkelaar of gebruiker zijn, maar ook een ander object of meerdere objecten. Een object kan tegelijkertijd client en server zijn, doordat dit object wordt aangeroepen en op zijn beurt weer een ander object aanroept. Clients en servers kunnen bij elkaar in hetzelfde proces of in verschillende processen voorkomen. In het eerste geval worden ze beide verpakt in een Dynamic Link Library (DLL) en in het tweede geval als executables (EXE).

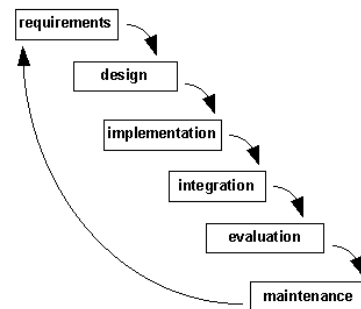
Ontwikkelen aan de hand van COM betekend gebruik maken van interfaces. Een interface is een groep aan elkaar gerelateerde functies waarmee objecten bewerkt kunnen worden. Interfaces zijn niet specifiek geprogrammeerd voor een bepaalde objectklasse, maar zijn toepasbaar op verschillende klassen met objecten. De interface zegt wat een object kan doen en een klasse zegt hoe dit gedaan moet worden (ESRI, 2001, 2). De aanbieder van functionaliteit, een object, is voor een client alleen toegankelijk via de interface zoals in de hiernaast afgebeelde figuur te zien is.



Software ontwikkeling

Het ontwikkelen van software kan op verschillende manieren plaatsvinden. Een van de methoden is de traditionele watervalmethode. Dit is een statische methode die stap voor stap doorlopen moet worden. Bij deze methode wordt een applicatie eerst opgespitst in een aantal losse componenten, ontwerp en implementatie vinden vervolgens per component plaats. Veranderingen boven aan de waterval zijn in een later stadium moeilijk door te voeren en te implementeren. Wel is de watervalmethode een overzichtelijke methode voor software ontwikkeling.

Ontwikkelingen in een later stadium van applicatieontwikkeling kunnen veranderingen met zich mee brengen ten opzichte van eerder genomen beslissingen. Dit is bij het gebruik van het standaard watervalmodel niet mogelijk. Een proces waarin verder uit wordt gegaan van het waterval model maar waarin het wel mogelijk is terug te keren naar een eerdere fase, wordt door de Champeaux *et al.* aangeduid met de fontein metafoor.



Een andere methode is de ‘dynamic systems development method’ (DSDM). Deze methode is er op gericht om op het juiste moment oplossingen af te leveren aan de opdrachtgever die voldoen aan de verwachtingen. De verwachtingen worden bepaald aan de hand van het MoSCoW principe. DSDM is een iteratief en imperatief proces dat gebruik maakt van snelle ontwikkeling door middel van prototypes (www.dsdm.org).

Must have (essentieel voor het project succes)

O

Should have (belangrijk, maar het project succes hangt er niet van af)

Could have (kunnen makkelijk weg worden gelaten zonder van invloed te zijn op het project)

O

Won’t have (kunnen deze keer weg worden gelaten en op een later tijdstip uitgevoerd worden)

Het MoSCoW principe

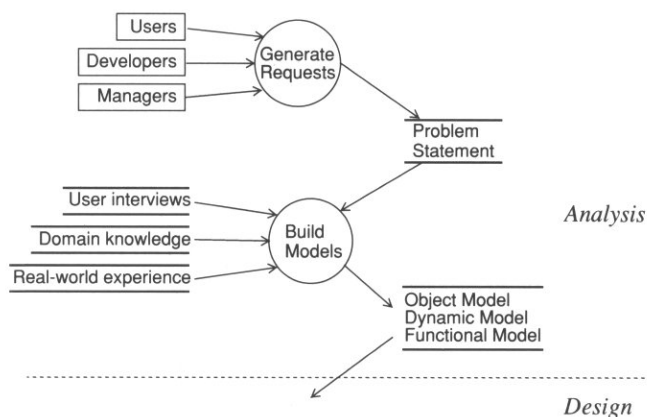
Object georiënteerde software ontwikkeling

De object georiënteerde benadering wordt in het hele traject van applicatieontwikkeling toegepast (<http://www.objectfaq.com/oofaq2>, 2003). Object oriëntatie heeft een totaal aan methoden en technieken voor analyse, ontwerp en implementatie in zich. Het klassieke onderscheid tussen analyse, ontwerp en implementatie is aan het vervagen. Objecten die gedefinieerd zijn in de analysefase werken door tot in de implementatiefase (de Champeaux *et al.*). In de literatuur worden verschillende methoden gegeven voor het proces van object georiënteerde applicatieontwikkeling, zie bijvoorbeeld Booch, 1994; de Champeaux *et al.* 1993 en Rumbaugh *et al.* 1991. In hoofdzaak worden bij deze methoden dezelfde werkwijzen en handelingen gehanteerd en verschilt alleen de gebruikte terminologie. Een van de veel gebruikte methoden is de door Rumbaugh *et al.* omschreven Object Modeling Technique. In dit rapport wordt de daarbij behorende terminologie gehanteerd. Wanneer een applicatie object georiënteerd ontwikkeld wordt, wordt een model van een toepassingsgebied gebouwd waarbij gedurende het ontwerp van het systeem implementatie details worden toegevoegd. De ontwikkeling bestaat uit vier fases, analyse, systeem ontwerp, object ontwerp en implementatie.

Fase 1: Analyse

De eerste fase in object modelleren is de analysefase. Analyse begint wanneer er een verzoek van een opdrachtgever of ontwikkelaar is. Dit verzoek mond vervolgens uit in een probleemstelling. Het is niet nodig dat deze probleemstelling vanaf het begin compleet en definitief is. Het is meer een basis voor het uitdiepen van de vereisten die aan het systeem worden gesteld (Rumbaugh *et al.* 1991).

Na het opstellen van de probleemstelling moet het systeem waarin het probleem of vraagstuk speelt verkend worden. De eerste stap hierin is het opzetten van een object model. Informatie voor dit model moet van experts uit het toepassingsgebied komen. Dit kan bijvoorbeeld door middel van het uitvoeren van interviews. Het object model is een statische weergave van het systeem waarin het probleem of vraagstuk zich voordoet. Vervolgens moeten er een dynamisch en een functioneel model opgesteld worden. Het dynamisch model geeft het tijdsafhankelijke gedrag van een systeem en haar objecten weer. Het functioneel model laat zien hoe waarden berekend worden. Het geeft weer welke waarden van elkaar afhangen en welke functies en relaties tussen deze waarden aanwezig zijn.



De analysefase. Het proces van applicatie ontwikkeling begint met een verzoek of wens van de gebruiker of opdrachtgever. Uit dit verzoek wordt een probleemstelling opgesteld. Vervolgens wordt deze probleemstelling verder uitgediept zodat verschillende modellen gebouwd kunnen worden.

Object model

De eerste stap bij het maken van een object model is het ontrafelen van het systeem. Hierbij worden alle relevante objectklassen bepaald en benoemd. Het doel van deze eerste stap is het bepalen van de grenzen van het systeem of de applicatie (Booch, 1994). De objectklassen zijn alle relevante zaken die met het probleem of vraagstuk te maken hebben. Dit kunnen zowel fysieke entiteiten zijn als niet fysieke elementen. Vervolgens moet in deze verzameling objectklassen een selectie gemaakt worden, waarbij de onnodige en niet correcte klassen verwijderd worden. Het resultaat van deze stap is een datawoordenboek met daarin een omschrijving van alle relevante objecten en objectklassen. De volgende stap in object modelleren is het onderscheiden van alle relaties en verbindingen tussen de klassen. Ook hier moet een selectie plaatsvinden. Na het onderscheiden van de objectklassen en de relaties hiertussen, kunnen de attributen bij de objectklassen onderscheiden worden. Attributen zijn de eigenschappen van individuele objecten. Ook attributen kunnen overbodig of foutief zijn, deze moeten dus verwijderd worden. Criteria op basis waarvan klassen, verbindingen en attributen verwijderd moeten worden, worden uitgebreid besproken in Rumbaugh *et al.* 1991: 153-155, 158-160, 162-163.

Wanneer alle objectklassen, relaties en attributen benoemd en helder omschreven zijn, moeten klassen geordend worden met behulp van overerving. Dit overerven kan in twee richtingen plaatsvinden; door het generaliseren van gemeenschappelijke aspecten van bestaande klassen of door het verfijnen van klassen in specifieke subklassen.

Uiteindelijk kunnen klassen die gezamenlijk een logisch onderdeel van het totale model vormen gegroepeerd worden in modules.

Een object model is vrijwel nooit correct nadat al deze stappen eenmaal doorlopen zijn. Daarom moeten deze stappen van het benoemen en uitsluiten van objecten, relaties en attributen tijdens het verdere proces herzien en nog een aantal keren doorlopen worden.

Dynamisch model

Het dynamisch model laat het tijdsafhankelijk gedrag van het systeem of model en de objecten daarbinnen zien. Om tot een dynamisch model te komen moet er gezocht worden naar gebeurtenissen en externe factoren die van invloed zijn. Ook zijn de reacties van het systeem of model op deze gebeurtenissen en externe factoren van belang. Iets is een gebeurtenis wanneer informatie uitgewisseld wordt tussen een object in het systeem en de buitenwereld. Alle gebeurtenissen die een object bereiken of die een object uitzendt moeten in het dynamisch model op worden genomen. Het dynamisch model is niet van groot belang voor puur statische opslag van data, wel voor interactieve systemen.

Functioneel model

In het functioneel model worden de functionele afhankelijkheden tussen attributen weergegeven. Dit wordt vaak weergegeven in een data stroomdiagram. De objecten uit het object model vormen de data (entiteiten) en de processen in het functioneel model corresponderen met de gebeurtenissen uit het dynamisch model.

Bij het opstellen van het functioneel model moet eerst de gewenste uitvoer data en de invoer data onderscheiden worden. Hierna kan het data stroomdiagram opgesteld worden. Dit kan het beste achterstevoren plaatsvinden, waarbij er dus van uitvoer terug geredeneerd wordt naar de invoer (Rumbaugh *et al.* 1991). Nadat het data stroomdiagram voltooid is, moeten de functies helder omschreven worden, hierbij gaat het om wat de functie moet doen, niet hoe deze geïmplementeerd moet worden. Vervolgens moeten de beperkingen tussen de objecten onderscheiden worden. Deze beperkingen zijn de functionele afhankelijkheden tussen objecten die niet aan elkaar gerelateerd zijn door middel van een invoer-uitvoer afhankelijkheid. Dit kunnen bijvoorbeeld voorwaarden zijn die een bepaald object stelt aan een ander object. Tot slot moeten er criteria opgesteld worden om de applicatie zo optimaal mogelijk te maken. Een voorbeeld is het maximaliseren van de transparantie van de applicatie.

Fase 2: Systeem ontwerp

In de tweede fase van applicatie ontwikkeling moet bepaald worden hoe tot het technisch ontwerp gekomen moet worden. In de analysefase werd bekeken wat er door de applicatie uitgevoerd moet worden. Bij het ontwerpen van de applicatie wordt gekeken naar hoe dit dan gedaan moet worden. Dit moet eerst in hoofdlijnen gebeuren zonder in details te treden. Het ontwerpen in hoofdlijnen vindt plaats in de systeem ontwerpfase.

De eerste stap in systeem ontwerp is het opdelen van het systeem in een aantal componenten. Een component is geen object of functie van de applicatie, maar is een verzameling klassen, relaties, operaties, gebeurtenissen en beperkingen die een onderlinge relatie hebben en een goed gedefinieerde, zo klein mogelijke interface hebben met andere subsystemen (Rumbaugh *et al.* 1991). De meeste interacties tussen objecten moeten binnen een subsysteem plaatsvinden.

De onderlinge relaties tussen subsystemen zijn weer te geven als lagen of als partities. Lagen delen een systeem horizontaal in stukken. Tussen lagen bestaat een client/server relatie. De onderste laag (server) verleent een functie aan de bovenste laag (client). Een laag weet alleen van het bestaan van de laag eronder af en niet wat er naar boven toe afspeelt. Een gelaagd ontwerp kan open of gesloten zijn. Bij een open ontwerp kan een laag gebruik maken van alle onderliggende lagen en bij een gesloten ontwerp alleen van de laag die direct onder de betreffende laag zelf zit. Partities delen een systeem verticaal in subsystemen. Deze subsystemen verlenen vaak een soort service en hebben verder weinig interactie met andere subsystemen.

In deze fase moet bepaald worden op welke manier data beheerd wordt binnen de applicatie. Er moet de keuze gemaakt worden of de opslag bijvoorbeeld plaats gaat vinden in losse files of dat een database gebruikt gaat worden. Beide hebben voor- en nadelen welke per systeem of te ontwikkelen applicatie tegen elkaar afgewogen moeten worden.

Door applicaties kan op een aantal manieren gecommuniceerd worden met de gebruiker. Welke methode gebruikt wordt is van belang voor het verdere ontwerp van de applicatie en de implementatie daarvan. Twee van de mogelijke methoden zijn de batchtransformatie en de interactieve interface. Bij batch transformatie wordt een grote hoeveelheid data eenmalig getransformeerd. Interactie met de gebruiker vindt alleen plaats bij het invoeren van de data en het bepalen van de transformatiemethode. De verdere bewerking wordt uitgevoerd zonder enige interactie. Bij een interactieve interface vindt veel interacties plaats tussen het systeem en de gebruiker. De belangrijkste aandachtspunten bij het ontwerpen van een interactieve interface zijn de communicatie tussen applicatie en gebruiker, de presentatie van de uitvoer, en de begrijpbaarheid van de gebruikersinterface.

Fase 3: Object ontwerp

Het maken van het object ontwerp is een proces van verfijning en detail toevoegen. Iedere actie die in het functioneel ontwerp gedefinieerd is, moet in het object model geformuleerd worden als algoritme. Het functioneel ontwerp geeft aan wat er uitgevoerd moet worden en de algoritmen geven aan hoe iets uitgevoerd moet worden. De ontwerper van algoritmen heeft vier taken. Ten eerste moet hij de meest geschikte algoritmen kiezen, ten tweede datastructuren bijpassend bij de algoritmen kiezen, ten derde indien nodig nieuwe interne klassen definiëren en tenslotte “verantwoordelijkheid” voor operaties toekennen aan klassen.

Algoritmen moeten zo simpel mogelijk gehouden worden. Aandachtspunten bij het ontwikkelen en kiezen van een algoritme zijn de complexiteit van de berekeningen, het gemak van implementatie/begrijpbaarheid, flexibiliteit en de fijnafstemming van het object model.

De datastructuren die gekozen worden, voegen geen informatie toe aan het functioneel ontwerp. De datastructuren organiseren de data alleen in een vorm zodat het makkelijk te gebruiken is in de algoritmen.

Bij het verder uiteen rafelen van de acties uit het functioneel ontwerp kan het zijn dat nieuwe objectklassen gedefinieerd moeten worden om tussentijdse resultaten vast te leggen. Vaak zijn objecten die lager in hiërarchie staan niet zichtbaar aan de buitenkant van een applicatie.

Operaties kunnen op een of meerdere objecten van toepassing zijn of op verschillende plaatsen in een algoritme toe te passen zijn. De operatie moet aan een klasse toegewezen worden. Over het algemeen kan de operatie het beste toegewezen worden aan de doelklasse van de operatie.

Fase 4: Implementatie

De laatste fase in applicatieontwikkeling is de implementatie. Implementeren, of het schrijven van programmacode is in feite een uitbreiding van het ontwerp. Het ontwerp moet vertaald worden in de programmeertaal die voor de implementatie gekozen is. Wezenlijke beslissingen moeten niet meer gemaakt worden.

Bijlage V: Objectklassen en attributen functioneel ontwerp GISMER

Relevante objectklassen per component

Relevante objectklassen component 1; “projectinstellingen”	
Tabel “Project”	Tabel waarin de project-ID en de projectnaam op worden geslagen.

Relevante objectklassen component 2; “importeren basisbestanden”	
Basisbestand	Dataset die als invoer voor GISMER gebruikt wordt. Dit kan een vector of raster dataset zijn zijn.
Dataset in Geodatabase	Het in de Geodatabase geïmporteerde basisbestand. De bestandsstructuur (vector of raster) is gelijk aan die van het basisbestand. De dataset kan als basis dienen voor een aspect en voor een SHRV.
Metadata	De beschrijvende data van het basisbestand. Deze wordt opgeslagen in de Geodatabase in XML formaat

Relevante objectklassen component 3; “creëren aspecten”	
Dataset in Geodatabase	Het in de Geodatabase geïmporteerde basisbestand. Van deze dataset wordt een aspect gecreëerd.
Plangrens	Dataset met de grootte en vorm van de plangrens.
Aspect	Entiteit die van invloed is op de geschiktheid of kwetsbaarheid van een gebied voor een bepaalde ruimtelijke functie of ingreep.
Tabel “aspect”	Tabel waarin alle beslissingen en bewerkingen die rechtstreeks verbonden zijn met een basisbestand worden opgeslagen.
Tabel “vergridmethoden”	Look-up tabel waaruit de mogelijke verrastermethodes gekozen moeten worden. (De keuze wordt gemaakt in de menustructuur van GISMER.)

Relevante objectklassen component 4; “beoordelen aspecten”	
Aspect	Entiteit die van invloed is op de geschiktheid of kwetsbaarheid van een gebied voor een bepaalde ruimtelijke functie of ingreep.
Beoordeeld aspect	Een ten aanzien van geschiktheid of kwetsbaarheid voor een bepaalde ruimtelijke functie of ingreep beoordeeld aspect.
Tabel “beoordeling”	Tabel waarin de oorspronkelijke aspectwaarden en de gegeven scores bij het beoordelen van aspecten worden opgeslagen.
Tabel “aspect”	Tabel waarin alle beslissingen en bewerkingen die rechtstreeks verbonden zijn aan een aspect worden opgeslagen.
Tabel “beoordeeld aspect”	Tabel met daarin een verzameling van alle beoordeelde aspecten en de kenmerken van deze aspecten.
Tabel “beoordelingsmethode”	Tabel met daarin de mogelijke methoden voor het beoordelen van aspecten.

Relevante objectklassen component 5; “samenvoegen beoordeelde aspecten”	
Beoordeeld aspect	Een ten aanzien van geschiktheid of kwetsbaarheid voor een bepaalde ruimtelijke functie of ingreep beoordeelde entiteit.
Thema	Een gewogen samenvoeging van beoordeelde aspecten die aan elkaar gerelateerd zijn.
Tabel “thema”	Tabel waarin per thema alle kenmerken worden opgeslagen.
Tabel “thema samenstellen”	Tabel waarin de aspecten en de gewichten die hier bij het samenvoegen tot een thema aan zijn gegeven, worden opgenomen.

Relevante objectklassen component 6; “samenvoegen thema’s”	
Thema	Een gewogen samenvoeging van beoordeelde aspecten die aan elkaar gerelateerd zijn.
Integrale KW-raster	Een gewogen samenvoeging van thema’s.
Tabel “integrale KW-raster”	Tabel waarin per integrale KW-raster alle kenmerken worden opgeslagen.
Tabel “integrale KW-raster samenstellen”	Tabel waarin de thema’s en de gewichten die hier bij het samenvoegen tot een visie aan zijn gegeven, worden opgenomen

Relevante objectklassen component 7; “samenstellen SHRV”	
Dataset(s) in Geodatabase	De in de Geodatabase geïmporteerde basisbestand(en). Uit deze dataset(s) worden selecties gemaakt welke gebieden tot een SHRV moeten behoren.
Tabel “SHRV”	Tabel waarin per gebruikte dataset wordt weggeschreven welke selectie er gemaakt is, tot welke SHRV ze behoren en welke plangrens en celgrootte gehanteerd zijn.
Plangrens	Dataset die de omvang van het plangebied aanduidt.
Samengestelde harde randvoorwaarde (SHRV)	Raster dataset met de omvang van het plangebied met daarin opgenomen het totaal aan gebieden dat wordt uitgesloten voor verdere analyse.

Relevante objectklassen component 8; “Invoegen SHRV”	
Tussenproduct	Een beoordeeld aspect, thema of integrale KW-raster.
Samengestelde harde randvoorwaarde (SHRV)	Raster dataset met de omvang van het plangebied met daarin opgenomen het totaal aan gebieden dat wordt uitgesloten voor verdere analyse.
Raster inclusief SHRV	Een beoordeeld aspect, thema of integrale KW-raster waar de SHRV aan toe is gevoegd.
Tabel “tussenproduct incl. SHRV”	Tabel waarin per tussenproduct wordt aangegeven welke samengestelde harde randvoorwaarde is toegepast. Een tussenproduct kan een aspect, thema of KW-raster zijn.

Objectklassen, attributen en mogelijke attribuutwaarden

Basisbestand	
<i>Id</i>	Unieke numerieke waarde voor ieder feature in het basisbestand
<i>Geometrie</i>	Beschrijft de geometrie van de features uit het basisbestand. De mogelijkheden zijn, punt, lijn of vlak.

Dataset in ruimtelijke database (vector)	
<i>Id</i>	Unieke numerieke waarde voor ieder ruimtelijk object in het basisbestand
<i>Geometrie</i>	Beschrijft de geometrie van de ruimtelijke objecten uit het basisbestand. De mogelijkheden zijn, punt, lijn of vlak.

Dataset in ruimtelijke database (raster)	
<i>Id</i>	Unieke numerieke waarde voor ieder ruimtelijk object in het basisbestand
<i>Geometrie</i>	Beschrijft de geometrie van de ruimtelijke objecten uit het basisbestand. De mogelijkheden zijn, punt, lijn of vlak.
<i>Value</i>	Iedere voorkomende waarde in het GRID komt een maal in dit veld voor.

Aspect^{*1} (Raster dataset in ruimtelijke database)	
<i>Object-ID</i>	Unieke numerieke waarde voor ieder voorkomende waarde in het value veld.
<i>Value</i>	Iedere voorkomende waarde in het GRID komt een maal in dit veld voor.
<i>Count</i>	Het aantal cellen in het GRID dat de bijbehorende waarde uit het value veld bevat.

^{*1} Indien geen euclidean distance is toegepast.

Beoordeeld aspect (Raster dataset in ruimtelijke database)	
<i>Object-ID</i>	Unieke numerieke waarde voor ieder voorkomende waarde in het value veld.
<i>Value</i>	Iedere voorkomende waarde in het GRID komt een maal in dit veld voor.
<i>Count</i>	Het aantal cellen in het raster dat de bijbehorende waarde uit het value veld bevat.

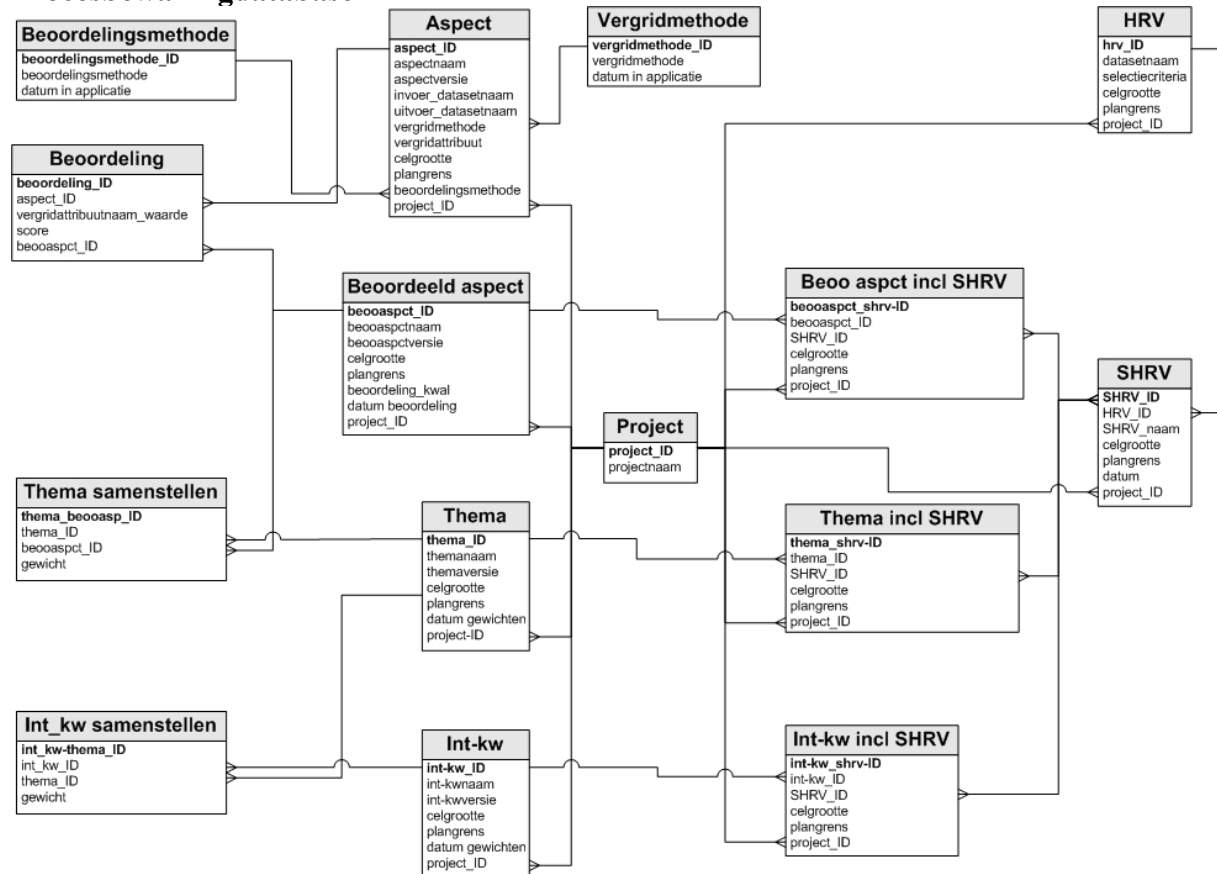
Thema (Raster dataset in ruimtelijke database)	
<i>Object-ID</i>	Unieke numerieke waarde voor ieder voorkomende waarde in het value veld.
<i>Value</i>	Iedere voorkomende waarde in het GRID komt een maal in dit veld voor.
<i>Count</i>	Het aantal cellen in het GRID dat de bijbehorende waarde uit het value veld bevat.

<i>Integrale KW-raster (Raster dataset in ruimtelijke database)</i>	
<i>Object-ID</i>	Unieke numerieke waarde voor ieder voorkomende waarde in het value veld.
<i>Value</i>	Iedere voorkomende waarde in het GRID komt een maal in dit veld voor.
<i>Count</i>	Het aantal cellen in het GRID dat de bijbehorende waarde uit het value veld bevat.

<i>SHRV (Raster dataset in ruimtelijke database)</i>	
<i>Object-ID</i>	Unieke numerieke waarde voor ieder voorkomende waarde in het value veld.
<i>Value</i>	Iedere voorkomende waarde in het GRID komt een maal in dit veld voor.
<i>Count</i>	Het aantal cellen in het GRID dat de bijbehorende waarde uit het value veld bevat.
<i>SHRV-omschrijf</i>	Omschrijving van de numerieke waarde van het value veld. Mogelijkheden zijn wel of geen (harde randvoorwaarde).

<i>Tussenproduct inclusief SHRV (Raster dataset in ruimtelijke database)</i>	
<i>Object-ID</i>	Unieke numerieke waarde voor ieder voorkomende waarde in het value veld.
<i>Value</i>	Iedere voorkomende waarde in het GRID komt een maal in dit veld voor. De mogelijke waarden liggen tussen 1 en 100. Cellen waar een SHRV voor geldt, krijgen Nodata.
<i>Count</i>	Het aantal cellen in het GRID dat de bijbehorende waarde uit het value veld bevat.
<i>(S)HRV-omschrijf</i>	Omschrijving van de numerieke waarde van het value veld. Mogelijkheden zijn wel of geen (harde randvoorwaarde).

Procesbewakingdatabase



Tabel “Project”	
<i>Project-ID</i>	Unieke numerieke waarde voor ieder project
<i>Projectnaam</i>	Tekstveld waarin de projectnaam komt te staan. Dit veld moet een unieke waarde bevatten.

Tabel “Vergridmethode”	
<i>Vergridmethode ID</i>	Unieke numerieke waarde voor iedere vergridmethode
<i>Vergridmethode</i>	Methode die gekozen kan worden om een aspect te creëren. Mogelijke keuzes zijn conversie van vector naar raster, een buffering in rasterformaat voor een vector of raster dataset en beoordelingsattribuut aanpassen voor een raster dataset.
<i>Datum in applicatie</i>	Datum wanneer deze methode (functionaliteit) is opgenomen in de applicatie

Tabel “Beoordelingsmethode”	
<i>Beoordelingsmethode ID</i>	Unieke numerieke waarde voor iedere beoordelingsmethode
<i>Beoordelingsmethode</i>	Methode die gekozen kan worden om een aspect te beoordelen. Mogelijke keuzes zijn absoluut, gradueel, in klassen, logaritmisch en exponentieel.
<i>Datum in applicatie</i>	Datum wanneer deze methode (functionaliteit) is opgenomen in de applicatie

Tabel “Aspect”	
<i>Aspect_ID</i>	Unieke numerieke waarde voor ieder voorkomend aspect
<i>Aspectnaam</i>	Naam van het aspect, vormt samen met aspectversie een uniek veld
<i>Aspectversie</i>	Versie van het aspect, vormt samen met aspectnaam een uniek veld
<i>Project_ID</i>	Verwijzing naar het project waar het thema bij hoort
<i>Invoer datasetnaam</i>	Naam van de invoer dataset uit de Geodatabase, inclusief complete padverwijzing.
<i>Uitvoer datasetnaam</i>	Naam van de uitvoer dataset uit de Geodatabase inclusief complete padverwijzing.
<i>Vergridmethode</i>	De methode die gebruikt is om het basisbestand om te zetten in een aspect. Mogelijke keuzes zijn conversie van vector naar raster, euclidean distance analyse voor vector en rasterstructuur en beoordelingsattribuut aanpassen voor een raster dataset. De mogelijkheden worden uitgelezen uit de tabel “Vergridmethode”
<i>Vergridattribuut</i>	Het attribuut uit de invoer dataset op basis waarvan deze dataset geconverteerd is naar een raster of welke gekozen is bij het aanpassen van het beoordelingsattribuut. Dit attribuut is niet van toepassing wanneer op het basisbestand een “straight line distance” bewerking is uitgevoerd. Als dit het geval is, wordt de standaardwaarde (niet van toepassing) ingevuld.
<i>Beoordelingsmethode</i>	De methode die gebruikt is bij het beoordelen van een aspect. De mogelijkheden zijn: absoluut, in klassen, gradueel, logaritmisch en exponentieel.
<i>Celgrootte</i>	Celgrootte van het raster dat het aspect representeert
<i>Plangrens</i>	Extent van het raster dat het aspect representeert

Tabel “Beoordeling”	
<i>Beoordeling_ID</i>	Unieke numerieke waarde, heeft verder geen functie
<i>Aspect_ID</i>	Verwijzing naar het aspect waarbij de beoordeling hoort
<i>Vergridattribuut_naam_Waarde</i>	De oorspronkelijke waarde of reeks van waarden van het vergridattribuut van het basisbestand.
<i>Score</i>	De score die aan de bijbehorende waarde(n) zijn toegekend.
<i>Beoospct_ID</i>	Verwijzing naar het beoordeelde aspect

Tabel “Beoordeeld aspect”	
<i>Beoospct_ID</i>	Unieke code voor ieder beoordeeld aspect, wordt naar verwezen vanuit de tabel “Beoordeling”
<i>Beoospctnaam</i>	De naam van het beoordeelde aspect, dit veld moet een unieke waarde bevatten.
<i>Beoospctversie</i>	Versie van het beoordeelde aspect, vormt samen met beoospctnaam een uniek veld
<i>Celgrootte</i>	Celgrootte van het raster dat het beoordeelde aspect representeert
<i>Plangrens</i>	Extent van het raster dat het beoordeelde aspect representeert
<i>Beoordeling_kwal</i>	Verwijzing naar rapport of document op basis waarvan beoordeling plaats heeft gevonden voor het beoordeelde aspect
<i>Datum beoordeling</i>	Datum waarop de beoordeling plaats heeft gevonden
<i>Project_ID</i>	Verwijzing naar het project waar het beoordeelde aspect bij hoort

Tabel “Thema samenstellen”	
<i>Thema_beoospct_ID</i>	Unieke numerieke waarde, heeft verder geen functie
<i>Thema_ID</i>	Numerieke waarde voor een thema. Dit veld kan vaker dezelfde waarde

	hebben omdat per beoordeeld aspect aangegeven moet worden tot welk thema ze behoort.
<i>Beoospct_ID</i>	Verwijzing naar een beoordeeld aspect dat deel uitmaakt van het thema.
<i>Gewicht</i>	Het gewicht dat aan het bijbehorende aspect is toegekend voor de gewogen somming.

Tabel “Thema”

<i>Thema_ID</i>	Unieke numerieke waarde, voor iedere combinatie van themanaam en themaversie is er een unieke waarde.
<i>Themanaam</i>	De naam van een thema, dit veld kan vaker dezelfde waarde hebben omdat per beoordeeld aspect aangegeven moet worden tot welk thema ze behoort.
<i>Themaversie</i>	Versie van het thema. Vormt samen met themanaam een uniek veld
<i>Celgrootte</i>	Celgrootte van het raster dat het thema representeert
<i>Plangrens</i>	Extent van het raster dat het thema representeert
<i>Datum gewichten</i>	Datum waarop de gewichtenset is toegekend en het thema is opgebouwd
<i>Project_ID</i>	Verwijzing naar het project waar het thema bij hoort

Tabel “Int_kw samenstellen”

<i>Int_kw-thema_ID</i>	Unieke numerieke waarde, heeft verder geen functie
<i>Int_kw-ID</i>	Numerieke waarde voor een integraal kwetsbaarheidraster. Dit veld kan vaker dezelfde waarde hebben omdat per beoordeeld aspect aangegeven moet worden tot welk thema ze behoort.
<i>Thema-ID</i>	Verwijzing naar een thema dat deel uitmaakt van het thema.
<i>Gewicht</i>	Het gewicht dat aan het bijbehorende thema is toegekend voor de gewogen somming.

Tabel “Int_kw”

<i>Int_kw-ID</i>	Unieke numerieke waarde, voor iedere combinatie van Int_kwnaam en Int_kwversie is er een unieke waarde.
<i>Int_kwnaam</i>	De naam van een integraal kwetsbaarheidraster. Vormt samen met Int_kwnaam een uniek veld.
<i>Int_kwversie</i>	Versie van het integrale kwetsbaarheidraster. Vormt samen met Int_kwnaam een uniek veld.
<i>Project-ID</i>	Verwijzing naar het project waar het integrale kwetsbaarheidraster bij hoort.
<i>Celgrootte</i>	Celgrootte van het raster dat het integrale kwetsbaarheidraster representeert
<i>Plangrens</i>	Extent van het raster dat het integrale kwetsbaarheidraster representeert
<i>Datum gewichten</i>	Datum waarop de gewichtenset is toegekend en het integrale kwetsbaarheidraster is opgebouwd

Tabel “SHRV”

<i>SHRV_ID</i>	Unieke numerieke waarde, heeft verder geen functie.
<i>SHRV-naam</i>	De naam van de samengestelde harde randvoorwaarde.
<i>Celgrootte</i>	Celgrootte van het raster dat de HRV representeert.
<i>Plangrens</i>	Extent van het raster dat de HRV representeert.
<i>Datum</i>	Datum waarop de SHRV is gedefinieerd.
<i>Project-ID</i>	Verwijzing naar het project waar de harde randvoorwaarde bij hoort.

Tabel “Beoo aspt incl. SHRV”	
<i>Beooasp_SHRV_ID</i>	Unieke numerieke waarde voor het raster dat ontstaat wanneer een SHRV wordt toegepast op een beoordeeld aspect.
<i>Beooaspct ID</i>	Verwijzing naar het beoordeelde aspect waarop de SHRV wordt toegepast.
<i>SHRV ID</i>	Verwijzing naar de toegepaste SHRV
<i>Celgrootte</i>	Celgrootte van het raster dat het beoordeelde aspect inclusief de SHRV representeert.
<i>Plangrens</i>	Extent van het raster dat het beoordeelde aspect inclusief de SHRV representeert.
<i>Project_ID</i>	Verwijzing naar het project waar het raster dat het beoordeelde aspect inclusief SHRV representeert bij hoort

Tabel “Thema incl. SHRV”	
<i>Thema_SHRV_ID</i>	Unieke numerieke waarde voor het raster dat ontstaat wanneer een SHRV wordt toegepast op een thema.
<i>Thema ID</i>	Verwijzing naar het thema waarop de SHRV wordt toegepast.
<i>SHRV ID</i>	Verwijzing naar de toegepaste SHRV
<i>Celgrootte</i>	Celgrootte van het raster dat het thema inclusief SHRV representeert
<i>Plangrens</i>	Extent van het raster dat het thema inclusief SHRV representeert
<i>Project_ID</i>	Verwijzing naar het project waar het raster dat het thema inclusief SHRV representeert bij hoort

Tabel “Int kw incl. SHRV”	
<i>Int_kw_SHRV-ID</i>	Unieke numerieke waarde voor het raster dat ontstaat wanneer een SHRV wordt toegepast op een beoordeeld aspect.
<i>Thema_ID</i>	Verwijzing naar het integrale kwetsbaarheid raster waarop de SHRV wordt toegepast.
<i>SHRV ID</i>	Verwijzing naar de toegepaste SHRV
<i>Celgrootte</i>	Celgrootte van het raster dat het thema inclusief SHRV representeert
<i>Plangrens</i>	Extent van het raster dat het thema inclusief SHRV representeert
<i>Project_ID</i>	Verwijzing naar het project waar het raster dat het thema inclusief SHRV representeert bij hoort

Relaties in de Keuzedatabase

Tussen de tabellen “vergridmethoden ” en “aspect” bevindt zich een een-op-veel relatie. De mogelijke vergridmethoden moeten uitgelezen worden uit de tabel “vergridmethoden”. Om in de tabel “aspect” voor te komen moet een vergridmethode zeker opgenomen zijn in de tabel “vergridmethoden”.

Tussen de tabellen “beoordelingsmethoden ” en “aspect” bevindt zich een een-op-veel relatie. De mogelijke beoordelingsmethoden moeten uitgelezen worden uit de tabel “vergridmethoden”. Om in de tabel “aspect” voor te komen moet een beoordelingsmethode zeker opgenomen zijn in de tabel “beoordelingsmethoden”.

Ieder aspect kan op meerdere manieren en met verschillende waarden beoordeeld worden. Daarom wordt vanuit de tabel “Beoordeling” via het attribuut “*aspect-ID*” verwezen naar de tabel “Aspect”. Een aspect-ID kan slechts een keer voorkomen in de tabel “Aspect” en meerdere malen in de tabel “Beoordeling”.

De tabel “Beoordeeld aspect” is gekoppeld aan de tabel “Beoordeling”. Dit is een een-op-veel relatie, want een beoordeeld aspect kan meerdere waarden hebben. Voor iedere waarde van het beoordeelde aspect is er een record in de tabel “Beoordeling”. Via het attribuut “*beooaspectnaam*” kan direct de beoordeling achterhaald worden.

Tussen de tabellen “Beoordeeld aspect” en “Thema samenstellen” bestaat een een-op-veel relatie via het attribuut “*beooaspect-ID*”. Een beoordeeld aspect kan in meerdere thema’s voorkomen en daardoor ook meerdere malen voorkomen in de tabel “Thema”.

De tabellen “Thema samenstellen” en “Thema” hebben een relatie via het attribuut “*Thema_ID*”. De opbouw van ieder thema dat in de tabel “Thema” voorkomt, is in de tabel “Thema samenstellen” te vinden.

Tussen de tabellen “Thema” en “Int_kw samenstellen” bestaat een relatie via het attribuut “*thema_ID*”. Een thema kan in meerdere integrale kwetsbaarheidrasters opgenomen worden en daardoor ook meerdere malen voorkomen in de tabel “Int_kw samenstellen”.

De tabellen “Int_kw samenstellen” en “Int_kw” hebben een relatie via het attribuut “*Int_kw_ID*”. De opbouw van iedere integrale kwetsbaarheidraster die in de tabel “Int_kw” voorkomt, is in de tabel “Int_kw samenstellen” te vinden.

In de tabel “Beoo aspect incl. SHRV” wordt vastgelegd welke combinaties gemaakt zijn tussen beoordeelde aspecten en SHRV’s. Op een beoordeeld aspect kunnen meerdere SHRV’s worden toegepast en een SHRV kan op meerdere beoordeelde aspecten worden toegepast.

In de tabel “Thema incl. SHRV” wordt vastgelegd welke combinaties gemaakt zijn tussen thema’s en SHRV’s. Op een thema kunnen meerdere SHRV’s worden toegepast en een SHRV kan op meerdere thema’s worden toegepast.

In de tabel “Int_kw incl. SHRV” wordt vastgelegd welke combinaties gemaakt zijn tussen Integrale kwetsbaarheidrasters en SHRV’s. Op een Integraal kwetsbaarheidraster kunnen meerdere SHRV’s worden toegepast en een SHRV kan op meerdere Integrale kwetsbaarheidrasters worden toegepast.

Validatieregels

De database is zo ontworpen zodat meerdere GISMER projecten in een database opgenomen kunnen worden. Het is dan wel noodzakelijk te weten welke bestanden bij een bepaald project horen. Anders is vrijwel niet meer te achterhalen welke plangrens en celgrootte gebruikt zijn bij de verschillende beoordeelde aspecten en HRV's en levert dit problemen op bij het samenvoegen en combineren. Daarvoor is in de tabellen waar dit nodig is, een attribuut project_ID opgenomen. De regels die hier betrekking op hebben zijn hieronder weergegeven in de vorm: tabel.attribuut = tabel.attribuut

Aspect.Project_ID = Beoordeeld aspect.Project_ID
Beoordeeld aspect.Project_ID = Beoordeeld aspect.Project_ID (Bij samenvoegen beoo aspect)
Beoordeeld aspect.Project_ID = Thema.Project_ID
Thema.Project_ID = Thema.Project_ID (Bij samenvoegen thema's)
HRV.Project_ID = HRV.Project_ID (Bij samenvoegen HRV's)
HRV.Project_ID = SHRV.Project_ID
Beoordeeld aspect.Project_ID = SHRV.Project_ID (Bij combineren beoordeeld aspect en SHRV)
Thema.Project_ID = SHRV.Project_ID (Bij combineren thema en SHRV)
Int_kw.Project_ID = SHRV.Project_ID (Bij combineren integraal kwetsbaarheidraster en SHRV)

Som van de gewichten per Thema.Themanaam = 100

Som van de gewichten per Int_kw.Int_kwnaam = 100

Bijdrage VI: Databeheer en ontwikkelmethoden in ArcInfo en ArcGIS

In dit hoofdstuk worden databeheer en de programmeerwijzen in ArcInfo en ArcGIS behandeld. Eerst worden het databeheer en de programmeerwijze van ArcInfo besproken, deze hebben betrekking op de huidige GISMER applicatie. Daarna worden de Geodatabase en het programmeren in ArcObjects besproken welke betrekking hebben op de implementatie van GISMER in ArcGIS.

Databeheer in ArcInfo workstation

ArcInfo ondersteunt verschillende typen datastructuren zoals vector, raster, tin (triangulated irregular network) en network. Van deze typen datasets hebben vector en raster de meeste toepassingsmogelijkheden (ESRI, 1994). Het zijn de enige twee die gebruikt worden in de huidige versie van GISMER. In dit hoofdstuk wordt daarom ook vooral aandacht besteed aan deze twee typen datasets. Binnen ArcInfo worden de geometrische gegevens en de thematische gegevens afzonderlijk van elkaar opgeslagen. De gegevens worden opgeslagen plaats in een bestand-directory structuur waarbij de directories samen een workspace vormen (ESRI, 1990, 1994).

Vector datastructuren

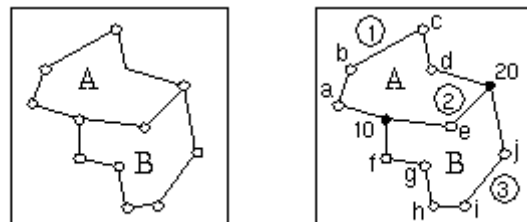
Een vector datastructuur is een geometrische structuur om ruimtelijke data in digitale vorm op te slaan, te representeren en weer te geven. Het gebruikt coördinaten om locaties op het aardoppervlak aan te duiden. Ruimtelijke objecten bestaan uit een punt, een keten van punten (lijn) of een gesloten keten van punten (polygoon).

Coverages

Binnen ArcInfo wordt vector data opgeslagen in coverages. Coverages kunnen uit meerdere feature classes zoals arc, node, polygons, label points en tics bestaan die vaak geografische elementen weergeven. Verschillende ruimtelijke verschijnselen worden in aparte coverages opgeslagen. Informatie over grondgebruik wordt bijvoorbeeld in een ander coverage opgeslagen dan informatie over grondwaterstanden.

Geometrische gegevens

De geometrische opslag van de features is gebaseerd op de arc-node datastructuur, deze zorgt er voor dat features van verschillende klassen effectief opgeslagen kunnen worden binnen een coverage (ESRI, 1994).



Arc Number	Start Node	Vertices	End Node
1	20	d, c, b, a	10
2	10	e	20
3	10	f, g, h, i, j	20

Polygon	Arc List
A	1, 2
B	2, 3

De arc-node datastructuur (ArcInfo help)

Een arc is een lijnstuk tussen twee nodes (eindpunten) en heeft daartussen meerdere vertices. De lijnstukken omsluiten op hun beurt weer de polygonen. Deze arc-node datastructuur is de basis voor topologie.

Topologie beschrijft ruimtelijke relaties tussen features en heeft drie hoofdconcepten. Deze zijn in onderstaand kader opgesomd. Topologie zorgt er voor dat gebieden en relaties tussen features gedefinieerd kunnen worden, ongeacht uit welke feature class ze komen.

Verbondenheid: lijnstukken zijn verbonden aan elkaar via de eindpunten
Oppervlakte definitie: lijnstukken die met elkaar verbonden zijn en zo een oppervlakte insluiten, definiëren een polygoon
Naburigheid: Lijnstukken hebben richting en een linker- en rechterkant

Drie hoofdconcepten van topologie (ESRI, 1994)

Thematische gegevens

Thematische gegevens van coverages worden opgeslagen in attribuut tabellen. Deze tabellen zijn zogenaamde INFO tabellen en hebben een aantal standaard attributen. Daarnaast kan de gebruiker zelf attributen toevoegen. De namen van de tabellen zijn gelijk aan de naam van de coverage. De drie lettercode achter de punt in de bestandsnaam geeft aan tot welke feature class de tabel behoort. Ieder feature krijgt een unieke code waarmee de geometrische en thematische data aan elkaar gekoppeld kunnen worden. De features in de coverage hebben een een-op-een relatie tussen de tabel met geometrische gegevens en de thematische tabel (ESRI, 1994).

Shapes

Een tweede vector datamodel is de shape, deze wordt vooral binnen Arcview gebruikt en wordt ook wel de lichtgewicht versie van de coverage genoemd. Binnen een shape wordt geen topologie aangemaakt en polygonen mogen elkaar overlappen. Dit zorgt er voor dat de ze makkelijker en simpeler is dan de coverage, maar heeft daardoor ook beperkingen met betrekking tot ruimtelijke analyses (NSIDC, 2003). Om shapes voor bewerkingen te kunnen gebruiken in ArcInfo moeten ze eerst geconverteerd worden naar coverages.

Raster datastructuren

Een raster dataset worden opgeslagen in een regelmatig grid met cellen die gevuld zijn met waarden. Iedere locatie wordt gerepresenteerd door een cel. Een matrix van cellen (rijen en kolommen) wordt een GRID genoemd. Een punt wordt weergegeven door een enkele cel, een lijn door een serie verbonden cellen die een gezamenlijk een lengte hebben en een vlak door een groep verbonden cellen die een vorm weergeven. De nauwkeurigheid van een raster is afhankelijk van de celgrootte.

GRID

Raster data wordt in ArcInfo opgeslagen in een grid. Binnen een GRID worden, net als bij coverages, gegevens die tot verschillende ruimtelijke verschijnselen behoren, apart opgeslagen. Ieder GRID representeert een ruimtelijke variabele en bestaat uit cellen die zijn opgeslagen als rijen en kolommen. De cellen hebben allemaal een numerieke waarde die de ruimtelijke variabele voor die locatie vertegenwoordigt.

In de value attribute table (VAT-tabel) worden de attribuutgegevens van het GRID opgeslagen. Elke waarde die voorkomt in het GRID is een record in de tabel. Naast het "VALUE" veld in de tabel heeft de "VAT"-tabel altijd een "COUNT" veld waarin het aantal cellen met de betreffende waarde staat. Verder kunnen er velden in de tabel op worden genomen die de waarden uit het GRID beschrijven. In tegenstelling tot de attribuut tabellen bij coverages bevat de "VAT" tabel dus gegevens over een bepaalde gridwaarde, en niet over specifieke geografische elementen (ESRI, 1994). Het koppelen van de geometrische en thematische gegevens gaat via het waarde veld, er bestaat dus een een-op-veel relatie tussen de geometrische tabel en de thematische tabel.

Een Digital Elevation Model (DEM) is een raster representatie van een continu oppervlak. Dit houdt in dat aan iedere locatie een hoogte is gekoppeld. De nauwkeurigheid van de DEM is afhankelijk van de

resolutie en van het datatype (integer of floating point). Uitschieters naar beneden in een DEM worden sinks genoemd en uitschieters naar boven peaks.

Overige datastructuren

Naast vector en raster datastructuren zijn er ook nog een aantal andere datastructuren. TIN en Network zijn er daar twee van.

TIN

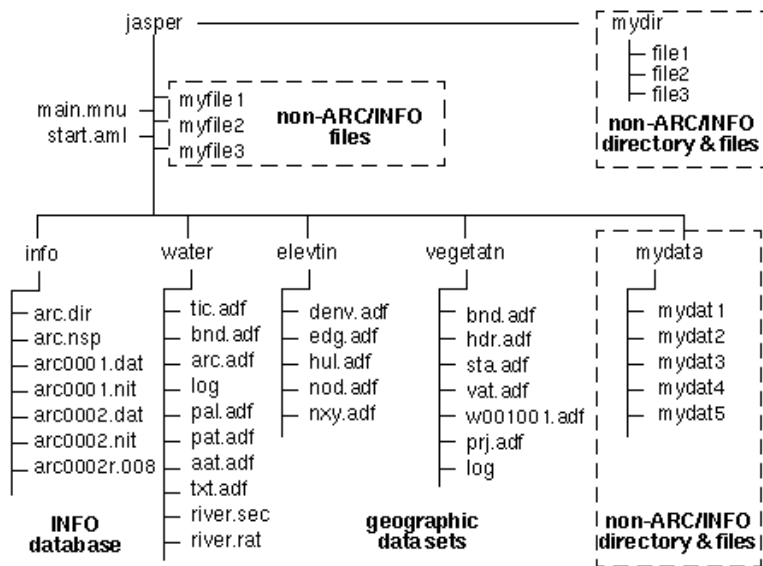
Een TIN kan gebruikt worden om een oppervlak mee weer te geven, in het bijzonder terreinen. Driehoeken, lijnstukken en punten zijn de drie hoofdelementen van een TIN. Als invoer kunnen punten, lijnen en vlakken gebruikt worden, bij lijnen en vlakken worden de waarden van de vertexen, dus in feite ook punten gebruikt. De punten en vertexen die gebruikt worden hebben allemaal een x, y en z waarde. Tussen de punten worden lijnen getrokken die vervolgens weer driehoeken vormen.

Network

Een network bestaat uit een aantal aan elkaar verbonden line features en is een abstracte weergave van real-world netwerken zoals een waterlopen- of wegenpatroon. Een network bestaat uit links, nodes, stops, centers en turns. Links zijn verbonden line features en zijn in feite gewoon arcs, ze zijn opgeslagen in een arc attribute tabel. Nodes zijn de eindpunten van arcs en zijn opgeslagen in een node attribute table. Stops zijn plaatsen in het network die bezocht worden en centers zijn plaatsen met een bepaalde aantrekkingskracht. Deze zijn opgeslagen in een INFO bestand. Turns zijn de afslagen die mogelijk zijn in het network en worden ook in een INFO bestand opgeslagen.

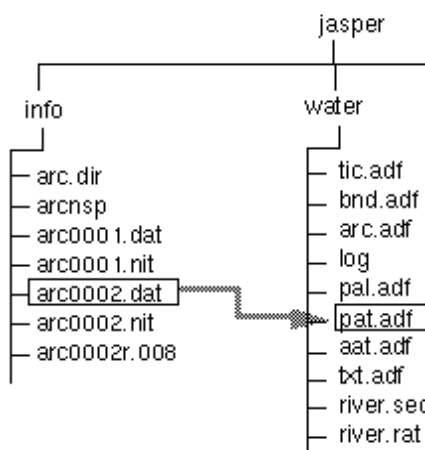
Workspace

Binnen ArcInfo wordt er gewerkt in een workspace. Deze is opgebouwd volgens een bestand-directory structuur. In onderstaande figuur is de opbouw van een workspace weergegeven.



Workspace structuur (ArcInfo help)

De workspace zelf is een directory waaronder een aantal bestanden en een aantal subdirectory's staan. De geografische datasets zijn binnen een workspace opgeslagen als een subdirectory. De subdirectory bevat alle geometrische en thematische gegevens van die dataset. Iedere feature class binnen de dataset heeft haar eigen set bestanden. In de figuur hiernaast heeft de dataset water bijvoorbeeld een feature class TIC, BND, ARC enz.



Koppeling tussen INFO database en de dataset water (ArcInfo help)

Naast deze datasets is er ook de INFO directory. In de figuur hiernaast is te zien dat deze bestanden bevat met verwijzingen naar de tabellen met de attribuuatgegevens in de datasets. Welke bestanden uit de INFO directory verwijzen naar welke dataset en welk bestand binnen die dataset is alleen te zien binnen de INFO module van ArcInfo.

De ArcInfo bestanden die direct onder de workspace staan zijn ondersteunende bestanden, zoals een projectiebestand of een AML bestand. Niet ArcInfo bestanden zoals een DBF bestand, zijn bestanden die wel gekoppeld kunnen worden aan een ArcInfo dataset, maar maken hier zelf geen deel van uit. Deze bestanden kunnen in een aparte directory buiten de workspace bewaard worden, of in een aparte directory die wel onder de workspace valt.

Ontwikkelen in ArcInfo

ArcInfo kan bestuurd worden met behulp van Arc Macro Language (AML). Hiermee kunnen allerlei acties uitgevoerd en geautomatiseerd worden. Ook kunnen eigen commando's en user interfaces gemaakt worden. Hierdoor kan ArcInfo meer gebruiksvriendelijk gemaakt worden. AML is gebaseerd op Prime's Computer's Command Procedural Language en werkt met losse commando's (Fitzgerald, 2002).

De commando's kunnen na elkaar ingetypt en uitgevoerd worden in het Arc scherm van ArcInfo. Ook kunnen ze achter elkaar gezet in een tekstbestand als Macro opgeslagen worden. De Macro's worden van boven naar beneden gelezen. De eerstgenoemde commando's worden dus het eerst uitgevoerd en de laatst genoemde het laatst. Naast dit type AML bestand, kunnen er ook nog menu's aangemaakt worden met AML. Deze bestaan eigenlijk uit een lijst met keuzes op het scherm en sturen vaak Macro's of andere menu's aan (ESRI, 1990).

Bij het uitvoeren van een Macro wordt een AML bestand vertaald door de AML processor die eerst voorbereidingen doet, zoals variabele substitutie. Vervolgens stuurt de AML processor het Arc programma aan om een commando uit te voeren. Het commando dat uiteindelijk wordt uitgevoerd door het Arc programma is dus afhankelijk van de interpretatie door de AML processor (ESRI, 1990).

Er bestaan drie typen AML elementen; directives, variabelen en functies (ESRI, 1990 en University of Durham, 2003). Directives voeren acties uit die niet direct door worden gegeven aan een programma. Ze sturen bijvoorbeeld een AML aan of plaatsen een bericht op het scherm. Met behulp van directives kunnen losse Macro's aan elkaar gekoppeld worden tot een complete applicatie die onder ArcInfo workstation draait. Variabelen worden gebruikt om dynamische informatie op te slaan of om tekst of bestandsnamen te verkorten. Ze kunnen later in de Macro aangeroepen worden. Voordat een commando wordt uitgevoerd vindt er eerst variabele substitutie plaats. Het derde type element binnen AML is een functie, deze wordt binnen een commando gebruikt. Als de functie is uitgevoerd, wordt de uitkomst hiervan verder gebruikt voor het commando waarbinnen de functie is opgenomen.

Data in ArcGIS

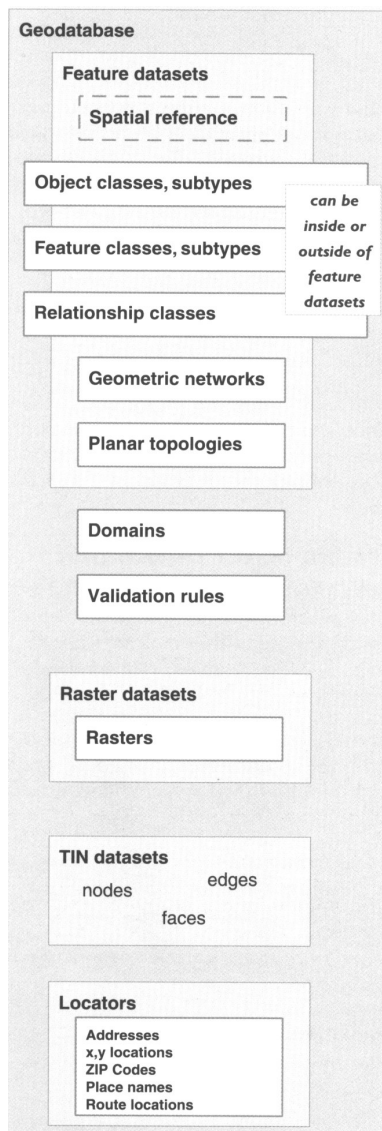
In ArcGIS bestaat de mogelijkheid datasets op te slaan in tabellen. De tabellen worden binnen ArcGIS beheerd in een nieuw datamodel, de Geodatabase. Deze heeft de structuur van een database management systeem (DBMS) (ArcGIS Desktop Help). In de Geodatabase kunnen eigenschappen, gedrag en relaties aan features meegegeven worden. De Geodatabase heeft drie belangrijke kenmerken. Deze zijn in de figuur hier onder weergegeven.

- Data kan tegelijkertijd, door meerdere personen bekeken en bewerkt worden en er kunnen versies van Geodatabases gemaakt worden.
- Een centraal beheer van geografische data in een DBMS.
- Aan features kan een vorm van gedrag en intelligentie worden toegekend.

De drie belangrijkste kenmerken van de Geodatabase (ESRI, 2003)

De Geodatabase is niet het enige datamodel om data op te slaan voor gebruik in ArcGIS. Datasets kunnen in ArcGIS ook gebruikt worden als ze op een schijf worden opgeslagen in een workspace zonder dat ze in een Geodatabase staan. Daarnaast kunnen ook bestaande datamodellen zoals shapes en coverages gebruikt worden in ArcGIS.

Personal en enterprise Geodatabases



Geodatabases bestaan in twee typen, personal en enterprise. Tussen deze twee typen Geodatabases bestaan twee grote verschillen. Ten eerste het data formaat, een personal Geodatabase is opgeslagen in een Microsoft Access bestand en heeft daardoor de beperking van 2 gigabyte. Een enterprise Geodatabase is opgeslagen in een DBMS dat beheerd wordt door ArcSDE (ESRI, 2003) en heeft geen beperkingen wat omvang betreft. Het tweede verschil zit in de mogelijkheid die het datamodel van de Geodatabase biedt om de enterprise database door meerdere mensen tegelijk te wijzigen en te raadplegen (multi-user-editing). Dit kan alleen met een enterprise Geodatabase. Hierdoor is het mogelijk om verschillende versies van dezelfde database te hebben voor verschillende fasen van een project. De verschillende versies kunnen met elkaar vergeleken worden en de oorspronkelijke database kan bijgewerkt worden (ESRI, 2003).

Databeheer in de Geodatabase

In de Geodatabase kunnen drie verschillende datastructuren opgenomen worden, vector, raster en TIN. Naast deze drie typen datastructuren kunnen er ook nog andere elementen in de Geodatabase voorkomen.

Feature dataset

Vector data wordt in de Geodatabase opgeslagen in een feature dataset. Een feature dataset is een verzameling feature classes met een gelijke ruimtelijke referentie. In een feature dataset moet deze ruimtelijke referentie dan ook altijd gegarandeerd zijn (Zeiler, 1999). Naast feature classes kan de feature dataset nog een aantal andere elementen bevatten. Hieronder worden alle mogelijke elementen kort toegelicht.

De Geodatabase en haar elementen (Zeiler 1999)

Feature Class

Geometrie wordt binnen een feature dataset opgeslagen in de feature class, dit is een verzameling van features met een gelijke geometrie, dezelfde attributen, hetzelfde gedrag en met dezelfde ruimtelijke referentie. Binnen feature classes is er de tweedeling te maken tussen simpele en topologische.

Simpele feature classes bevatten punten, lijnen, of polygonen zonder topologische relaties. Deze mogen ook buiten de dataset, los in de Geodatabase opgeslagen worden. Punten uit een feature class kunnen bijvoorbeeld samenvallen met eindpunten van een lijn uit een andere feature class. Er is dan geen topologische relatie, dus kunnen de features afzonderlijk van elkaar gewijzigd worden.

Bij topologische feature classes is dit niet het geval. Hier is er wel een relatie tussen de verschillende classes en kunnen features niet afzonderlijk van elkaar gewijzigd worden. Topologische feature classes moeten dan ook altijd in de feature dataset staan.

De geometrie van een feature is opgeslagen in het "SHAPE" veld van de feature class tabel. In dezelfde tabel kunnen meerdere velden worden opgenomen om de features te beschrijven.

Object class

Een object class is een tabel met niet-ruimtelijke gegevens, maar die wel met ruimtelijke gegevens gerelateerd kunnen zijn. Object classes hebben zelf geen geometrie en zijn dus geen features. Een voorbeeld van een object class is een tabel met gegevens over welke gewassen wanneer op welk kavel hebben gestaan.

Relationship class

Een relationship class is een tabel die de relaties tussen features of elementen uit twee feature classes of object classes bevat. Relaties bepalen de afhankelijkheid en de associaties tussen elementen (Zeiler, 1999). Deze elementen kunnen zowel ruimtelijk als niet ruimtelijk zijn. Relationship classes bestaan in twee types, namelijk simple en composite.

Simple relationship class

Bij simple relationship classes bestaan records in de twee gerelateerde tabellen onafhankelijk van elkaar. Indien record A gerelateerd is aan record B en A wordt verwijderd, dan blijft B bestaan. Bij composite relationship classes is dit niet het geval, daar zijn de gerelateerde records afhankelijk van elkaar.

Composite relationship

Met dit type relationship class kan een record in een tabel de levensduur van records in andere tabellen controleren. Relationship rules geven aan welke subtypes uit de verschillende tabellen met elkaar gerelateerd mogen zijn en geven de cardinaliteit van een relatie aan. De cardinaliteit geeft aan hoeveel records uit een tabel gerelateerd mogen zijn met een record uit de andere tabel.

Validation rules, subtypes en domains

Om te zorgen dat er alleen elementen in de Geodatabase voorkomen die voldoen aan door de gebruiker opgestelde regels, worden validation rules opgenomen. Ze houden de data integriteit tussen elementen in de Geodatabase in de gaten. Er bestaan drie verschillende typen validation rules, namelijk relationship rules, attribute rules en network connectivity rules.

Relationship rules geven aan welke subtypes met elkaar gerelateerd mogen zijn. Subtypes zijn groepen features uit een feature class die dezelfde attributen hebben. Een feature class met wegtypen kan bijvoorbeeld onderverdeeld worden in drie subtypen, lokale wegen, provinciale wegen en nationale wegen. Daarnaast is het mogelijk om met validation rules de cardinaliteit te bepalen. Dit wil zeggen: het aantal features of objecten uit een tabel bepalen dat gerelateerd mag zijn aan het aantal features of objecten uit een andere tabel.

Attribute rules zijn regels die bepalen welke waarden een attribuut mag bevatten in een tabel. Dit gebeurt aan de hand van attribute domains. Domains bevatten de toegestane waarden voor een attribuut. Wanneer subtypes zijn aangemaakt, kan aan ieder subtype een ander attribute domain worden toegewezen. Ook is het mogelijk om per subtype standaardwaarden (default values) toe te kennen aan attributen.

Network connectivity rules zijn regels die het type en het aantal netwerk features beperken dat in een Geodatabase aan elkaar verbonden kan zijn.

Validation rules, subtypes en attribute domains worden door ArcGIS in tabellen in de Geodatabase weggeschreven. De tabellen waarin dit gebeurt worden bij het creëren van een nieuwe Geodatabase automatisch aangemaakt. Deze tabellen zijn niet als tabel in de standaard gebruikersinterface van ArcGIS benaderbaar.

Raster dataset

De raster dataset in de Geodatabase is een twee dimensionale matrix met cellen. Iedere cel heeft een eigen numerieke waarde. Het is vanaf ArcGIS 9 mogelijk om een raster dataset in een personal Geodatabase op te nemen. In eerdere versies van ArcGIS is het al mogelijk een raster in een enterprise Geodatabase op te nemen.

TIN dataset

TIN-datasets zijn opgeslagen als een serie punten met een x, y en z coördinaat. Deze punten staan niet in een regelmatig grid, de dichtheid van de punten hangt af van de hellingsverandering van het oppervlak. Hoe meer veranderingen in helling er zijn en hoe groter deze zijn, hoe dichter de punten bij elkaar liggen.

Gedrag en intelligentie van features

In de Geodatabase kunnen regels, relaties en topologische verbindingen opgenomen worden waarmee eigenschappen meegegeven kunnen worden aan features. Features waar gedrag aan mee is gegeven, worden custom features genoemd. Er kunnen bijvoorbeeld relaties gelegd worden tussen object classes en feature classes, of tussen feature classes onderling. Features in de Geodatabase kunnen hierdoor veel meer hun gedaante en gedrag aannemen uit de praktijk.

Een voorbeeld is te vinden in de figuur hiernaast, die een ongelijkvloerse kruising tussen twee wegen voorstelt. In de Geodatabase is het niet meer nodig dat er een extra punt wordt opgeslagen op de plek waar de wegen kruisen, er worden nu twee wegen opgeslagen in plaats van vier lijnstukken die twee wegen voor moeten stellen.



Intelligentie van features (Zeiler,

Geometric networks

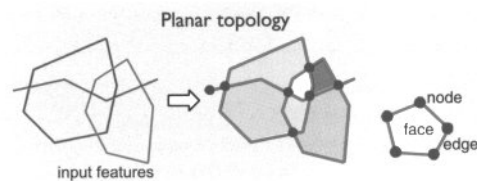
Geometrische netwerken bestaan uit edges die aan hun uiteinden nodes hebben. Deze nodes worden ook wel junctions genoemd en kunnen met een of meerdere edges verbonden zijn. De verbindingen worden tot stand gebracht door topologische relaties die automatisch gehandhaafd worden.



Geometric network (Zeiler, 1999)

Planar topology

Planar topology kan gecreëerd worden uit een aantal invoer features, dit kunnen zowel lijnen als polygonen zijn. In feite vindt er een soort union overlay plaats waarbij er nodes (eindpunten), edges (lijnstukken) en faces (vlakken) worden aangemaakt. Wanneer nu een punt van een vlak gewijzigd wordt, wordt ook meteen het andere vlak waar het punt deel van uit maakt gewijzigd.



Planar topology (Zeiler, 1999)

Locator

Met een locator kunnen locaties gevonden worden die vervolgens opgeslagen worden als een nieuwe feature class. De locator is de combinatie van de methode waarmee locaties gevonden kunnen worden en de referentiedata. De invoer voor een locator is een tabel met locatie data, deze beschrijft de locaties die gevonden moeten worden in de dataset.

Databeheer in ArcGIS

Data wordt binnen ArcGIS beheerd in een aparte applicatie, ArcCatalog. Met deze applicatie kan door datasets gebladerd worden en kunnen datasets en Geodatabases gemaakt en ingericht worden. Het bekijken van datasets kan op drie niveaus. Ten eerste kan de inhoud van een dataset bekeken worden, vervolgens kan een preview van de geografie en van de tabellen gemaakt worden en tenslotte kan metadata beschreven en bekeken worden. Bestaande datasets kunnen makkelijk bewerkt worden door ze te kopiëren van naam te wijzigen of door items toe te voegen. Nieuwe datasets of een Geodatabase worden in ArcCatalog gecreëerd en ingericht voordat ze gebruikt kunnen worden.

Ontwikkelen in ArcGIS 9

Binnen ArcGIS9 zijn een aantal middelen beschikbaar om tools en functies te ontwikkelen. Dit zijn de model builder, scripting en programmeren met behulp van ArcObjects. Hieronder worden deze methoden kort beschreven.

Model builder.

Met de model builder is het mogelijk om schematisch een model op te bouwen door verschillende GIS functies met elkaar te verbinden in een soort stroomschema. De gebruiker kan de inputs definiëren als parameter. Deze kunnen dus gewijzigd kunnen worden. Een model dat is opgeslagen kan bewerkt worden als stroomschema, maar kan ook geopend worden in de vorm van een dialoogscherm. In dit dialoogscherm worden alle parameters gevraagd die door de gebruiker zijn gedefinieerd. Modellen zijn ook te exporteren naar script (Python, Vbscript, Jscript). Een model kan ook weer als tool in een model gebruikt worden.

Scripting

Scripting kan gebruikt worden voor het automatiseren van processen. De ontwikkelaar kan zelf scripts schrijven of kan scripts die gegenereerd zijn door de model builder bewerken. Welke taal hiervoor gebruikt wordt maakt niet veel uit, als die maar COM-ondersteunend is. Door ESRI wordt python geadviseerd. Scripts zijn te gebruiken in de model builder.

ArcObjects

ArcObjects is het ontwikkelplatform waarmee ArcGIS applicaties zoals ArcMap en ArcCatalog uitgebreid kunnen worden. Programmeren met ArcObjects is object georiënteerd en is gebaseerd op

het Component Object Model (COM) van Microsoft. Daarom is het mogelijk om delen van de bestaande functionaliteit van ArcGIS, of een andere object georiënteerd geprogrammeerde applicatie, op te nemen in een in ArcObjects geprogrammeerde tool of applicatie. Handelingen kunnen zo vereenvoudigd of uitgebreid worden (ESRI, 2001, 2).

Ontwikkelen met behulp van ArcObjects kan op twee manieren. Ten eerste door zelf te programmeren aan de hand van het ArcGIS Object Model. Ten tweede met behulp van ArcGIS Engine.

Voor het programmeren aan de hand van het ArcGIS Object Model kan iedere COM ondersteunende programmeertaal gebruikt worden. Visual Basic is echter de meest gebruikte programmeertaal om componenten te ontwikkelen die samenwerken met ArcMap of ArcCatalog (<http://arcobjectsonline.esri.com/>, 2003).

ArcGIS Engine is een ontwikkelproduct waarmee GIS applicaties gecreëerd en gelanceerd kunnen worden. ArcGIS desktop functionaliteit kan op worden genomen in een applicatie. Deze applicatie kan vervolgens draaien op een workstation zonder ArcGIS, maar met een zogenaamde deployment licentie.

Uitwisselbaar

De resultaten van de verschillende methoden zijn uitwisselbaar. Modellen en scripts kunnen aangeroepen worden vanuit een andere programmeeromgeving. Dit wil echter niet zeggen dat scripts (bijvoorbeeld in vbScript formaat) rechtstreeks gekopieerd kunnen worden in Visual Basic wanneer geprogrammeerd wordt met behulp van ArcObjects. Wanneer een script gebruikt moet worden bij een applicatie die geprogrammeerd is binnen ArcObjects moet eerst de geoprocessing environment aangeroepen worden. Van daaruit kan vervolgens het script aangestuurd worden. De ontwikkelaar kan de techniek kiezen die het meest geschikt is voor het ontwikkelen van die specifieke tool of applicatie (Eijkelenboom, 2003).

Bijlage VII: Scripts en dialoogschermen gegenereerd door ArcGIS Desktop 9.0 Pre Release

Scripts

Hier onder worden de scripts en de dialoogschermen weergegeven van de tweede en derde component. De scripts en dialoogschermen zijn gegenereerd door de ArcGIS model builder. De scripts zijn in python formaat.

C2 importeren vector basisdata

```
# -----
# C2 import vector.py
# Created on: Thu Apr 15 2004 01:09:41 PM
# (generated by ArcGIS/ModelBuilder)
# Usage: C2 import vector <Invoer_vector_basisdata> <Uitvoer_vector_dataset_naam>
<Uitvoer_workspace>
# Description:
# Euclidean distance + importeren van data in Gdb
# -----

# Import system modules
import sys, string, os, win32com.client

# Create the Geoprocessor object
gp = win32com.client.Dispatch("esriGeoprocessing.GpDispatch.1")

# Load required toolboxes...
gp.AddToolbox("C:/ArcGIS/ArcToolbox/Toolboxes/Data Management Tools.tbx")

# Set the Geoprocessing environment...
gp.SAEnvExtent = "DEFAULT"
gp.outputZFlag = "Same As Input"
gp.outputZValue = ""
gp.outputMFlag = "Same As Input"
gp.outputSpatialRef = ""
gp.cluster_tolerance = ""
gp.defaultWorkspace = "M:/Implementatie/ArcMER9_11022004.mdb/Basisbestanden_feat"
gp.workspace = ""

# Script arguments...
Invoer_vector_basisdata = sys.argv[1]

Uitvoer_vector_dataset_naam = sys.argv[2]

Uitvoer_workspace = sys.argv[3]
if Uitvoer_workspace == '#':
    Uitvoer_workspace = "M:/Implementatie/ArcMER9_11022004.mdb" # provide a default value if
    unspecified

# Local variables...
Uitvoer_ruimtelijke_objecten = "M:/Implementatie/ArcMER9_11022004.mdb/landuse11"
Uitvoer_ruimtelijke_referentie = "{B286C06B-0879-11D2-AACA-00C04FA33C20}"
```

Tijdelijke_ruimtelijke_objecten = "xtmplyr"

Uitvoer_vector_dataset = "M:/Implementatie/ArcMER9_11022004.mdb/landuse11"

Process: Maken tijdelijke ruimtelijke objecten...

```
gp.MakeFeatureLayer_management(Invoer_vector_basisdata, Tijdelijke_ruimtelijke_objecten, "",  
Uitvoer_workspace, "AREA AREA VISIBLE;PERIMETER PERIMETER VISIBLE;LANDUSE_ID  
LANDUSE_ID VISIBLE;C2 C2 VISIBLE")
```

Process: Ruimtelijke referentie bepalen...

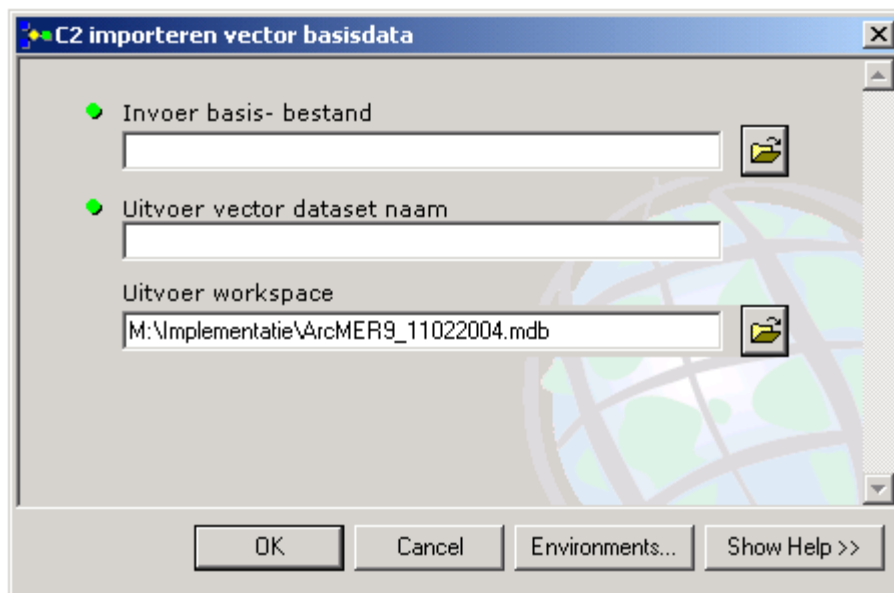
```
gp.CreateSpatialReference_management("{B286C06B-0879-11D2-AACA-00C04FA33C20}",  
Invoer_vector_basisdata, "", "", "", "", "", "0", )
```

Process: Vector dataset aanmaken...

```
gp.CreateFeatureclass_management(Uitvoer_workspace, Uitvoer_vector_dataset_naam,  
"POLYGON", "", "Same As Template", "Same As Template", Uitvoer_ruimtelijke_referentie, "",  
"0", "0", "0", )
```

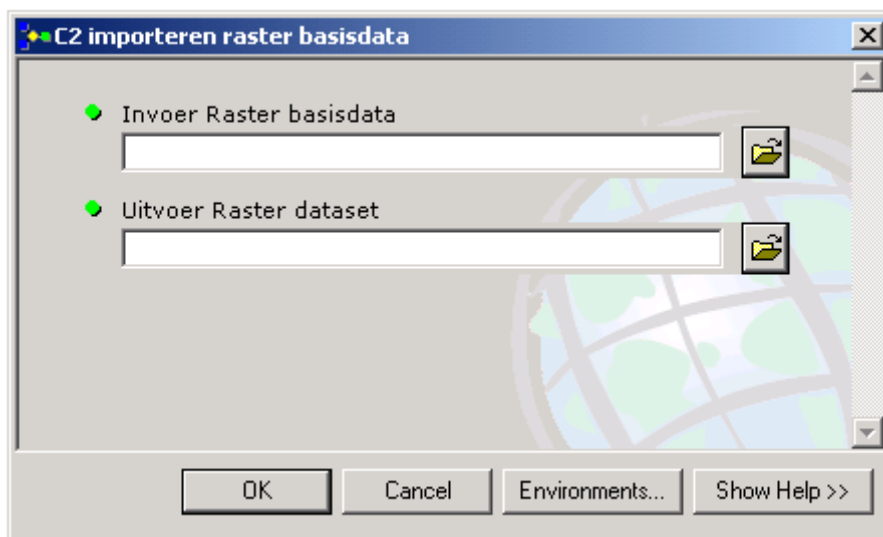
Process: Toevoegen features...

```
gp.Append_management("xtmplyr", Uitvoer_vector_dataset, "TEST", )
```



C2 import rasterdata

```
# -----  
--  
# C2 import raster.py  
# Created on: Thu Apr 15 2004 01:09:24 PM  
# (generated by ArcGIS/ModelBuilder)  
# Usage: C2 import raster <Invoer_Raster_basisdata>  
<Uitvoer_Raster_dataset>  
# -----  
--  
  
# Import system modules  
import sys, string, os, win32com.client  
  
# Create the Geoprocessor object  
gp = win32com.client.Dispatch("esriGeoprocessing.GpDispatch.1")  
  
# Load required toolboxes...  
gp.AddToolbox("C:/ArcGIS/ArcToolbox/Toolboxes/Data Management Tools.tbx")  
  
# Script arguments...  
Invoer_Raster_basisdata = sys.argv[1]  
  
Uitvoer_Raster_dataset = sys.argv[2]  
  
# Local variables...  
  
# Process: Importeren Raster basisbestand...  
gp.CopyRaster_management(Invoer_Raster_basisdata, Uitvoer_Raster_dataset,  
    "")
```



C3 conversie

```
# -----
--
# C3 conversie.py
# Created on: Thu Apr 15 2004 01:11:53 PM
# (generated by ArcGIS/ModelBuilder)
# Usage: C3 conversie <Invoer_dataset> <Vergrid-_attribuut> <Celgrootte>
<Uitvoer_aspect__dataset-_naam_> <Plangrens>
# -----
--

# Import system modules
import sys, string, os, win32com.client

# Create the Geoprocessor object
gp = win32com.client.Dispatch("esriGeoprocessing.GpDispatch.1")

# Load required toolboxes...
gp.AddToolbox("C:/ArcGIS/ArcToolbox/Toolboxes/Conversion Tools.tbx")

# Set the Geoprocessing environment...
gp.SAEnvExtent = "DEFAULT"
gp.outputZFlag = "Same As Input"
gp.outputZValue = ""
gp.outputMFlag = "Same As Input"
gp.outputSpatialRef = ""
gp.cluster_tolerance = ""
gp.defaultWorkspace = ""
gp.workspace = ""

# Script arguments...
Invoer_dataset = sys.argv[1]

Vergrid-_attribuut = sys.argv[2]

Celgrootte = sys.argv[3]

Uitvoer_aspect__dataset-_naam_ = sys.argv[4]

Plangrens = sys.argv[5]
if Plangrens == '#':
    Plangrens = "0.000000 0.000000 10.000000 10.000000" # provide a default
value if unspecified

# Local variables...
Uitvoer_aspect__datasetnaam_ = ""
Vergridattribuut = ""

# Process: Conversie ...
tempEnvironment0 = gp.SAEnvExtent
gp.SAEnvExtent = "0.000000 0.000000 10.000000 10.000000"
gp.FeatureToRaster_conversion(Invoer_dataset, Vergridattribuut,
Uitvoer_aspect__datasetnaam_, Celgrootte)
gp.SAEnvExtent = tempEnvironment0
```

C3 conversie

◆ Invoer dataset

◆ Vergridattribuut

Celgrootte (optional)

◆ Uitvoer aspect (datasetnaam)

Plangrens (optional)
As Specified Below

Top
10.000000

Left
0.000000

Right
10.000000

Bottom
0.000000

Snap Raster
<NONE>

OK Cancel Environments... Show Help >>

C3 bufferen in rasterformaat

```
# -----  
--  
# C3 buffer raster.py  
# Created on: Thu Apr 15 2004 01:11:38 PM  
# (generated by ArcGIS/ModelBuilder)  
# Usage: C3 buffer raster <Invoer_dataset> <Plangrens> <Celgrootte>  
<Uitvoer_aspect__datasetnaam__>  
# -----  
--  
  
# Import system modules  
import sys, string, os, win32com.client  
  
# Create the Geoprocessor object  
gp = win32com.client.Dispatch("esriGeoprocessing.GpDispatch.1")  
  
# Load required toolboxes...  
gp.AddToolbox("C:/ArcGIS/ArcToolbox/Toolboxes/Spatial Analyst Tools.tbx")  
  
# Script arguments...  
Invoer_dataset = sys.argv[1]  
  
Plangrens = sys.argv[2]  
  
Celgrootte = sys.argv[3]  
if Celgrootte == '#':  
    Celgrootte = "1" # provide a default value if unspecified  
  
Uitvoer_aspect__datasetnaam__ = sys.argv[4]  
if Uitvoer_aspect__datasetnaam__ == '#':  
    Uitvoer_aspect__datasetnaam__ = "M:/Implementatie/ArcMER9_11022004.mdb/" #  
    provide a default value if unspecified  
  
# Local variables...  
Uitvoer_richting__optioneel__ = ""  
  
# Process: Buffer in rasterformaat...  
tempEnvironment0 = gp.SAEnvCellSize  
gp.SAEnvCellSize = "1"  
tempEnvironment1 = gp.SAEnvExtent  
gp.SAEnvExtent = "DEFAULT"  
gp.EucDistance_sa(Invoer_dataset, Uitvoer_aspect__datasetnaam__, "", "1",  
    Uitvoer_richting__optioneel__)  
gp.SAEnvCellSize = tempEnvironment0  
gp.SAEnvExtent = tempEnvironment1
```

C3 buffer in rasterformaat

● Invoer dataset

Plangrens (optional)

Default

Top

Left Right

Bottom

Snap Raster

<NONE>

Celgrootte (optional)

As Specified Below

1.000000

Uitvoer aspect (datasetnaam)

M:\Implementatie\ArcMER9_11022004.mdb\

OK Cancel Environments... Show Help >>

C3 beoordelingsattribuut aanpassen

```
# -----  
--  
# C3 beoo aanp.py  
# Created on: Thu Apr 15 2004 01:10:09 PM  
# (generated by ArcGIS/ModelBuilder)  
# Usage: C3 beoo aanp <Invoer_raster_dataset> <Beoordelingsattribuut>  
<Uitvoer_aspect__datasetnaam_> <Plangrens> <Celgrootte>  
# -----  
--  
  
# Import system modules  
import sys, string, os, win32com.client  
  
# Create the Geoprocessor object  
gp = win32com.client.Dispatch("esriGeoprocessing.GpDispatch.1")  
  
# Load required toolboxes...  
gp.AddToolbox("C:/ArcGIS/ArcToolbox/Toolboxes/Spatial Analyst Tools.tbx")  
  
# Script arguments...  
Invoer_raster_dataset = sys.argv[1]  
  
Beoordelingsattribuut = sys.argv[2]  
  
Uitvoer_aspect__datasetnaam_ = sys.argv[3]  
  
Plangrens = sys.argv[4]  
  
Celgrootte = sys.argv[5]  
if Celgrootte == '#':  
    Celgrootte = "MAXOF" # provide a default value if unspecified  
  
# Local variables...  
Reclassification = "1326.000000 1;1326.000000 6445.000000 2;6445.000000  
7644.000000 3;7644.000000 8457.000000 4"  
  
# Process: Beoordelingsattribuut aanpassen ...  
tempEnvironment0 = gp.SAEnvCellSize  
gp.SAEnvCellSize = "MAXOF"  
tempEnvironment1 = gp.SAEnvExtent  
gp.SAEnvExtent = "DEFAULT"  
gp.Reclassify_sa(Invoer_raster_dataset, Beoordelingsattribuut,  
Reclassification, Uitvoer_aspect__datasetnaam_, "DATA")  
gp.SAEnvCellSize = tempEnvironment0  
gp.SAEnvExtent = tempEnvironment1
```


C3 beoordelingsattribuut aanpassen

• Invoer raster dataset
 

• Beoordelingsattribuut

• Uitvoer aspect (datasetnaam)
 

Plangrens (optional)
 

Top
Left Right
Bottom

Snap Raster
 

Celgrootte (optional)
 