

Q 3266

Opdrachtgever:

DG Rijkswaterstaat, RIKZ

Habitatinstrument

Technisch ontwerp

december 2002

	bibliotheek postbus 177 - 2600 MH Delft waterbouw, landbouw, natuur en milieu
BD	G1011
WL	Q3266
EXPL	WL Delft Hydraulics



C 143916

Opdrachtgever:

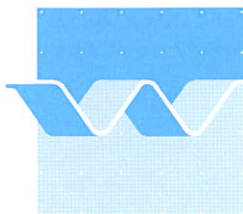
DG Rijkswaterstaat, RIKZ

Habitatinstrument

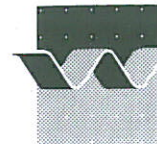
Technisch ontwerp

R. Brinkman, P.J.A. Gijsbers, M. Haasnoot,
R.R. van Kappel, G.E.M. van der Lee,

december 2002



wl | delft hydraulics



OPDRACHTGEVER: DG Rijkswaterstaat, RIKZ

TITEL: Technisch ontwerp modelinstrument Habitat

SAMENVATTING:

Deze rapportage geeft het technisch ontwerp voor een nieuw te ontwikkelen habitatinstrument zoals dat door WL, RIZA en RIKZ gebruikt zal gaan worden. Het technisch ontwerp geeft aan hoe dit instrument eruit moet gaan zien qua functionaliteiten en programmatische/technische uitwerking. In het ontwerp is opgenomen een presentatie van de lay-out en een beschrijving van het werkproces van de gebruiker.

REFERENTIES:

VER.	AUTEUR	DATUM	OPMERK.	REVIEW	GOEDKEURING
	Rob Brinkman	1-10-2002		Peter Gijsbers	Peter Glas
PROJECTNUMMER		Q3266			
TREFWOORDEN:		technisch ontwerp, habitat			
AANTAL BLADZIJDEN		44			
VERTROUWELIJK		☐ JA, tot (datum)		☒ NEE	
STATUS:		☐ VOORLOPIG	☐ CONCEPT	☒ DEFINITIEF	

Inhoud

1	Inleiding	1-1
1.1	Achtergrond	1-1
1.2	Uitgangspunten	1-1
2	Componenten	2-1
2.1	Taal en ontwikkel-omgeving	2-1
2.2	Externe componenten	2-1
3	Klassen	3-1
3.1	Overzicht.....	3-1
3.2	Habitat-relaties.....	3-2
3.3	Visualisatie.....	3-3
3.4	Algemene Eigenschappen.....	3-4
3.5	Klassenbeschrijving	3-4
3.5.1	Item.....	3-4
3.5.2	Project.....	3-5
3.5.3	ModelHolder.....	3-6
3.5.4	Case	3-6
3.5.5	RuleHolder.....	3-7
3.5.6	Model.....	3-8
3.5.7	Rule.....	3-9
3.5.8	Parameters.....	3-10
3.5.9	Parameter	3-10
3.5.10	Map.....	3-11
3.5.11	ValueParameter	3-11
3.5.12	TableParameter	3-11
3.5.13	Classification	3-12
3.5.14	Range	3-12
3.5.15	ProjectEditor.....	3-13
3.5.16	CaseEditor.....	3-13
3.5.17	ModelEditor	3-14
3.5.18	RuleEditor.....	3-15
3.5.19	RuleTable	3-16
3.5.20	ParameterTable	3-16

3.5.21	ClassificationTable	3-17
3.5.22	MapEngine	3-17
3.5.23	Batch	3-18
3.5.24	PCRaster.....	3-18
4	Applicatie	4-1
4.1	Werkwijze.....	4-1
4.2	Scher-opbouw	4-1
4.3	Runnen van een case	4-2
4.4	Browser	4-2
4.5	Acties.....	4-3
5	Bestandsformaten.....	5-1
5.1	Aansturingbestand voor PCRaster.....	5-1
5.2	Kaartformaat.....	5-1
5.3	Tabelformaat.....	5-1
6	User Interfaces.....	6-1
6.1	Project Editor.....	6-1
6.2	Case Editor	6-2
6.3	Model Editor	6-2
6.4	Rule Editor	6-3

I Inleiding

I.1 Achtergrond

Het huidige habitatmodelleninstrumentarium bestaat uit een verzameling modelinstrumenten die de afgelopen jaren door verschillende instituten vanuit verschillende vraagstellingen en voor verschillende toepassingen zijn ontwikkeld. Bij RIZA, RIKZ en WL bestaat de wens om een nieuw habitatinstrument te ontwikkelen dat alle gewenste functionaliteiten van bestaande modellen verenigt. Dit instrument moet een gezamenlijk produkt zijn en een standaardinstrument voor gebruikers van habitatmodellen in Nederland.

I.2 Uitgangspunten

De uitgangspunten van dit technisch ontwerp is het functionaal ontwerp. Dit technisch ontwerp zal uitsluitend de functionaliteit in fase 1 dekken.

2 Componenten

2.1 Taal en ontwikkel-omgeving

De applicatie zal ontwikkeld worden in Java. Er zal gebruik gemaakt worden van de JDK1.3.1.

Bij voorkeur wordt gebruik gemaakt van JBuilder 4. Dit vanwege de ervaring binnen het WL en de mogelijkheid grafisch user interfaces te ontwerpen.

De applicatie zal ActiveX-componenten bevatten. Deze zullen worden ingepakt in een Java-klasse met behulp van de Jacozoom library, die het WL in zijn bezit heeft. (zie www.jacozoom.de).

2.2 Externe componenten

De applicatie zal niet alleen met nieuw ontwikkelde componenten gerealiseerd worden, maar zal tevens gebruik maken van een aantal ingekochte of reeds bestaande componenten. Deze zullen niet of nauwelijks aangepast worden aan de behoeften van het Habitat-instrument.

Deze modules zijn:

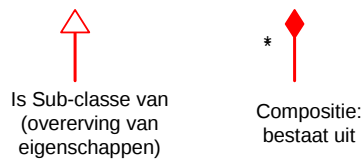
Module	Functionaliteit	Herkomst
PCRaster	Een rekenmodule die de GIS-berekening voor zijn rekening neemt	PES
Map Engine	OCX die een kaart visualiseert.	WL
DelftWise	Java library die data- en workflowmanagement realiseert.	WL
JEP	Java Expression Parser, een package dat in staat is formules te valideren en te berekenen. zie http://jep.sourceforge.net/	Freeware

3 Klassen

3.1 Overzicht

In deze applicatie wordt alle te ontwikkelen functionaliteit in Java-klassen geïmplementeerd. Deze Java-klassen worden weergegeven in de volgende klasse-diagrammen.

De klassen-diagrammen worden weergegeven in UML-stijl. Hierin worden overerving en relaties op de volgende manieren getoond.



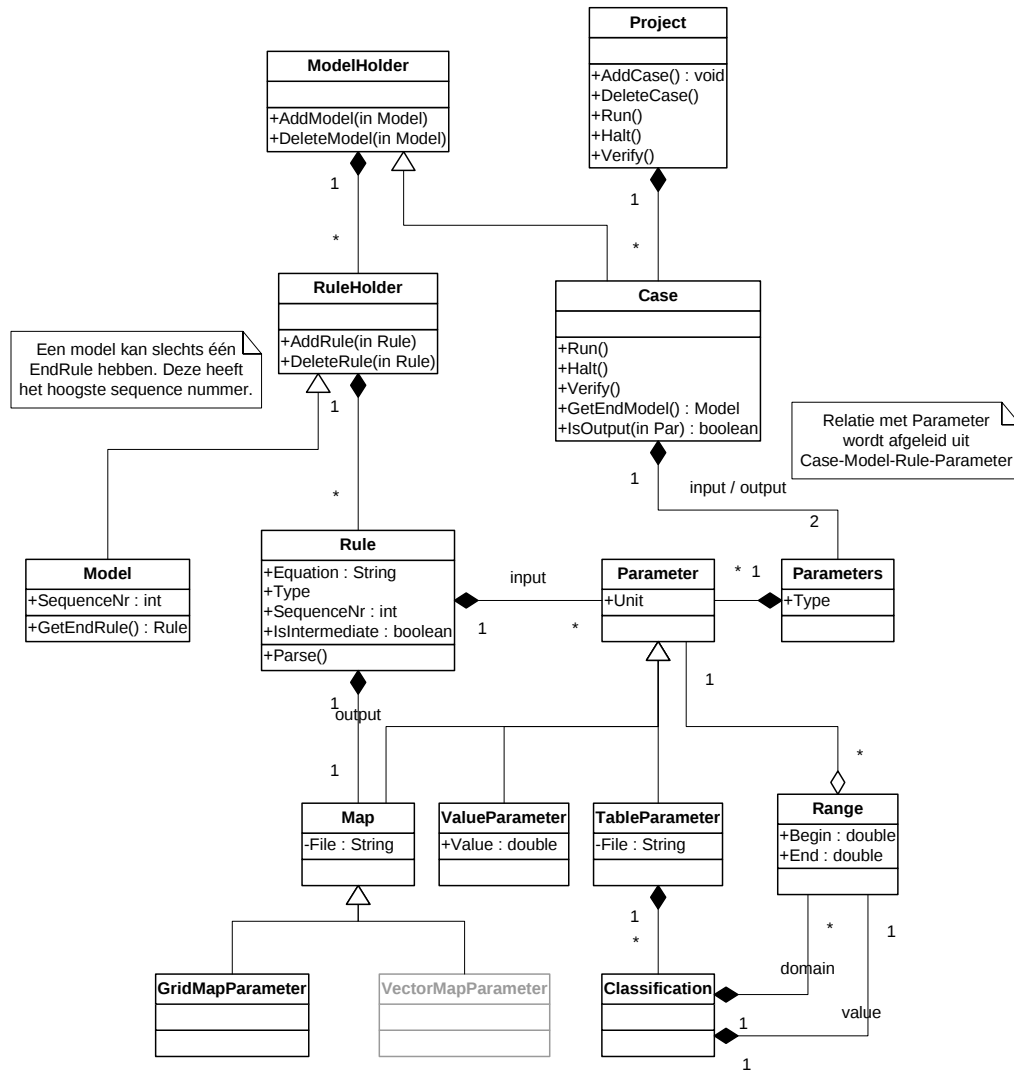
Er volgen drie diagrammen, te weten

- | | |
|--------------------------|--|
| • Habitat-relaties | Hoe liggen de relaties tussen de verschillende typen klassen |
| • Visualisatie | De editors / viewers van diverse typen klassen |
| • Algemene eigenschappen | Welke eigenschappen hebben de verschillende typen klassen met elkaar gemeenschappelijk |

De property's en methods van alle klassen worden in detail in paragraaf 3.4 "Klassenbeschrijving" besproken

3.2 Habitat-relaties

De relaties tussen de verschillende klassen zijn op de volgende manier gelegd.



Dit diagram beschrijft het volgende:

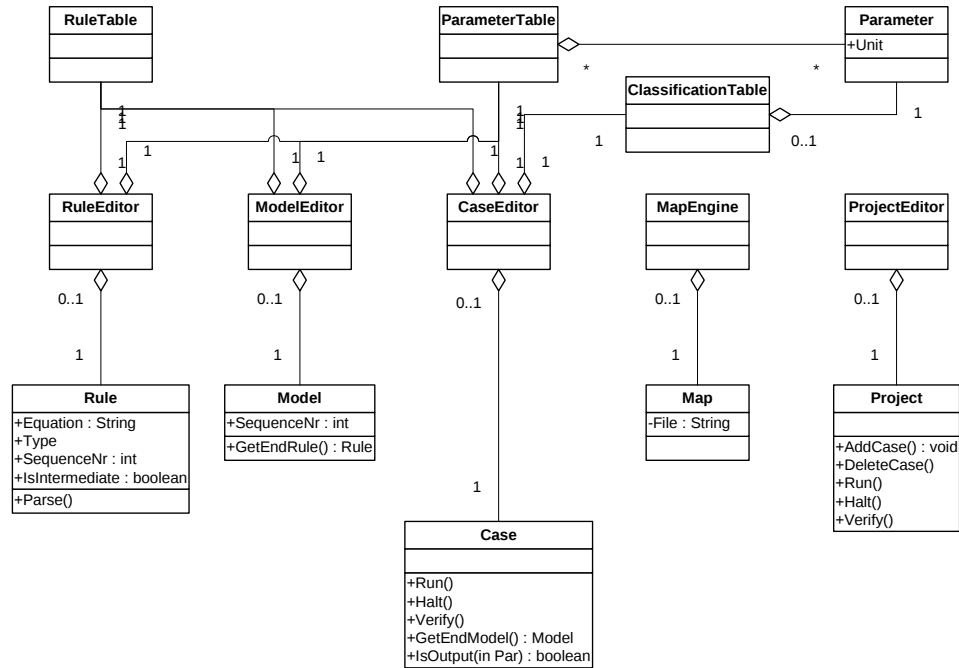
- Een project, het hoogste ingangspunt, bestaat uit een of meer cases.
- Een case bestaat uit een aantal modellen, die in een bepaalde volgorde staan.
- Ook de knowledge library (een ModelHolder) bevat een aantal modellen.
- Een model bestaat uit een aantal rules (kennisregels).
- Een rule heeft een aantal input-parameters en een output-parameter, die altijd een map is.
- Een parameter is of een map, of een table, of een value.
- Een table bestaat uit een aantal classificaties.
- Een classificatie geeft een waarde die hoort bij een bepaald domein van een parameter (bijvoorbeeld van $[0,1] \rightarrow 1$, $\langle 1,3 \rangle \rightarrow 2$). De waarde kan ook gegeven worden als

functie van domeinen van verschillende parameters. Daarnaast kan de eindwaarde zelf ook een bereik voorstellen; er dient dan lineair geïnterpoleerd te worden.

- Uit het voorgaande is af te leiden dat een case met een aantal parameters werkt, die hetzij als input worden gegeven, hetzij worden afgeleid.

3.3 Visualisatie

De verschillende klassen worden op de volgende wijze gevisualiseerd.

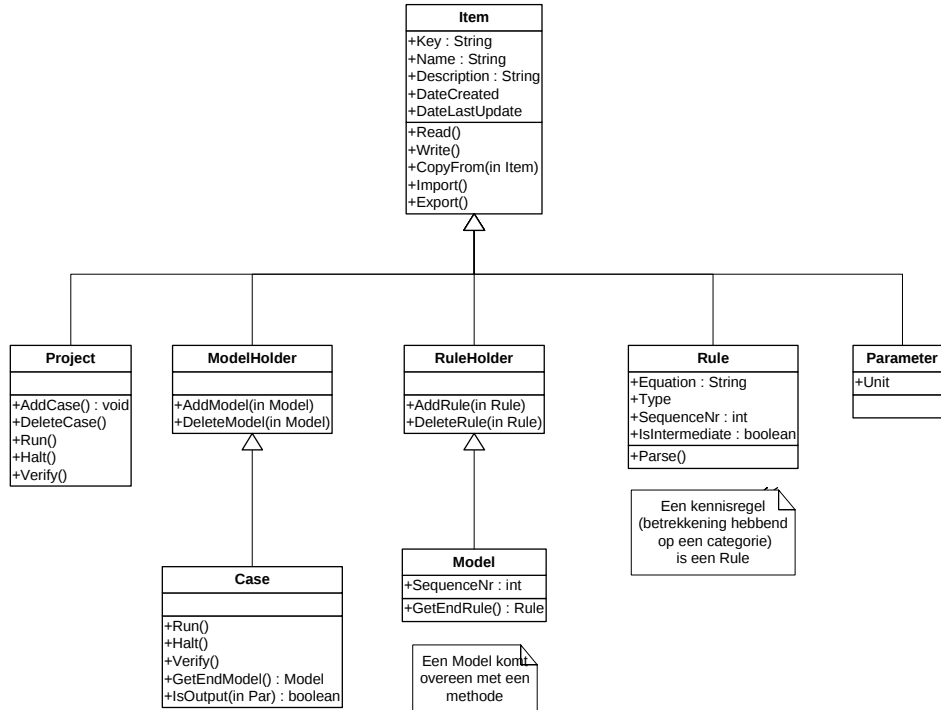


Dit diagram beschrijft het volgende:

- Een project, een case, een model en een rule hebben ieder een eigen type editor.
- De editors van de case, model en rule bevatten onder meer een tabel, waarin parameters ge-edit kunnen worden.
- De editors van de case, model en rule bevatten ook een tabel, waarin rules (de formules) ge-edit kunnen worden.

3.4 Algemene Eigenschappen

De volgende klassen spelen een rol binnen het Habitat-instrument



Dit diagram beschrijft het volgende:

- Een project, een model holder (een case of de knowlegde library), een model, een rule en een parameter hebben de eigenschappen van item gemeenschappelijk.

3.5 Klassenbeschrijving

De klassen worden in deze paragraaf in detail besproken.

3.5.1 Item

Algemeen

Naam	nl.wldelft.habitat.Item
Afgeleid van	nl.wldelft.dataserver.Item
Omschrijving	Basisklasse waarin de gemeenschappelijke eigenschappen van een habitat-klassen worden geïmplementeerd

Property's

Key	String	Unieke identificatie op opslag-niveau. Mag niet gewijzigd worden (is reeds geïmplementeerd in superklasse)
Name	String	Unieke identificatie op gebruikers-niveau. Mag niet gewijzigd worden (is reeds geïmplementeerd in superklasse)
Description	String	Door gebruiker ingevoerde vrije tekst
DateCreated	Date	Tijdstip van creatie
DateLastUpdate	Date	Tijdstip van laatste wijziging.

Methods

Read		Leest item uit opslag-systeem (is reeds geïmplementeerd in superklasse)
Write		Schrijft item in opslag-systeem (is reeds geïmplementeerd in superklasse)
CopyFrom	in Source	Copieert item-property's en alle gerelateerde items van source (is reeds geïmplementeerd in superklasse)
Import	in File	Leest een item, inclusief alle gerelateerde items, in uit een file
Export	in File	Schrijft een item, inclusief alle gerelateerde items, naar een file

3.5.2 Project

Algemeen

Naam	nl.wldelft.habitat.Project
Afgeleid van	nl.wldelft.workflowserver.Batch
Omschrijving	Een project is een verzameling bij elkaar horende cases.

Property's

Geen

Methods

AddCase	in Case	Voegt een case toe aan het project (is reeds geïmplementeerd in superklasse door addRelatedItem)
RemoveCase	in Case	Verwijdert een case uit het project (is reeds geïmplementeerd in superklasse door removeRelatedItem)
Import	in File	Leest een project in uit een file

Export	in File	Schrijft een project naar een file
Verify		Verifieert alle onderliggende cases
Run		Runt alle onderliggende cases (wordt geïmplementeerd in superklasse)
Halt		Stopt de run onmiddellijk (wordt geïmplementeerd in superklasse)

Refereert aan

Cases

3.5.3 ModelHolderAlgemeen

Naam	nl.wldelft.habitat.ModelHolder
Afgeleid van	nl.wldelft.habitat.Item
Omschrijving	Een model-holder bestaat uit een aantal modellen. De meest voorkomende toepassing van een ModelHolder is een Case. Daarnaast wordt een ModelHolder gebruikt voor de knowledge library.

Property's

geen

Methods

AddModel	in Model	Voegt een model toe (is reeds geïmplementeerd in superklasse door addRelatedItem)
RemoveModel	in Model	Verwijdert een model (is reeds geïmplementeerd in superklasse door removeRelatedItem)

Refereert aan

Models

3.5.4 CaseAlgemeen

Naam	nl.wldelft.habitat.Case
Afgeleid van	nl.wldelft.workflowserver.Case
Omschrijving	Een case stelt een bij elkaar horende verzameling gegevens, input en output, waarop een berekening kan worden toegepast.

Constructie

Bij het construeren van een case zal reeds een Model worden gecreëerd, zodat er altijd een end-model zal bestaan. Daarnaast zullen twee Parameters-object worden gecreëerd, die de invoer- en uitvoerparameters zullen gaan groeperen.

Property's

Geen

Methods

Run		Start de berekening op de achtergrond. (is reeds geïmplementeerd in superklasse)
Halt		Stopt de berekening onmiddellijk (is reeds geïmplementeerd in superklasse)
Verify		Verifieert of alle noodzakelijke invoergegevens op ieder moment in de berekening zijn gespecificeerd.
IsOutput	in Parameter out boolean	Geeft aan of een parameter als output van een rule voorkomt.

Refereert aan

Models

Parameters	Een lijst met alle parameters waarvoor input is verschuldigd. Dit zijn dus alle parameters, die niet als output van een rule voorkomen
Parameters	Een lijst met alle parameters waarvoor geen input is verschuldigd. Dit zijn dus alle parameters, die tenminste een keer als output van een rule voorkomen.

3.5.5 RuleHolder

Algemeen

Naam	nl.wldelft.habitat.RuleHolder
Afgeleid van	nl.wldelft.habitat.Item
Omschrijving	Een RuleHolder bestaat uit een aantal Rules. Meestal wordt een Model geïntantieerd om een aantal rules te bevatten. Een voorbeeld van een RuleHolder dat geen Model is, is een ectopen-classificatie. Deze bevat immers ook een aantal rules.

Property's

Geen

Methods

AddRule	in	Model	Voegt een rule toe (is reeds geïmplementeerd in superklasse door addRelatedItem)
RemoveRule	in	Model	Verwijdert een rule (is reeds geïmplementeerd in superklasse door removeRelatedItem)

Refereert aan

Rules

3.5.6 ModelAlgemeen

Naam	nl.wldelft.habitat.Model
Afgeleid van	nl.wldelft.habitat.RuleHolder
Omschrijving	Een model stelt een methode voor. Een model kan al dan niet een ecotopenclassificatie voorstellen. Dit houdt het volgende in: • Geen ecotopenclassificatie; er kunnen een onbeperkt aantal rules worden toegevoegd. • Ecotopenclassificatie; er kan slechts één rule worden toegevoegd en die is van het type multiclassificering.

Constructie

Bij het construeren van dit object zal reeds een Rule worden gecreëerd, zodat er altijd een end-rule zal bestaan.

Property's

Sequence	int	Volgnummer van het model in de PCRaster berekening. De eerste heeft nummer 1, daarna wordt het nummer telkens met 1 opgehoogd. Het volgnummer wordt afgeleid uit de positie van het model in de modellenlijst in een case.
----------	-----	---

Methods

GetEndRule	out Rule	Geeft de rule die het laatste door de rekenmodule geëxecuteerd zal worden. De output-parameter van deze Rule zal dus het eindresultaat van het Model zijn.
------------	----------	---

3.5.7 Rule

Algemeen

Naam	nl.wldelft.habitat.Rule
Afgeleid van	nl.wldelft.habitat.Item
Omschrijving	Een rule stelt een kennisregel voor.

Property's

Equation	String	Formule die de kennisregel voorstelt.
Type	enum	Classificering, gebroken lineair, univariant of multivariant, multi-classificering; De eerste vier kunnen voorkomen bij een modelcase, de laatste bij een ecotopencase. - Een classificering is van de volgende vorm: $y := C * x$, waarbij x en y kaarten zijn en C een tabel, bijvoorbeeld: $[0,1] \rightarrow 0$; $[1,5] \rightarrow 1$; $[5, \rightarrow] \rightarrow 0$; - Een gebroken lineaire rule heeft dezelfde vorm, alleen wordt het mogelijk een bereik op te geven aan de resulterende kant, bijvoorbeeld: $[0,1] \rightarrow [2,0] \text{ (0,25 resulteert in 1,5)}$; - Univariant is in de vorm van $y := \sum a_n x^n$, alle a en x kaarten. - Multivariant is in de vorm van $y := \sum a_n x^n + \sum b_n z^n$, alle a, b, x en z kaarten. - Multiclassificering is een uitbreiding op classificering volgens $y := C(x_1, x_2, x_3, \dots)$, waarbij x kaarten zijn en C een tabel, bijvoorbeeld: $([0,1], [0,2], [0,1]) \rightarrow 0$; $([1,2], [0,2], [0,1]) \rightarrow 1$; $([1,2], [0,2], [1,2]) \rightarrow 2$;
Sequence	int	Volgnummer van de rule binnen het model in de PCRaster berekening. De eerste heeft nummer 1, daarna wordt het nummer telkens met 1 opgehoogd. Het volgnummer wordt afgeleid uit de positie van de rule in de lijst met rules in het model.

Methods

Parse	Valideert de equation en, indien valide, werkt de interne lijst met parameters bij. Er wordt gebruik gemaakt van JEP-software.
-------	--

Refereert aan

Parameter	Een lijst met input parameters, dus de parameters die aan de rechterzijde van de vergelijking voorkomen.
Map	De te berekenen kaart, dus de parameter die aan de linkerzijde van de vergelijking voorkomt.

3.5.8 Parameters

Algemeen

Naam	nl.wldelft.habitat.Parameters
Afgeleid van	nl.wldelft.dataserver.Items
Omschrijving	Groepering van een aantal parameters. Deze klasse wordt gebruikt om de invoer- en uitvoer-parameters van een case te groeperen. De interne lijst met parameter-objecten zal altijd up-to-date gehouden worden.

Property's

Type	String	Invoer of uitvoer.
------	--------	--------------------

Methods

Geen

Refereert aan

Parameter	Een lijst met parameters.
-----------	---------------------------

3.5.9 Parameter

Algemeen

Naam	nl.wldelft.habitat.Parameter
Afgeleid van	nl.wldelft.habitat.Item
Omschrijving	Een parameter is een invoer- of berekende waarde, hetzij een kaart, hetzij een tabel, hetzij een enkele waarde.

Property's

Unit	String	Beschrijving van de eenheid. Deze waarde wordt buiten informatieve doeleinden niet gebruikt.
------	--------	--

Methods

Geen

3.5.10 Map

Algemeen

Naam	nl.wldelft.habitat.Map
Afgeleid van	nl.wldelft.habitat.Item
Omschrijving	Een invoer- of uitvoerkaart.

Property's

File	String	Relatief pad naar het bestand dat de file bevat in BIL-formaat.
------	--------	---

Methods

Geen

3.5.11 ValueParameter

Algemeen

Naam	nl.wldelft.habitat.ValueParameter
Afgeleid van	nl.wldelft.habitat.Item
Omschrijving	Een enkelvoudige invoerwaarde

Property's

Value	float	De waarde
-------	-------	-----------

Methods

Geen

3.5.12 TableParameter

Algemeen

Naam	nl.wldelft.habitat.TableParameter
Afgeleid van	nl.wldelft.habitat.Item
Omschrijving	Een tabel met invoergegevens

Property's

File	String	Relatief pad naar het bestand dat de file bevat in FMT-formaat.
------	--------	---

Methods

Geen

3.5.13 ClassificationAlgemeen

Naam	nl.wldelft.habitat.Classification
Afgeleid van	nl.wldelft.dataserver.Item
Omschrijving	Een set domeinen van een aantal parameters met daarbij behorend een resulterende eindwaarde. Komt overeen met een regel in de ClassificationTable, bijvoorbeeld: ([0,1>,[0,2>,[0,1>) → 0;

Property's

Geen

Methods

Geen

Refereert aan

Domain	Domein van beginwaarden waarvoor de eindwaarde geldig is
Value	Resulterende eindwaarde; als er slechts één domein is opgegeven, dan kan de eindwaarde gedefinieerd zijn als een range waarbij lineair geïnterpoleerd dient te worden.

3.5.14 RangeAlgemeen

Naam	nl.wldelft.habitat.Range
Afgeleid van	nl.wldelft.dataserver.Item
Omschrijving	Definieert een domein waarden van een parameter, bijvoorbeeld [0,1] zijn alle waarden tussen 0 en 1 inclusief.

Property's

Begin	double	Beginwaarde
End	double	Eindwaarde

Methods

Geen

Refereert aan

Parameter Verwijst naar een parameter waarop dit domein van toepassing is.

3.5.15 ProjectEditorAlgemeen

Naam	nl.wldelft.habitat.ProjectEditor	
Afgeleid van	nl.wldelft.shell.ShellObject	
Omschrijving	Een editor waarin alle zaken op project-niveau bekeken en gewijzigd kunnen worden.	

Property's

Item	Item	Het project dat ge-edit wordt (is reeds geïmplementeerd in superklaase)
------	------	---

Methods

Execute	in actie	Voert een actie uit die via het menu, de toolbar of het popup-menu is binnengekomen.
---------	----------	--

Schermbouw

Text	JTextArea	Veld waarin tekst vrijelijk kan worden ingevoerd
------	-----------	--

3.5.16 CaseEditorAlgemeen

Naam	nl.wldelft.habitat.CaseEditor	
Afgeleid van	nl.wldelft.shell.CaseObject	
Omschrijving	Een editor waarin alle zaken op case-niveau bekeken en gewijzigd kunnen worden.	

Property's

Item	Item	De case die ge-edit wordt (is reeds geïmplementeerd in superklaase)
------	------	---

Methods

Execute	in actie	Voert een actie uit die via het menu, de toolbar of het popup-menu is binnengekomen.
---------	----------	--

Schermbouw

Grid	Parameter-Table	Tabel met alle parameters waarvoor een invoer vereist is.
Case Rules	RuleTable	Editor voor een of meerdere rules waarmee het eindresultaat van de case wordt gedefinieerd. Deze rules worden vervat in een speciaal daartoe gecreëerd eindmodel.

3.5.17 ModelEditor

Algemeen

Naam	nl.wldelft.habitat.ModelEditor
Afgeleid van	nl.wldelft.shell.ShellObject
Omschrijving	Een editor waarin alle zaken op model-niveau bekeken en gewijzigd kunnen worden.

Property's

Item	Item	Het model dat ge-edit wordt (is reeds geïmplementeerd in superklaase)
------	------	---

Methods

Execute	in actie	Voert een actie uit die via het menu, de toolbar of het popup-menu is binnengekomen.
---------	----------	--

Schermbouw

Model Rules	RuleTable	Editor voor een of meerdere rules waarmee het resultaat van de rule wordt gedefinieerd.
Grid	Parameter-Table	Tabel met alle parameters binnen het model waarvoor een invoer vereist is.

Acties

Opslaan	Slaat het model op
Toevoegen	Voegt een rule toe in de ModelEditor
Wissen	Verwijdert de geselecteerde rule uit de ModelEditor

3.5.18 RuleEditor

Algemeen

Naam	nl.wldelft.habitat.RuleEditor
Afgeleid van	nl.wldelft.shell.ShellObject
Omschrijving	Een editor waarin een rule ge-edit kan worden.

Property's

Item	Item	Het model dat ge-edit wordt (is reeds geïmplementeerd in superklaase)
------	------	---

Methods

Execute	in actie	Voert een actie uit die via het menu, de toolbar of het popup-menu is binnengekomen.
---------	----------	--

Schermpopbouw

Soort	JComboBox	Selectie mechanisme waarmee de soort rule gekozen kan worden (classificering, gebroken lineair, univariant, multivariant)
Model Rules Table Editor	RuleTable JTable	Editor voor een formule. Editor waarmee de waarden in een bepaald domein vastgelegd kunnen worden, herzij als vaste waarde hetzij lineair te interpoleren tussen een begin- en eindwaarde.
Grid	Parameter-Table	Tabel met alle parameters binnen de rule waarvoor een invoer vereist is.
RulePreviewer	JEP-routine	Toont een grafiek van de ingevoerde rule als functie van een van de invoerparameters.

Acties

Opslaan	Slaat het model op
Toevoegen	Voegt een rule toe in de ModelEditor
Wissen	Verwijdert de geselecteerde rule uit de ModelEditor

3.5.19 RuleTable

Algemeen

Naam	nl.wldelft.habitat.RuleTable
Afgeleid van	javax.swing.JTable
Omschrijving	Een tabel waarin formules ge-edit kunnen worden. Per regel in de tabel wordt de linker- en de rechterzijde van de vergelijking ingevoerd. Men kan zo'n veld pas verlaten als de formule syntactisch gezien correct is.

Property's

Rules	Items	De rules die ge-edit worden.
-------	-------	------------------------------

Methods

Add	in Rule	Voegt een rule toe aan de tabel.
Remove	in Rule	Verwijdert een rule uit de tabel.

3.5.20 ParameterTable

Algemeen

Naam	nl.wldelft.habitat.ParameterTable
Afgeleid van	javax.swing.JTable
Omschrijving	Een tabel waarin parameters ge-edit kunnen worden. Per parameter worden een aantal zaken weergegeven: een beschrijving, de eenheid en het type (waarde, tabel of kaart). Per parameter kan een invoerbestand worden aangegeven, als tekst of met browsen). Dit bestand kan op een andere machine staan en zelfs op een Unix-machine.

Property's

Parameters	Parameters	De parameters die ge-edit worden.
------------	------------	-----------------------------------

Methods

Geen

3.5.21 ClassificationTable

Algemeen

Naam	nl.wldelft.habitat.ClassificationTable
Afgeleid van	javax.swing.JTable
Omschrijving	Een grid waarin een tabel ge-edit kan worden. Een tabel kan voorkomen in een Rule van het type classificering, gebroken lineair en multi-classificering.

Property's

Parameter	TableParameter	De parameter waaronder alle classificaties vallen
-----------	----------------	---

Methods

Geen

3.5.22 MapEngine

Algemeen

Naam	nl.wldelft.habitat.MapEditor
Afgeleid van	nl.wldelft.control.ActiveX
Omschrijving	Wrapper rond de MapEngine (netter.ocx).

Property's

File	File	De map die getoond wordt, in bil- of asc-formaat. Het bestand kan op een andere (Unix) machine staan. In dat geval moet het bestand toch getoond worden, eventueel na copy naar de huidige machine.
------	------	---

Methods

De huidige functionaliteit van de Map Engine zal uitgebreid worden zodat de volgende functies mogelijk zijn: inzoomen, toevoegen van tekstbijschrift, een schaalbalk, een noordpijl en RD- of andere cöördinaatkruispunten en het visualiseren/afdrukken van 1-4 kaarten per blad

3.5.23 Batch

Algemeen

Naam	nl.wldelft.workflowserver.Batch
Afgeleid van	nl.wldelft.workflowserver.Activity
Omschrijving	Runt een aantal cases in een batch, dat wil zeggen een voor een. Er wordt geen onderlinge afhankelijkheid tussen de cases aangenomen.

Property's

Items	Items	De cases die gerund zullen gaan worden.
-------	-------	---

Methods

Run	Start de berekening op de achtergrond. (is reeds geïmplementeerd in superklasse)
Halt	Stopt de berekening onmiddellijk (is reeds geïmplementeerd in superklasse)

3.5.24 PCRaster

Algemeen

Naam	nl.wldelft.habitat.PCRaster
Afgeleid van	nl.wldelft.workflowserver.WorkflowRunnable
Omschrijving	Wrapper rond een dll die de PCRaster-berekening omvat.

Property's

Progress	float	Geeft de voortgang van de berekening aan als een getal tussen de 0 en de 1.
----------	-------	---

Methods

Run		Start de berekening op. Eerst wordt het besturingsbestand gegenereerd en vervolgens wordt dit aan de PCRaster-dll doorgegeven.
UseItem	Item	Wordt aangeroepen door de workflowserver om deze klasse te vertellen welke objecten hij als invoer en als uitvoer moet beschouwen. Als invoer zal de case worden meegegeven, als uitvoer de te genereren kaarten.

4 Applicatie

Dit hoofdstuk beschrijft hoe de afzonderlijke onderdelen worden samengesteld tot een eindapplicatie.

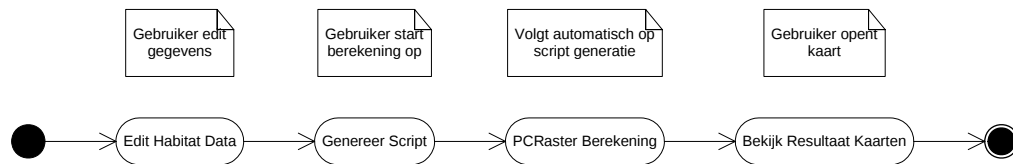
4.1 Werkwijze

In het algemeen zal de werkwijze zodanig zijn dat eerst de invoerdata wordt ge-edit, vervolgens de berekening wordt opgestart en tenslotte de resultaten worden bekenen.

In de habitat-applicatie betekent dit dat eerst projecten, cases, modellen, parameters en kennisregels worden ingevoerd. Vervolgens wordt de berekening opgestart en tenslotte worden de gegenereerde kaarten geopend.

De berekening opstarten is een tweestartsraket. Eerst wordt een script gegenereerd welke door PCRaster wordt gebruikt als invoer.

Het bovenstaande wordt in de onderstaande figuur gevisualiseerd.



4.2 Scherm-opbouw

De applicatie wordt opgebouwd met DelftShell, onderdeel van de DelftWise library. Dit leidt tot de volgende opbouw van het scherm:

1. Aan de linkerkant van de applicatie wordt een browser getoond. Dit is een boomstructuur waarin alle relevante objecten binnen de applicatie worden getoond.
2. Aan de onderkant van de applicatie worden informatie- en foutmeldingen getoond. Foutmeldingen dienen daartoe in de log stream 'Error' te worden weggeschreven. Voor informatie geldt dat deze in de log stream 'Information' dient te worden weggeschreven.
3. In het overige deel (het grootste deel) worden de documenten getoond. Dit zijn visualisaties van objecten, bijvoorbeeld een case-editor, een model-editor, een rule-editor of een map-viewer. Er kunnen meerdere documenten gelijktijdig geopend zijn, maar er is er altijd één actief.

4.3 Runnen van een case

Het runnen van een case wordt overgelaten aan de WorkflowServer, onderdeel van de DelftWise library. De case-editor wordt gebaseerd op de DelftShell-klasse die speciaal is ontworpen om een case te visualiseren. Dit leidt tot de volgende functionaliteit:

1. Berekeningen worden in achtergrond opgestart. Gedurende een berekening kan de gebruiker dus ‘andere dingen’ met de applicatie doen.
2. Een berekening kan op ieder moment afgebroken worden.
3. De voortgang van een case wordt middels een progress-bar gevisualiseerd.
4. De beschikbaarheid van data (bijv. maps) wordt bijgehouden. Een map is bijvoorbeeld niet beschikbaar als de berekening loopt of als de gerelateerde invoerdata is gewijzigd. Invoerschermen en de browser reageren op gepaste wijze, bijvoorbeeld middels het enablen/disablen van editors.

Om de workflowserver goed te laten runnen, dient een workflowscript te worden geschreven, waarin de te runnen taken staan. In het habitat-instrument staat hierin slechts de PCRaster-klasse, met als invoer de case en als uitvoer de case-uitvoer.

4.4 Browser

De browser toont alle in deze applicatie relevante data en reflecteert hun onderlinge samenhang. De browser wordt gefaciliteert door de dataserver, onderdeel van de DelftWise-library. Door gebruik te maken van de dataserver wordt in de volgende functionaliteit voorzien:

1. De browser reflecteert altijd de actuele toestand van de objecten. Als bijvoorbeeld aan een project een case wordt toegevoegd, dan wordt dit onmiddellijk, zonder extra code, getoond door de browser.

Ieder object wordt getoond als aparte node. Een node stelt ook altijd een object voor. Consequentie is wel dat voor header nodes (nodes die alleen gebruikt worden om een aantal onderliggende nodes te groeperen) aparte objecten gecreëerd moeten worden. De onderlinge samenhang tussen de verschillende typen objecten die getoond worden in de browser, is als volgt:

Eerst volgt het deel dat de cases en aanverwante zaken bevat.

Node	Parent	Klasse ¹	Aantal	Omschrijving
Root	-	dataserver.Database	1	Topnode in de boomstructuur
Projecten	Root	habitat.Projecten	1	Header node van alle projecten
Project	Projecten	habitat.Project	0..n	Een habitat-project
Case	Project	habitat.Case	0..n	Een habitat-case
Model	Case	habitat.Model	1..n	Een habitat-model
Rule	Model	habitat.Rule	1..n	Een habitat-rule
Parameters	Case	habitat.Parameters	2	Header voor invoer- en uitvoerkaarten van een case
Map	Parameters	habitat.Map	0..n	Een invoer- of uitvoerkaart van een case.

Nu volgt het deel met voorgedefinieerde gegevens.

Node	Parent	Klasse	Aantal	Omschrijving
Knowlegde Library	Root	habitat.ModelHolder	1	Bibliotheek van sjabloon-modellen
Sjabloon-Model	Knowlegde Library	habitat.Model	0..n	Sjabloon model
Ecotoop Library	Root	habitat.RuleHolder	1	Ecotopen-classificatie, bevat een aantal voorgedefinieerde rules.
Rule	Sjabloon-Model	habitat.Rule	0..n	Rule in een sjabloon-model
Rule	Ecotoop Library	habitat.Rule	0..n	Rule in de ecotoop-library

4.5 Acties

Binnen de applicatie kan de gebruiker een aantal acties uitvoeren. Deze worden opgestart via met menu, de toolbar en het popup-menu.

De volgende acties hebben betrekking op de browser. In de browser, die een boomstructuur heeft met alle in het habitatinstrument relevante objecten, heeft één object de focus. Deze heeft een blauwe achtergrond., conform de Windows-standaard. De uit te voeren actie is afhankelijk van het type van het gefocuseerde object.

Actie	Menu	Focus	Effect
Open	Bestand	Project Case Model Rule Map	Opent een case/model/rule/map in een nieuwe editor, indien nog niet geopend. Brengt anders de editor van het gefocuseerde object naar de voorgrond (geïmplementeerd in superklasse). Een map wordt getoond met de mapper-ocx.
Nieuw	Bestand	Root	Maakt een nieuw object aan. Er wordt een nieuw

¹ Wordt voorafgegaan door nl.wldelft.

Actie	Menu	Focus	Effect
		Project Case	'child' van de focus aangemaakt, dus als de Focus de Root is, een project, als de focus een project is, en case en als de focus een case of de knowledge library is, een model. Bij deze actie wordt een dialoog opgestart waarbij de gebruiker een unieke naam moet kiezen. Bij een case en een model kan hij een sjabloon-model kiezen, waarbij het model of het eerste model van de case gebaseerd wordt op dat sjabloon. Het baseren op een sjabloon betekent dat het sjabloon wordt gecopieerd naar het model, inclusief alle onderliggende rules en parameters. Bij het creëren van een case heeft de gebruiker de keuze of hij een modellenncase of een ecotopencase wil creëren.
Verwijderen	Bestand	Project Case Model	Verwijdert het object.

De volgende acties hebben betrekking op het actieve document.

Actie	Menu	Document	Effect
Opslaan	Bestand	Case Project	Slaat het object op. Bij een case worden ook alle onderliggende objecten, dus models, parameters en berekende kaarten, opgeslagen.
Opslaan als	Bestand	Case Project	Slaat een copie van het object op onder een andere naam, waartoe een dialoog wordt opgestart. Het object onder de oorspronkelijke naam blijft ongewijzigd. In het actieve document wordt vervolgens doorgewerkt met de copie van het object.
Opslaan als sjabloon	Bestand	Model	Slaat een copie van het model op in de knowledge library.
Opslaan als algemeen sjabloon	Bestand	Model Map	Slaat een copie van het model op in de algemene knowledge library.
Eigen-schappen	Bestand	Project, Case, Model, Rule	Tont een dialoog waarin de naam van het object gewijzigd kan worden en waarmee de date-created en date-last-update uitgelezen kunnen worden.
Importeren	Bestand	Project Case	Importeert een project/case van file. Middels een file-browser kan de gebruiker aangeven van welk bestand er geïmporteerd dient te worden. Er wordt geïmporteerd van XML-formaat.
Exporteren	Bestand	Project Case Map	Exporteert een project/case naar een file. Middels een file-browser kan de gebruiker aangeven naar welk bestand er ge-exporteerd dient te worden. Bij een project/case wordt er geëxporteed naar XML-formaat. Bij een map naar ASC, BMP- of JPEG-formaat.

Afdrukken	Bestand	Map	Print de kaart.
Toevoegen	Bestand	Map	Indien de focus in de browser op een map staat, dan wordt deze toegevoegd aan de kaart-visualisatie. Op deze manier kan dus een gecombineerde kaart worden opgebouwd.
Zoom In	Beeld	Map	Zoomt in op de kaart
Zoom Out	Beeld	Map	Zoomt uit.
Verplaatsen	Beeld	Map	Verplaatst de kaart middels een 'handje'.
Start	Bereken	Project Case	Start de berekening op, bij een project van alle onderliggende cases achter elkaar. Er dient rekening gehouden te worden met het feit dat PCRaster niet-reentrant is, dus dat er maar een berekening gelijktijdig kan lopen. (te implementeren in workfloserver)
Stop	Bereken	Project Case	Stopt de berekening onmiddellijk
Verifiëren	Bereken	Project Case	Verifieert of de berekening uitgevoerd kan worden.
Toevoegen	Bewerken	Case Model	Voegt een rule toe aan de case/model
Wissen	Bewerken	Case Model	Verwijdert de geselecteerde rule uit de case/model.

De volgende acties bestaan in de applicatie, maar hebben geen betrekking op de focus in de browser of op het actieve document.

Actie	Menu	Effect
Einde	Bestand	Beëindigt de applicatie
Beeld	Tabs/Panels	Zet de presentatie-wijze van de documenten in tabbladen of in een gesplitst panel.
Beeld	Volgende/Vorige	Activeert het volgende c.q. vorige document.
Copiëren	Bewerken	Copieert de geselecteerde tekst / tabelcellen naar het clipboard. Cellen in Excel formaat
Plakken	Bewerken	Copieert het clipboard naar de geselecteerde tekst / tabelcellen. Cellen in Excel formaat
Help	Help	Start on-line help op
Help	Info	Geeft versie- en copyright-informatie

5 Bestandsformaten

5.1 Aansturingbestand voor PCRaster

De informatie in de database behorende bij de selectie wordt beschreven in een 'case-file' in XML format. XML is een standaard beschrijving van informatie die inleesbaar is in andere pakketten. Deze case-file in XML wordt vervolgens doorgegeven aan het rekenhart. Dit doorgeven bestaat uit een vertaling van de XML beschrijving naar een modelscript voor de betreffende case. Deze scripts worden doorgerekend door het rekenhart te activeren. Na de berekening zijn de resultaten beschikbaar voor visualisatie en verdere bewerking.

5.2 Kaartformaat

Voor dynamische grids worden de volgende twee formaten ondersteund:

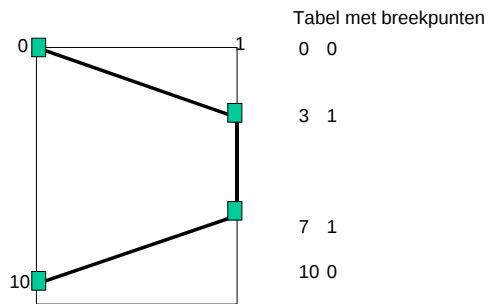
1. Het dynamische BIL formaat (*.bil, industry standard) : Je slaat voor iedere tijdstap de data voor alle cellen op. Dat kost veel ruimte, maar je kan wel de echte resultaten opslaan (en dus weer terughalen)
2. Incrementeel formaat (*.inc): Je slaat alleen veranderingen op. Dat is zoiets als een compressed AVI file, maar naast dat er tijdseries in kunnen worden opgeslagen, kunnen er ook de tijdseries van meerdere grids in worden opgeslagen. Echter veranderingen worden gedefinieerd door klassen te definiëren en een verandering is gedefinieerd als een verandering van klasse. In de file zijn alleen de indices van de klassen opgeslagen. Ideaal formaat als slechts een beperkt deel van de cellen verandert (bijvoorbeeld overtrekkende wolken, golffronten etcetera), nadeel je bent de echte data kwijt, je hebt alleen nog een referentie naar een klasse (waarvan je uiteraard het de grenzen wel van kent)

Ten behoeve van import en export wordt het Arcview (*.asc) formaat volledig ondersteund.

5.3 Tabelformaat

Parameters kunnen behalve kaarten en enkelvoudige waarden ook bestaan uit tabellen. Ten behoeve van blok- en lineaire tabellen wordt het FMT-formaat, dat momenteel in PCRaster wordt gebruikt, ondersteund. Hiermee kunnen tabellen met breekpunten, zoals in de onderstaande tabel, vastgelegd worden.

Tabel met breekpunten:



Blokfunctie:

Klasse	Bodemtype	Geschiktheid
1	Klei	0.1
2	Zand	0.5
3	Slib	1
4	Veen	0.7

Ecotoopclassificatietabel:

In de ecotoopclassificatietabel staan de klassegrenzen of klassecodes van verschillende kaarten gegeven op basis waarvan de klassen van de ecotopenkaart ingedeeld worden.

Diepte	Bodemklasse	Ecotoopklasse
[0,2>	1 'Klei'	1 'Ondiep water met kleibodem'
[2,5>	1 'Klei'	2 'Matig diep water met kleibodem'
[0,2>	2 'Zand'	3 'Ondiep water met zandbodem'
[2,5>	2 'Zand'	4 'Matig diep water met kleibodem'

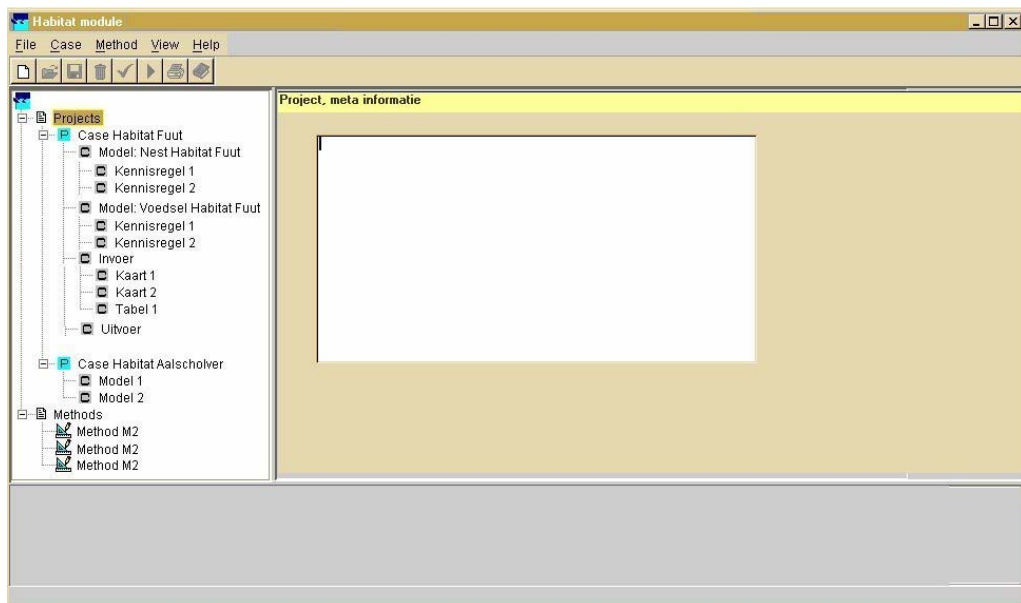
6 User Interfaces

De volgende screendumps geven een idee van de layout van de verschillende editors. Inhet algemeen is er het volgende over te zeggen.

1. Aan de bovenzijde staat een menu en een knoppenbalk. Hiermee worden acties uitgevoerd op het actieve document en de browser.
2. Aan de linkerzijde is een boomstructuur met alle mogelijke objecten in de habitat-applicatie. Hiervandaan kunnen deze geopend worden.
3. Aan de benedenzijde is ruimte voor logging en foutmeldingen.
4. Aan de rechterzijde, het grootste gedeelte, is ruimte voor een aantal editors voor projecten, cases, modellen, kennisregels, kaarten en tabellen. Deze worden in verschillende containers of in tabbladen naast elkaar gezet. Er is er altijd precies één actief; hierop hebben acties vanuit het menu betrekking.

6.1 Project Editor

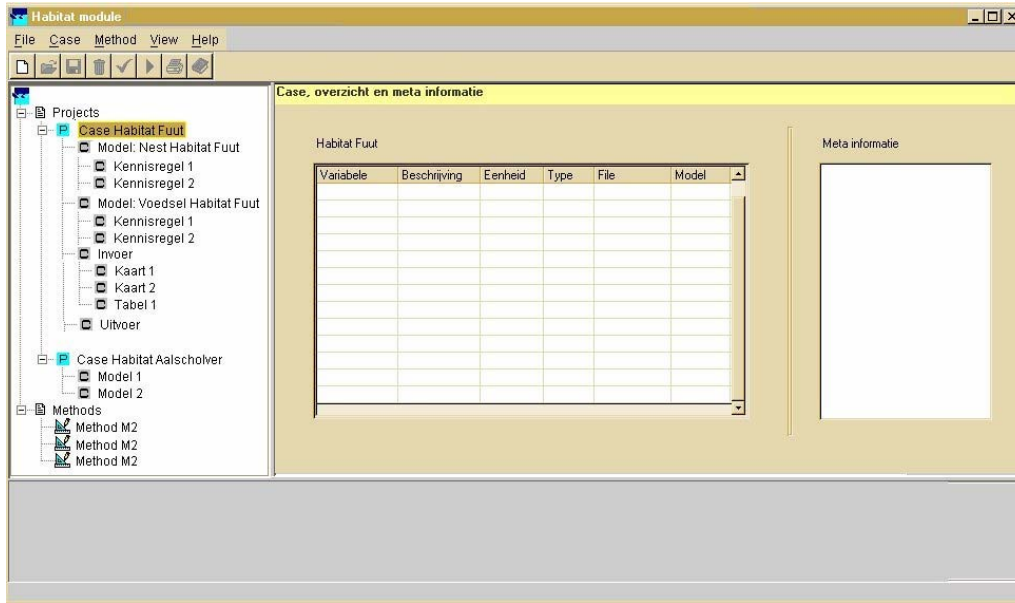
De project editor bevat slechts de mogelijkheid om een beschrijvende tekst in te voeren.



6.2 Case Editor

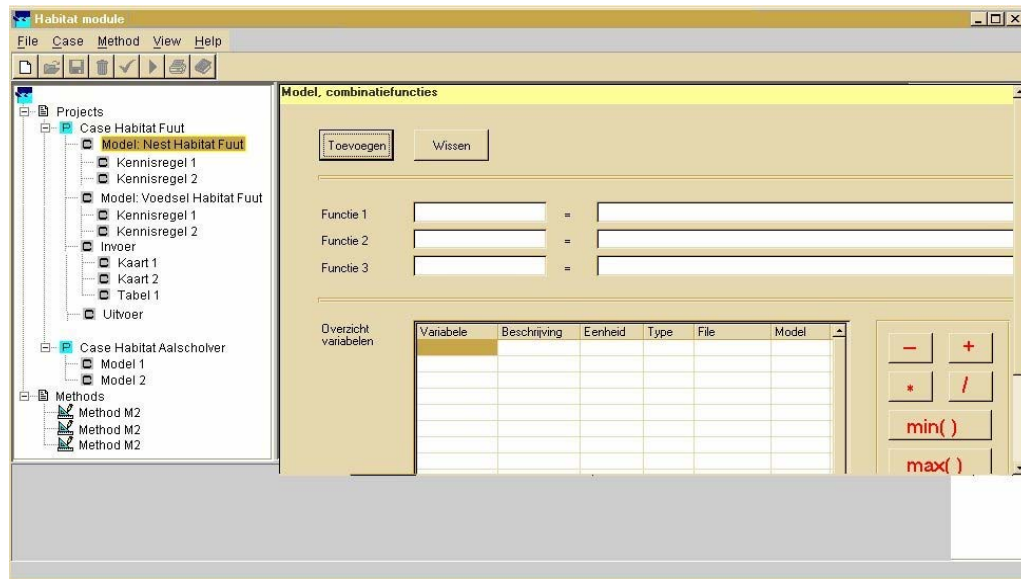
De case editor bevat een tabel waarin alle invoerparameters kunnen worden gespecificeerd. Daarnaast is er ruimte voor een beschrijvende tekst.

In fase 1 zal er ook een mogelijkheid moeten komen om een script aan te wijzen, welke door PCRaster zal worden geëxecuteerd. In dat geval zal het script dus niet afgeleid worden uit de onderliggende modellen, paramters, etc.



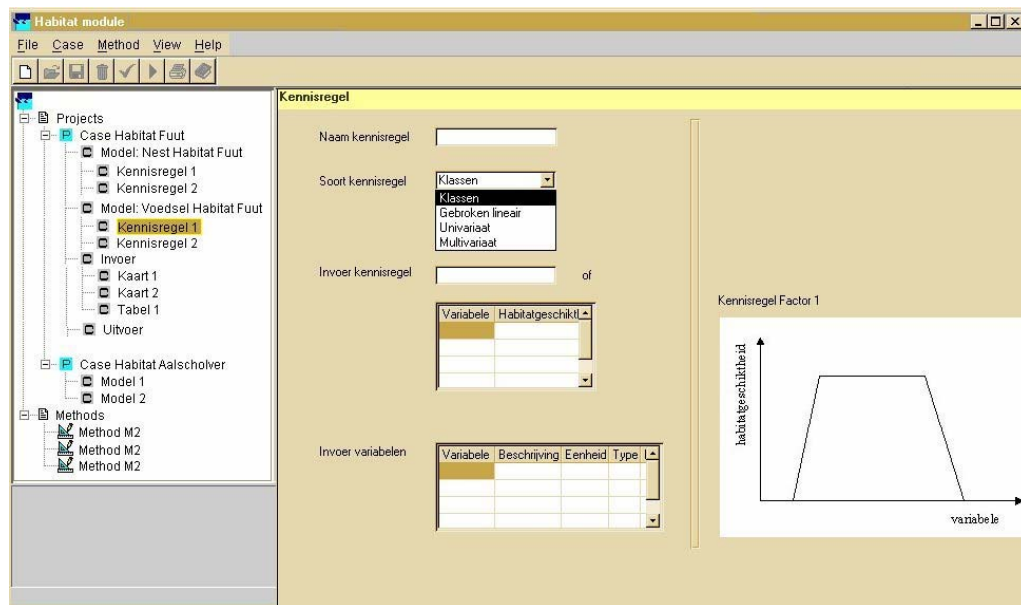
6.3 Model Editor

In de model editor kunnen een aantal kennisregels worden ge-edit. De typen classicifering, gebroken lineair, univariant en multivariant kunnen worden ge-edit.



6.4 Rule Editor

In de rule- of kennisregeleeditor kunnen kennisregels worden ge-edit. Dit kan een van de vormen classicifering, gebroken lineair, univariant en multivariant zijn. In het geval dat er een formule wordt ingegeven, dan wordt deze geparst op zijn geldigheid en wordt deze ter illustratie geplot (met voorbeeld-variabelen).

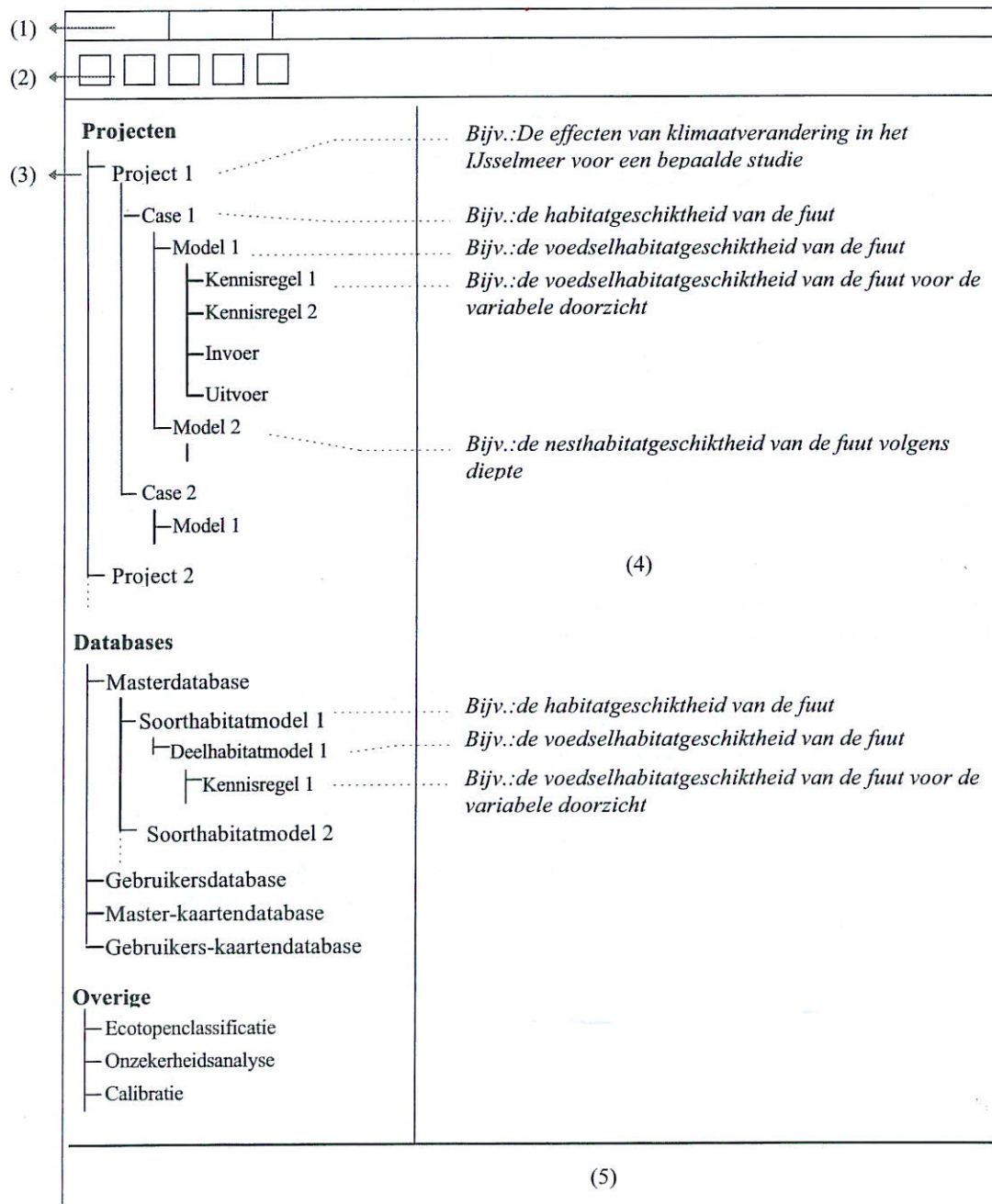


Bijlage

Schermontwerp en werkproces

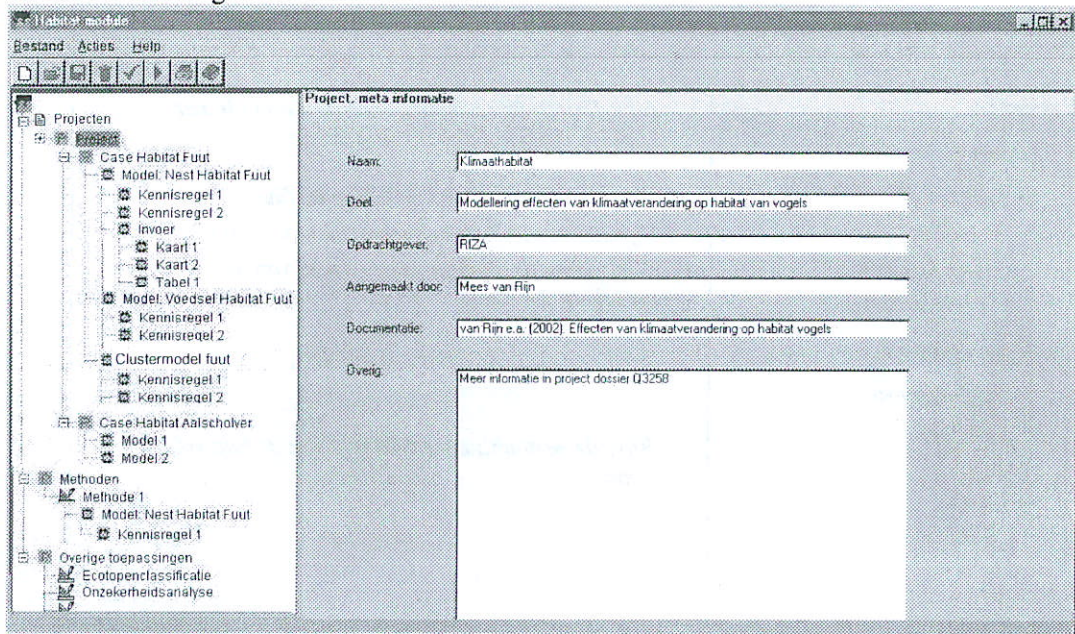
Structuur van de user-interface

Het begin scherm van de Habitat Module is opgebouwd uit 5 onderdelen, te weten: 1) pull-down menu's, 2) taakbalk, 3) projecten management, 4) werkgebied en 5) berichtengebied.



Werkproces van de gebruiker

1. Project: Bij de aanvang van een nieuwe studie maakt de gebruiker allereerst een project aan. Men staat op het item 'Projecten' in de projectmanagement structuur en kiest via *Bestand, Nieuw* voor *Project*. In het werkgebied kan meta informatie van het project worden ingevuld. Voor ieder nieuwe scenario, alternatief of ander studiegebied moet een nieuw project worden aangemaakt. Het is ook mogelijk bestaande projecten van elders te kopiëren, importeren of naar elders te exporteren. Door bestaande projecten te kopiëren kunnen gemakkelijk nieuwe alternatieven of scenario's worden gemaakt.



Figuur 1 User-interface bij het aanmaken van een project

2. Case: Daarna wordt een case aangemaakt. Men staat op het item 'Project' in de projectmanagement structuur en kiest via *Bestand, Nieuw* voor *Case*. Vervolgens kan in het menu worden gekozen voor een habitatanalyse, ecotopenclassificatie of voor overige toepassingen. Onder overige toepassingen wordt verstaan calibratie, validatie of onzekerheidsanalyse. Dit wordt in een latere versie van het modelinstrument uitgewerkt. Het werkgebied toont een overzicht van de variabelen, die binnen de case worden gebruikt. Afhankelijk van de aangevinkte opties (in, out, in-out en all) worden alle invoer, uitvoer, doorgeef en alle variabelen gepresenteerd. Op het tabblad 'Meta informatie' kan weer de meta informatie over de case worden opgegeven volgens hetzelfde format als bij een project.

[illegible]

Figuur 2 Het werkgebied van de user-interface bij het aanmaken van een case. Het tabblad met 'Meta-informatie' heeft hetzelfde format als de meta-informatie bij een project

3. Habitatmodellering: hiermee wordt de habitatgeschiktheid voor een soort berekend.

3a. Case: In het geval van een habitatgeschiktheidsanalyse kunnen modellen worden toegevoegd uit de masterdatabase (zie ook figuur 1). In de masterdatabase staan de habitatmodellen inclusief de bijbehorende kennisregels. Voor het modelleren van de habitatgeschiktheid van de fuut kan bijvoorbeeld het soorthabitatmodel van de fuut worden toegevoegd uit de masterdatabase. Er kunnen natuurlijk ook geheel nieuwe modellen worden aangemaakt. De nieuwe modellen kunnen worden opgeslagen in een eigen gebruikers database, zodat ze bij nieuwe projecten weer gemakkelijk te gebruiken zijn. De habitat modellen in de masterdatabase zijn toegevoegd door de beheerder van het habitatinstrument. Het is niet de bedoeling dat de gebruiker de gegevens in de masterdatabase kan wijzigen. Het afschermen van de database is niet voorzien in fase 1. Op caseniveau wordt ook een overzicht gegeven van de in- en uitvoer (zie figuur 2). De informatie van de in- en uitvoer wordt automatisch bij de modellen met kennisregels vandaan gehaald, maar kan hier ook worden aangepast. Dit verandert dan automatisch in de modellen met de kennisregels.

3b. Model: Een case bestaat uit een aantal modellen, die gekopieerd worden vanuit de database of nieuw worden aangemaakt. Gekopieerde modellen kunnen binnen een case worden aangepast. Op modelniveau kunnen combinatiefuncties worden toegevoegd of aangepast (zie figuur 3). Een voorbeeld van een combinatiefunctie: nesthabitatkwaliteit = min (factorindices van diepte, doorzicht, fosfaat). Op modelniveau wordt ook een overzicht gegeven van de in- en uitvoer. De informatie van de in- en uitvoer wordt automatisch bij de kennisregels vandaan gehaald.

Model, combinatiefuncties

Combinatiefuncties

HGltot-min = Min(HGI-df, HGltotP)

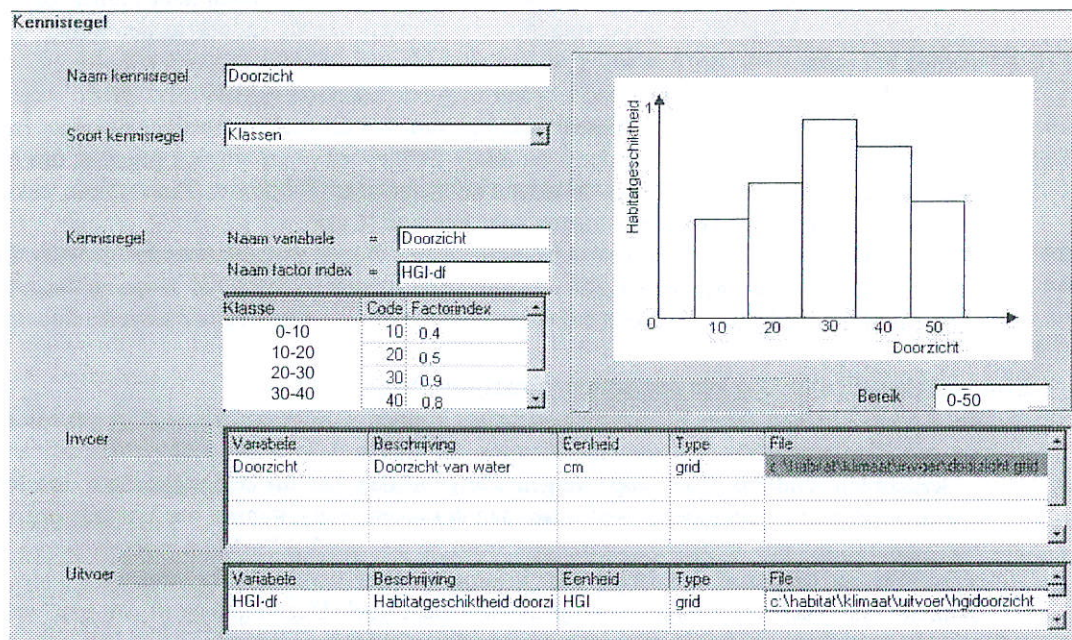
HGltot-gem = Gem(HGI-df, HGltotP)

Overzicht van variabelen

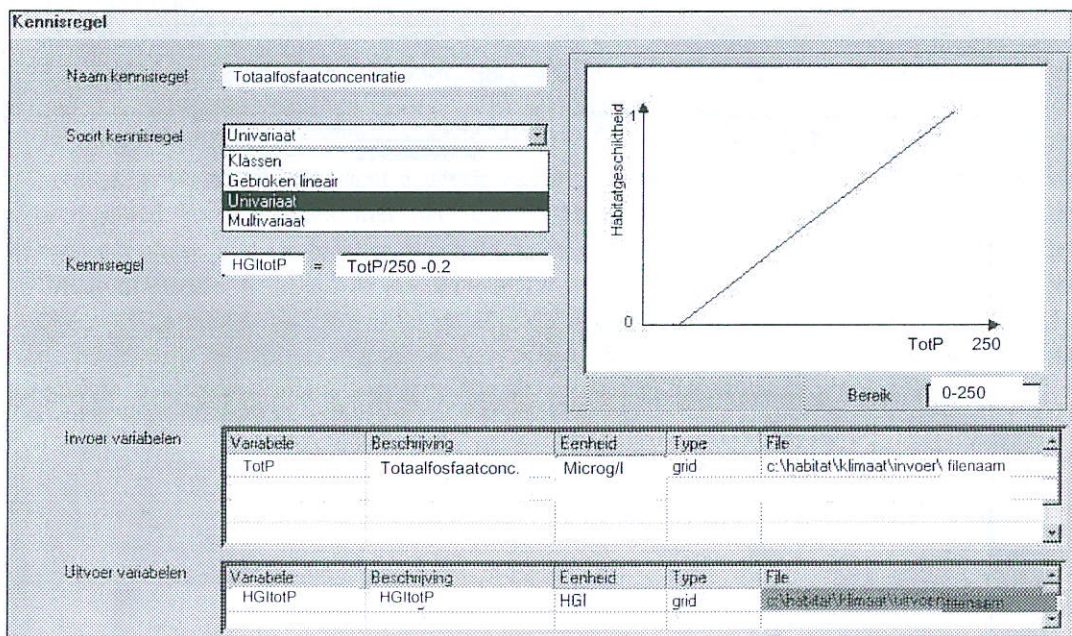
Indices	Beschrijving	Eenheid	Type	File
HGI-df	HGI-df	HGI	kaart	c:\hgidoorzicht.map
HGltotp	HGltotp	HGI	kaart	c:\hgltotp.map
HGltot-min	HGltotaal-minimum	HGI	kaart	c:\hgltotmin.map
HGltot-gem	HGltotaal-gemiddelde	HGI	kaart	c:\hgltotgem.map

Figuur 3 User-interface bij het bewerken van een model

3c Kennisregels: Een model bestaat uit een aantal kennisregels. De kennisregels kunnen in het werkgebied worden aangepast of aangemaakt. Bij 'naam kennisregel' geeft de gebruiker een korte beschrijving van de kennisregel (bijvoorbeeld: doorzicht in zomer of geschiktheid volgens doorzicht zomer). Deze naam verschijnt links in het scherm voor 'projecten management' (zie figuur 1). Vervolgens wordt dan het soort kennisregel gekozen: klassen, gebroken lineair, univariaat of multivariaat. Afhankelijk van de functie kan vervolgens de functie of een tabel worden ingevuld. De formule parser van het instrument herkent zelf de namen van de variabelen en operatoren. Deze verschijnen in de overzichtstabel op het scherm. Deze tabel geeft informatie over: invoerfile, beschrijving, type en eenheid. In de tabel moet de gebruiker de locaties opgeven van de bestanden met kaartinformatie over de variabelen. Het is de bedoeling dat in de help files van het modelinstrument een overzicht komt te staan van algemeen gebruikte variabelen. Hier kan dan worden gezien hoe variabelen standaard worden genoemd en in welke eenheid deze worden gepresenteerd, zodat er niet meerdere namen en bestanden van dezelfde variabele worden gemaakt. Wanneer de geschiktheid middels klassen wordt aangegeven moeten de naam van de variabele (de habitatfactor die de geschiktheid bepaalt, bijvoorbeeld waterdiepte), de naam van de factorindex (geschiktheid volgens diepte) en de klassentabel worden ingevuld (zie figuur 4). In de klassentabel moet worden aangegeven wat de factorindex is binnen een bepaalde klasse van de variabele. De kennisregels worden uiteindelijk gepresenteerd in een grafiek. Onder de grafiek moet de gebruiker de variabele voor de x-as opgeven en het bereik op de x-as. In fase 1 wordt geen grafiek getoond bij multivariate kennisregels. NB in de schermen voor het bewerken van een model en een kennisregel mist nog een overzichtje van de operatoren (+, -, * e.a.) die de gebruiker mag hanteren.



Figuur 4 Het werkgebied van de user-interface bij het bewerken van een kennisregel. In dit geval gaat het om een klassenverdeling



Figuur 5 Het werkgebied van de user-interface bij het bewerken van een kennisregel. In dit geval gaat het om een univariate of multivariate functie

3d. Bekijken invoer: door in het case management venster te browsen naar invoer, op een kaart of tabel te staan en deze vervolgens te activeren, kan de invoer in het werkgebied worden bekeken.

3e. Berekenen: Daarna kan de case worden uitgerekend door via het menu item 'Acties' naar rekenen te gaan. Daar kan worden gekozen voor het doorrekenen van een project (alle cases binnen een project worden dan berekend). Er kan ook voor worden gekozen om een reeks van cases uit te rekenen door te kiezen voor het item 'Batch'. Er kan ook een enkele case worden doorgerekend door te kiezen voor het item 'Case'.

3f. Resultaten: de uitvoer (kaarten en tabellen) kan bekeken worden onder de directory uitvoer. Door in het case management gebied op een kaart of tabel te staan, kan het resultaat in het werkgebied worden bekeken.

4 Ecotopenclassificatie: hier kan een ecotopenkaart worden gemaakt.

4a. Aanmaken/kopiëren: Ecotopenclassificaties kunnen worden aangemaakt of gekopieerd uit een database voor ecotopenclassificaties. In figuur 1 zijn de ecotopenclassificaties ondergebracht onder overige toepassingen. De gebruiker kan dit zelf veranderen door een aparte database met ecotopenclassificaties aan te maken (een database heeft het niveau van een project) Bij een ecotopenclassificatie staat in het werkgebied een overzicht van de abiotische en biotische variabelen met informatie over invoerfile, beschrijving, type en eenheid. Daarnaast staat er een ecotopenclassificatietabel. Deze tabel geeft aan met welke combinaties van waarden van variabelen ecotopen gedefinieerd zijn. Hier kan ook een bestaande tabel worden binnengehaald. Een bestaande tabel dient aan een afgesproken format te voldoen. Alle informatie kan in het werkgebied worden aangepast.

Overige toepassingen, ecotopenclassificatie

Variabele	Beschrijving	Eenheid	Type	File
Vegetatiestruct	vegetatiestructuur bij huidig beheer		kaart	c:\habitat\klimaat\ecotoop\vegstructuur
Overstromingsd	overstromingsduur		kaart	c:\habitat\klimaat\ecotoop\hydrodynamiek

Classificatietabel

Ecotoop	Beschrijving	vegetatiestructuur	overstromingsduur	variabele 3	variabele 4
1	Hoogwatervrije ruigte	4	6		
2	Grazige gorzen	4	3		
3	Ondiep open water	1	1		

Figuur 6 Het werkgebied van de user-interface tijdens een ecotopenclassificatie

4b. Invoer bekijken: de invoer kan weer bekeken worden door in het case management gebied op een kaart te staan en deze te activeren. Deze wordt dan rechts gepresenteerd.

4c. Berekenen: Daarna kan de classificatie worden uitgevoerd door via het menu item 'Acties' naar rekenen te gaan.

4d. Resultaten: resultaten kunnen bekeken worden onder de directory uitvoer. Door in het case management gebied op een kaart te staan, kan het resultaat in het werkgebied worden bekeken. De uiteindelijke kaart kan als invoer dienen voor een habitatmodellering.

5. Overige toepassingen; hier kan een calibratie, validatie of onzekerheidsanalyse van een bepaalde case of model worden uitgevoerd. Dit wordt in een latere versie van het modelinstrument uitgewerkt.

Werkproces van de beheerder

Masterdatabases van kaarten, modellen en ecotopenclassificaties moeten in principe alleen te veranderen zijn door een beperkt aantal aangewezen personen. Dit is te realiseren door een inlogfunctie in te bouwen of door een aparte beheersversie te maken. Dit is niet voorzien in fase 1, zodat in deze fase de masterdatabases in principe nog door elke gebruiker te wijzigen zijn. Gebruikers kunnen het beste een aparte eigen gebruikersdatabase aanmaken.

1. Masterdatabase habitatmodellen. In de Masterdatabase zijn verschillende soorthabitatmodellen opgeslagen. Deze soortmodellen kunnen bestaan uit deelhabitatmodellen. Een voorbeeld van een soorthabitatmodel is de habitatgeschiktheid van de fuut. Deelhabitatmodellen kunnen in dat geval ontwikkeld zijn voor het nesthabitat van de fuut of de voedselhabitat van de fuut. Het combineren van resultaten uit deelhabitatmodellen in een ruimtelijk model (clustermodel) is niet voorzien in fase 1.

1a. Soorthabitatmodel. Op dit niveau in de masterdatabase kan de beheerder meta-informatie invullen en wordt een overzicht gegeven van de in- en uitvoer gegevens. Het scherm van het werkgebied komt overeen met het 'Case' scherm bij de projecten, behalve dat in de masterdatabase niet de locaties ingevuld kunnen worden van kaartbestanden. Het tabblad 'Meta-informatie' wijkt ook af van die van 'Project' en 'Case' (figuur 1) en is weergegeven in figuur 7. Hier kan ook informatie worden gegeven over het type watersysteem waarin het soorthabitatmodel (bijvoorbeeld: rivier, meer, delta) geldig is en voor welke regio (bijvoorbeeld: Nederland, IJsselmeer).

Soorhabitatmodel, meta informatie	
Naam:	
Doel:	
Type watersysteem:	
Regio:	
Aangemaakt door:	
Documentatie:	
Overig:	

Figuur 7 Meta informatie van een soorhabitatmodel in de Masterdatabase

1c. Kennisregels. Het onderdeel kennisregels komt overeen met de kennisregels bij de projecten (figuur 4 en 5).

- 2. Master-kaartendatabase.** In de Master-kaartendatabase zijn de standaardkaarten opgeslagen. Het werkgebied bestaat uit twee tabbladen. Op het tabblad 'Meta-informatie' kan de meta-informatie over de kaart worden ingevuld door de beheerder. De gebruiker kan deze informatie alleen bekijken. Op het blad 'presentatie' is een plaatje van de kaart te zien (zie figuur 9). In figuur 9 zijn nog niet de functieknoppen voor presentatiemogelijkheden opgenomen (inzoomen, toevoegen tekst etc.).

Kaartendatabase

Meta informatie | Presentatie

Naam:

Beschrijving:

Eenheid:

Eigenaar bestand:

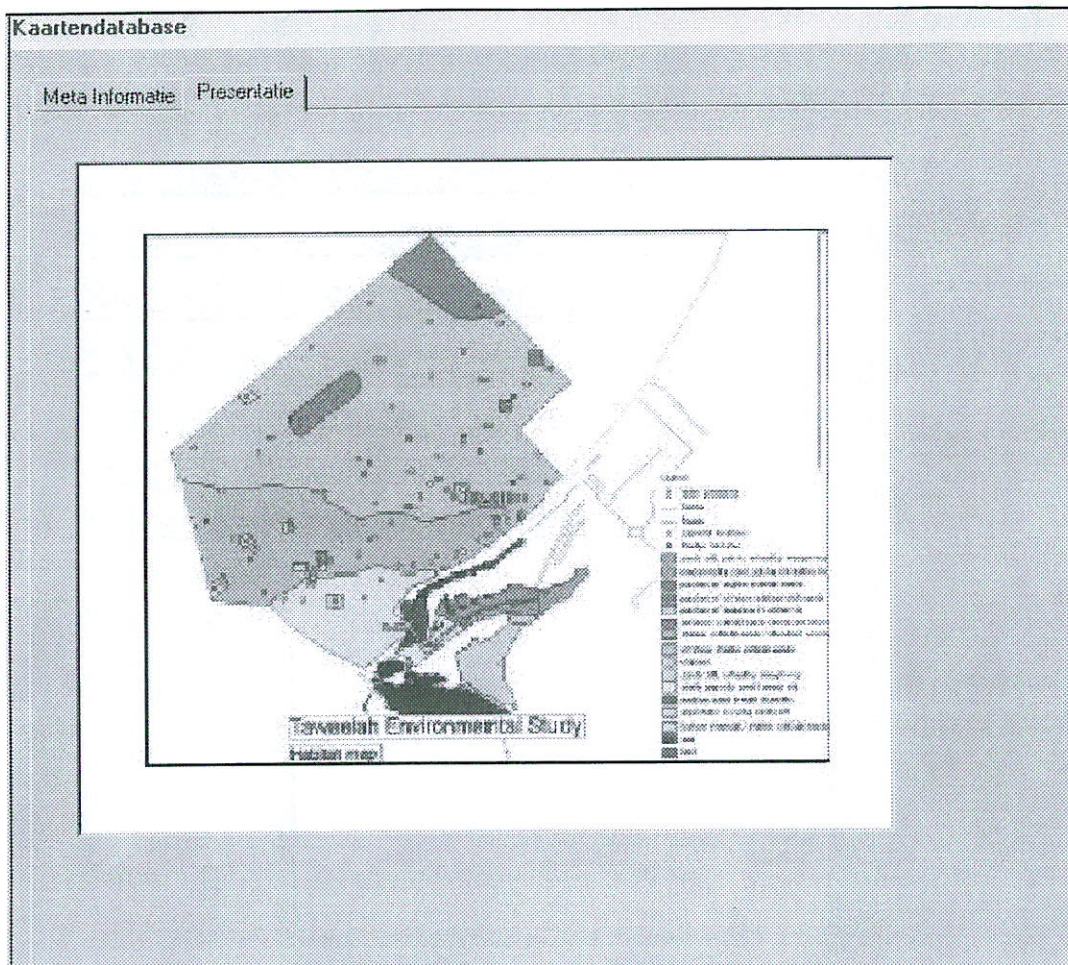
Vrij te gebruiken?:

Aangemaakt door:

Documentatie:

Overig:

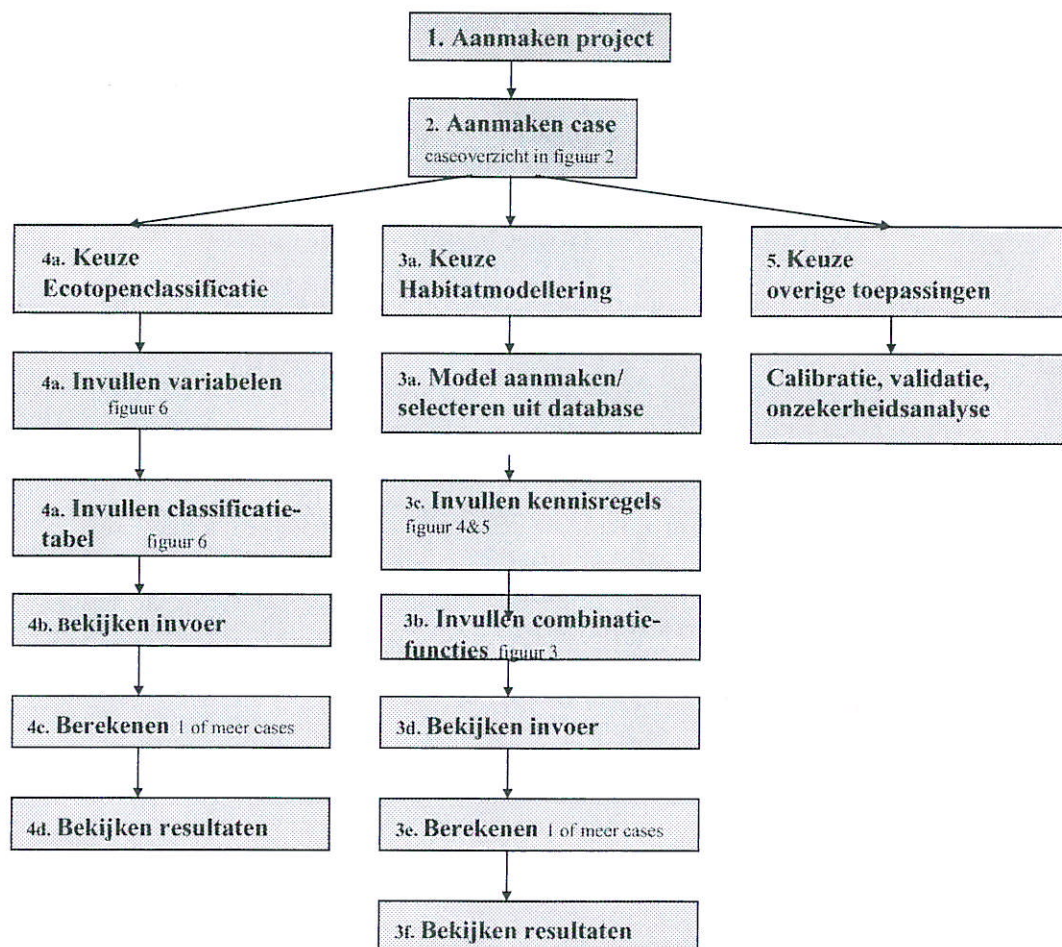
Figuur 8 Werkgebied van het Habitatinstrument wanneer de kaartendatabase actief is. In dit geval het tabblad 'Meta informatie'



Figuur 9 Werkgebied van het Habitatinstrument wanneer de kaartendatabase actief is. In dit geval het tabblad waar de kaart bekeken kan worden

Stroomschema

Het werkproces van de gebruiker is samengevat in het stroomschema in figuur 10. NB De keuze voor overige toepassingen wordt pas geïmplementeerd in een toekomstige fase van het modelproject, afhankelijk van de wensen van gebruikers.



* Nummers corresponderen met nummers in de tekst. Daar wordt meer informatie gegeven over de actie.

Figuur 10. Stroomschema van het werkproces van de gebruiker.



WL | Delft Hydraulics

Rotterdamseweg 185
postbus 177
2600 MH Delft
telefoon 015 285 85 85
telefax 015 285 85 82
e-mail info@wldelft.nl
internet www.wldelft.nl

Rotterdamseweg 185
p.o. box 177
2600 MH Delft
The Netherlands
telephone +31 15 285 85 85
telefax +31 15 285 85 82
e-mail info@wldelft.nl
internet www.wldelft.nl

