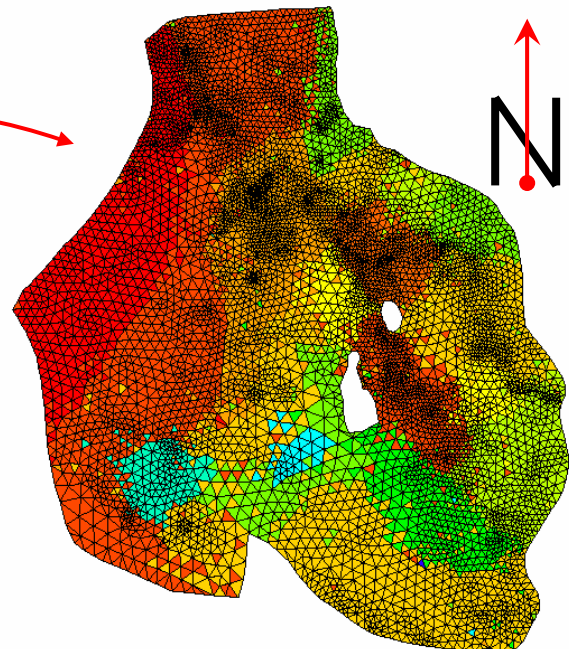
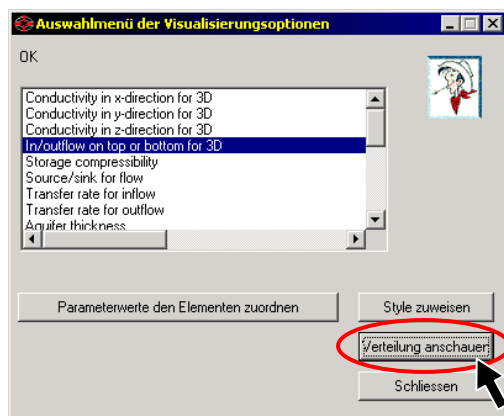


Thomas Graf

## MODINT Smallworld GIS gestützte Grundwassermodellierung der Staufener Bucht



Diplomarbeit unter der Leitung von Prof. Dr. Ch. Leibundgut  
Freiburg i. Br., im Dezember 2001

**Institut für Hydrologie** der Albert-Ludwigs-Universität Freiburg i. Br.

Thomas Graf

**MODINT Smallworld GIS gestützte  
Grundwassermodellierung der  
Staufener Bucht**

Referent: Prof. Dr. Ch. Leibundgut

Koreferent: Prof. Dr. S. Demuth

Diplomarbeit unter der Leitung von Prof. Dr. Ch. Leibundgut  
Freiburg i. Br., im Dezember 2001



## Danksagung

An erster Stelle gebührt mein herzlichster Dank meiner Frau Mardi! Sie hat mich während der Diplomarbeit ständig ermutigt und seelisch massiert. Bei jedem Rückschlag konnte ich mich auf ihre Hilfe verlassen und neue Kraft schöpfen. Das war wirklich ganz prima! Mardi hat außerdem die englische Zusammenfassung für mich korrigiert.

Meinen Eltern Helga und Dr. Günter Graf danke ich für die tolle Unterstützung während meines gesamten Studiums. Sie haben mir diese Ausbildung ermöglicht und waren immer für mich da.

Herzlich danke ich Volker Abraham, der mir zu Beginn meiner Diplomarbeit bei der Einrichtung meines Rechner-Arbeitsplatzes zur Seite stand. Durch seine schnelle und unkomplizierte Art der Hilfe hielt sich der Zeitverlust in Grenzen.

Meinem Betreuer Stephen Schrempp sowie Wolfgang Klink und Jan Kiefer danke ich dafür, dass sie mir dabei geholfen haben meine anfangs rudimentären Programmierkenntnisse in MAGIK zu intensivieren.

Mein besonderer Dank gebührt den Damen und Herren, die die Geduld hatten und die sich für mich Zeit nahmen, die Zusammenfassung in ihre Muttersprache zu übersetzen. Dies sind namentlich Patrick Verté (französisch), Šárka Najbrtová (tschechisch), Zuzaná Zemberova (slowakisch), Zhang Yang (chinesisch) und Shizu Kaga (japanisch).

Schließlich bedanke ich mich bei Wolfi, Sven, Tilo, Jannis und Peter, ohne die das Studium und Leben in Freiburg nicht halb so schön gewesen wäre.

Freiburg, den 20.12.2001      Thomas Graf

# Inhaltsverzeichnis

Danksagung.....	IV
Inhaltsverzeichnis .....	V
Verzeichnis der Abbildungen im Text.....	VII
Verzeichnis der Tabellen im Text .....	IX
Verzeichnis der Abbildungen im Anhang .....	X
Verzeichnis der Tabellen im Anhang .....	XI
Liste der verwendeten Symbole und Abkürzungen .....	XII
Zusammenfassung .....	XIV
Abstract .....	XVI
Sommaire / Zusammenfassung auf französisch.....	XVIII
Shrnutí / Zusammenfassung auf tschechisch .....	XX
Zhrnutie / Zusammenfassung auf slowakisch.....	XXII
摘要 / Zusammenfassung auf chinesisch .....	XXIV
要旨 / Zusammenfassung auf japanisch .....	XXV
1 Einleitung .....	1
2 Problemstellung und Zielsetzung .....	3
3 Anwendungsgebiet und bisherige Arbeiten .....	5
3.1 Anwendungsgebiet.....	5
3.2 Bisherige Arbeiten .....	6

---

3.2.1	Erstellung eines Modells für das Untersuchungsgebiet .....	6
3.2.2	Das bodenkundlich-meteorologische Modul BOMET .....	7
3.2.3	Die BOMET-Erweiterung BOMET_BS .....	7
3.2.4	Das Oberflächen- und Grundwassermodul OGRUT .....	7
3.2.5	Das Hydrogeologie Modul HGEO .....	8
<b>4</b>	<b>SMALLWORLD GIS .....</b>	<b>9</b>
4.1	Das GIS und dessen Architektur .....	9
4.2	Der EMACS: Entwicklungsumgebung in MAGIK .....	11
4.3	MAGIK .....	12
4.3.1	MAGIK als Sprache .....	12
4.3.2	Threads .....	13
4.3.3	Vererbung .....	13
4.3.4	Definition von Exemplaren und Methoden .....	14
4.3.5	Editoren .....	14
4.3.6	Speichereinheiten von SMALLWORLD GIS .....	15
4.4	Das CASE - Tool .....	16
4.5	Die Architektur der Geo - Datenbank .....	18
4.6	Module und Fachschalen .....	20
4.7	Fazit .....	21
<b>5</b>	<b>Der Simulator FEFLOW .....</b>	<b>23</b>
5.1	Einleitung .....	23
5.2	Relevante Dateitypen in FEFLOW .....	24
5.3	In FEFLOW verwendete Parameter .....	25
5.4	Fazit .....	25
<b>6</b>	<b>Das Modul MODINT zur Modellintegration .....</b>	<b>26</b>
6.1	Vorgehensweise und Umsetzung .....	26
6.2	Umbau und Integration von OGRUT .....	27
6.3	Die "neue" Oberfläche von MODINT .....	28

6.4	Geschriebene Parser .....	30
6.4.1	Der *.fem - Parser .....	30
6.4.2	Der *.dac - Parser .....	30
6.5	Die Output - Datei *.eco .....	34
6.6	Auswahl zur Visualisierung von Modellparametern .....	36
6.7	Datenmodell von MODINT .....	36
6.8	Fazit.....	38
<b>7</b>	<b>Ergebnisse .....</b>	<b>39</b>
7.1	MODINT als GIS - Werkzeug.....	39
7.2	Ergebnisplots.....	40
<b>8</b>	<b>Schlussbetrachtung und Ausblick.....</b>	<b>45</b>
<b>9</b>	<b>Literaturverzeichnis .....</b>	<b>47</b>
	<b>Anhang.....</b>	<b>50</b>
	<b>Index .....</b>	<b>59</b>
	<b>Ehrenwörtliche Erklärung .....</b>	<b>61</b>

## Verzeichnis der Abbildungen im Text

Abbildung 1.1: Trinkwasserentnahme in Deutschland im internationalen Vergleich (nach Water Services Association, 1996).....	1
Abbildung 2.1: Überblick über das Projekt WAQIS (ergänzt nach WAQIS, 2000).....	3
Abbildung 3.1: Lage des Untersuchungsgebietes .....	5
Abbildung 3.2: Das Untersuchungsgebiet südlich von Freiburg.....	6
Abbildung 4.1: Überblick über die vereinfachte SMALLWORLD GIS-Architektur.....	10
Abbildung 4.2: Die Entwicklungsumgebung EMACS .....	11
Abbildung 4.3: Verwendung von Speicherplatz in SMALLWORLD GIS (verändert nach GE Smallworld, 1999a) .....	15
Abbildung 4.4: Verwendung von Speicher beim Erzeugen und Speichern neuer Objekte (verändert nach GE Smallworld, 1999a).....	16
Abbildung 4.5: Alternativenhierarchie unter SMALLWORLD GIS (verändert nach GE Smallworld, 1999b) .....	19
Abbildung 5.1: Finite Elemente Netz (FEFLOW) und Finite Differenzen Netz (MODFLOW) .....	23
Abbildung 6.1: Wirkungsweise der Schnittstelle an den Simulator FEFLOW .....	26
Abbildung 6.2: Struktur der gekapselten Module in der Entwicklerdatenbank .....	27
Abbildung 6.3: Der zentrale Editor von MODINT .....	28
Abbildung 6.4: Modifikation einer in FEFLOW generierten *.fem-Datei durch MODINT ...	29
Abbildung 6.5: Einlesen eines FE-Netzes von OGRUT .....	31
Abbildung 6.6: Ausgabeformen von MODINT beim Einlesen einer *.dac-Datei .....	31
Abbildung 6.7: Funktionsweise zur Erzeugung flächenhafter Parameterverteilungen in MODINT .....	33
Abbildung 6.8: Menü zur Auswahl der Einleseoptionen .....	34
Abbildung 6.9: Struktur der *.eco-Datei .....	35
Abbildung 6.10: Menü zur Auswahl des darzustellenden Parameters .....	36
Abbildung 6.11: Die Editoren des Datenmodells von MODINT .....	37
Abbildung 7.1: Erzeugung einer räumlichen Verteilung der FEFLOW-Parameterwerte.....	41
Abbildung 7.2: Zusammenspiel der verschiedenen Module von SMALLWORLD GIS und dem Simulator FEFLOW .....	42
Abbildung 7.3: Das zugrunde liegende Finite Elemente Netz .....	43
Abbildung 7.4: Verteilung des Speicherkoeffizienten im Untersuchungsgebiet .....	44



## Verzeichnis der Tabellen im Text

Tabelle 4.1: Beispiele zum Entity-Relationship-Modell .....	17
Tabelle 6.1: Beschreibung der Slots von <code>parameter</code> .....	32

## **Verzeichnis der Abbildungen im Anhang**

Abbildung A.1: Verteilung der hydraulischen Leitfähigkeit im Untersuchungsgebiet.....	55
Abbildung A.2: Verteilung der Grundwasserneubildung im Untersuchungsgebiet.....	56
Abbildung A.3: Verteilung der Porosität im Untersuchungsgebiet.....	57
Abbildung A.4: Verteilung der Aquifermächtigkeit im Untersuchungsgebiet.....	58

## **Verzeichnis der Tabellen im Anhang**

Tabelle A.1: Für das pragma-statement verwendete Levels.....	50
Tabelle A.2: Materialparameter für 3D Lösungen (WASY, 1998).....	54
Tabelle A.3: Stofftransportparameter (WASY, 1998) .....	54

## Liste der verwendeten Symbole und Abkürzungen

<i>ACE</i>	Activation Configuration Environment
<i>API</i>	Application Programming Interface
<i>ASCII</i>	American Standard Code for Information Interchange
<i>BOMET</i>	Bodenkundlich-meteorologisches Teilinformationssystem
<i>BOMET_BS</i>	Bodenschätz-Modul von BOMET
<i>CASE</i>	Computer-Aided Software Engineering
<i>DBMS</i>	Datenbank-Managementsystem
<i>DGM</i>	Digitales Geländemodell
<i>DOS</i>	Disk Operation System
<i>DS</i>	Datastore
<i>EMACS</i>	1. Escape Meta Alt Control Shift 2. Editor MACroS
<i>FD</i>	Finite Differenzen
<i>FE</i>	Finite Elemente
<i>FEFLOW</i>	Finite Element Subsurface FLOW System
<i>FE-Netz</i>	Finite Elemente Netz
<i>FRAC3DVS</i>	3Dimensional Variably-Saturated Fractured Rock Aquifer Model
<i>FV</i>	Finite Volumina
<i>GC</i>	Garbage Collection
<i>GE</i>	General Electric
<i>GIS</i>	Geographisches Informationssystem
<i>GNU</i>	<u>G</u> NU is <u>n</u> ot <u>U</u> NIX
<i>GUI</i>	Graphical User Interface
<i>GWNB</i>	Grundwasserneubildung
<i>HGEO</i>	Hydrogeologisches Modul

<i>IHF</i>	Institut für Hydrologie, Albert-Ludwigs-Universität Freiburg
<i>IFM</i>	Interface Manager
<i>ISO</i>	International Standardization Organization
<i>LISP</i>	List Processing
<i>LIWIS</i>	Lahmeyer International Wasserinformationssystem
<i>MODFLOW</i>	MODular three-dimensional finite-difference groundwater FLOW model
<i>MODINT</i>	Modellintegration
<i>OGRUT</i>	Oberflächen- und Grundwasser Teilinformationssystem
<i>OOP</i>	Object-Oriented Programming
<i>RDBMS</i>	Relationales Datenbank-Managementsystem
<i>RIS</i>	raumbezogenes Informationssystem
<i>RWO</i>	Real World Object
<i>SDS</i>	Static Object Space
<i>SGML</i>	Standard Generalized Markup Language
<i>SOC</i>	Spatial Object Controller
<i>SOM</i>	Spatial Object Manager
<i>TIN</i>	Triangulated Irregular Network
<i>UWSP</i>	Ungleichförmiges Wasserspiegellinienprogramm
<i>VIS_TOOL</i>	Visualisierungstool
<i>VMDS</i>	Version Managed Data Store
<i>WAQIS</i>	Wasserqualitätsinformationssystem
<i>WASY</i>	Gesellschaft für wasserwirtschaftliche Planung und Systemforschung mbH
<i>WSA</i>	Water Services Association

## Zusammenfassung

Anthropogene Belastungen des Grundwassers erfordern nicht nur einen bewussten Umgang damit, sondern insbesondere im Hinblick auf Trinkwassergewinnung auch die Option die maximal entnehmbare Grundwassermenge so exakt wie möglich bestimmen zu können. Das Strömungsverhalten von Grundwasser kann mathematisch näherungsweise (numerisch) mit Modellen beschrieben werden. Modelle basieren auf den zur Analytik notwendigen Inputdaten, die im Gelände erhoben werden. Bei der Sicherung der Grundwassersourcen ergibt sich eine große Zahl unterschiedlicher Problemstellungen, wie die Sicherung von Boden und Aquifer, die Wasserqualität, den Einfluß von Oberflächengewässern oder Fragen nach der Grundwasserneubildung. Darüber hinaus entstehen weitere Daten durch die Zuweisung von Gebietsparametern und Randbedingungen an das Modell.

Damit der Anwender einfach auf die Daten in einem Untersuchungsgebiet zugreifen bzw. mit den Informationen umgehen und sie nützen kann, wird ein Geoinformationssystem (GIS) verwendet. Die als Folge der zahlreichen Fragestellungen großen Datenmengen müssen in der zugehörigen Geodatenbank gespeichert werden und sind auf diese Weise zu weiterer Anwendung und Verarbeitung verfügbar. Das Wasserqualitäts- Informationssystem WAQIS ist genau diese Geodatenbank. Das WAQIS-Projekt wird am Institut für Hydrologie der Albert-Ludwigs-Universität Freiburg (IHF) in Zusammenarbeit mit dem lokalen Trinkwasserversorger badenova-AG bearbeitet.

Die für den Raum Freiburg genutzten Trinkwassergewinnungsgebiete sind im Süden die Staufener Bucht und in Freiburgs Osten das Zartener Becken. Zur modellhaften Erfassung dieser Gebiete wird der Simulator FEFLOW (Finite Elemente Flow Simulation Program) verwendet. Im Rahmen des WAQIS-Projekts wurden in Diplomarbeiten schon eine Reihe von Modulen und Tools entwickelt (BOMET, BOMET\_BS, OGRUT, HGEO), die aber nicht zentral dem Anwender zur Verfügung stehen. Ziel dieser Arbeit ist es, eine solche Oberfläche zu schaffen und übersichtlich zu gestalten.

Zur integralen Lösung wurde außerdem die Möglichkeit geschaffen, die Parametereinstellungen von FEFLOW auch im GIS sichtbar zu machen. Mit der Implementierung dieses Visualisierungswerkzeuges kann nun der Nutzer nicht nur die vorgehaltenen Funktionalitäten gebrauchen, sondern sich auch in beliebiger Weise ein Bild von den Einstellungen des zugrunde liegenden Modells machen. Zur Realisierung wurde ein weiteres Modul MODINT (Modellintegration) als Softwareanwendung des WAQIS unter SMALLWORLD GIS entwickelt. Hierzu eignet sich SMALLWORLD wegen seiner leistungsstarken Datenbank-Architektur besonders gut.

MODINT wurde in MAGIK, der objektorientierten Programmiersprache von SMALLWORLD, geschrieben. Von zentraler Bedeutung dieses programmierten Moduls ist die Anbindung an FEFLOW. Zu deren Verwirklichung wurden Parser geschrieben, die

Outputdateien von FEFLOW lesen und analysieren. Dies ist zum einen die \*.fem-Datei, die beim Preprocessing des Grundwassermodells generiert wird und in der die gesamte zur Modellierung notwendige Information (Randbedingungen, Zeitschritte, flächenhafte Verteilung der Parameterwerte) steht. Zum anderen wird beim Processing von FEFLOW die \*.dac-Datei erzeugt, in der außer den Modelleinstellungen alle Ergebnisse zu jedem Zeitpunkt stehen.

Die flächenhafte Verteilung der Parameterwerte wird von MODINT verstanden und zur Visualisierung dem zugehörigen Finiten Elemente Netz zugeordnet. Dabei bedient sich MODINT des im Rahmen der Entwicklung eines Bodenschätzmoduls für BOMET geschaffenen Style-Editors, in dem es möglich ist, die numerischen Werte der Parameter mit selbstdefinierten Farben zu belegen. Schließlich sieht der Nutzer die flächenhafte Verteilung eines ausgesuchten Gebietskennwertes im GIS als farblichen Verlauf.

Mit der Vereinheitlichung einiger SMALLWORLD GIS Komponenten, die am IHF entwickelt wurden und mit der Integration des Simulators FEFLOW ergänzt diese Arbeit die Angebote des WAQIS. Dieser Diplomarbeit ist außerdem ein Handbuch beigelegt, das sowohl dem Anwender als auch dem gewieften Programmierer und nicht zuletzt dem neugierigen Laien einen Einblick in die Funktionsweise von MODINT geben soll.

### **Schlüsselbegriffe**

Geographisches Informationssystem  
SMALLWORLD GIS  
Grundwassermodellierung  
FEFLOW  
Objektorientiertes Programmieren  
MAGIK  
Modellintegration  
Parser  
MODINT  
WAQIS-Projekt

## Abstract

Anthropogenic stress of the groundwater requires not only the conscious use of it, but also the option of how to determine the amount of water that can be safely extracted in view of the use as drinking water. The groundwater flow pattern can be approximately (numerically) described mathematically with the use of models. Models are based upon input data, gained in the field and necessary for computing. With the maintenance of groundwater, many questions arise, such as how to preserve soil and aquifer, how to preserve the water quality, how to predict the impact of surface water, or the issue of the groundwater recharge. Furthermore, more data generates due to assigning field parameters and boundary conditions to the model.

In order to allow the user to have easy access to the data of the investigated area and to utilize the information, a Geographical Information System (GIS) is used. As a consequence of the many considered problems, a large amount of data is generated, that has to be saved in the respective geo-databases so that it is useable for further applications and inquiries. The Water Quality Information System WAQIS represents this geo-database. The WAQIS-Project is carried out by the Institute of Hydrology at the Albert-Ludwigs-Universität Freiburg (IHF) in cooperation with the local water supplier badenova-AG.

The areas where drinking water is gained for Freiburg are, in the South, the Staufen Basin and to Freiburg's East the Zarten Basin. To describe these areas with a model, the simulator FEFLOW (Finite Elemente Flow Simulation Program) is used. During the WAQIS-Project, numerous investigations were performed and several modules and tools were developed (BOMET, BOMET\_BS, OGRUT, HGEO) which are not yet accessible from a central surface. One purpose of this work is, to create a user-friendly surface where the existing programs can be executed.

Apart from that, the ability to make the parameter adjustments from FEFLOW visual in the GIS was established. The implementation of this visualization tool not only enables the user to apply the provided methods, but also to get an overview of the spatial distribution of any parameter values in the considered model. In order to carry out this second step, another module called MODINT (model integration) was developed as a software application of WAQIS under SMALLWORLD GIS. Due to its powerful database-structure, SMALLWORLD is especially well suited for this task.

MODINT was written in MAGIK, the object-oriented programming language of SMALLWORLD. In MODINT, the integration of FEFLOW is of key concern. In its creation, parsers were written, which are able to read and analyze output files of FEFLOW. On the one hand, there is the \*.fem-file which is generated during the preprocessing of the groundwater model and which contains the whole information that is required to compute the results (boundary conditions, time steps, spatial distribution of parameter values). On the other hand, the processing in FEFLOW makes a \*.dac-file which contains, apart from the model adjustments all the results to every time step.



The spatial distribution of the parameter values is understood by MODINT and assigned to the respective Finite Element Mesh. To do so, MODINT makes use of a style-editor which is written for the BOMET extension and which makes it possible to actually assign numerical values to self defined colors. Finally, the user can see the spatial distribution of a selected parameter in the GIS as color range.

With the standardization of some SMALLWORLD GIS components which are developed in the IHF and with the integration of FEFLOW, this work is a contribution to WAQIS. To the following thesis, a manual is enclosed with which the user, the clever programmer and last but not least the curious novice should get an idea about how MODINT works.

## **Keywords**

Geographical Information System  
SMALLWORLD GIS  
Groundwater Modelling  
FEFLOW  
Object Oriented Programming  
MAGIK  
Modellintegration  
Parser  
MODINT  
WAQIS-Project

## Sommaire / Zusammenfassung auf französisch

Face à la pression exercée par nos sociétés sur les nappes phréatiques, il est non seulement nécessaire de faire un usage consciencieux de celles-ci, mais également de pouvoir déterminer aussi exactement que possible la quantité d'eau qui peut être captée pour l'approvisionnement durable en eau potable. D'une manière générale, le comportement des flux aquifères peut être décrit au moyen de modèles numériques. Ces modèles sont basés sur les données nécessaires à l'analyse, récoltées préalablement sur le terrain. La conservation des ressources aquifères reste confrontée à un grand nombre de problématiques, comme la préservation du sol et de la nappe, la question de la qualité de l'eau, la prédiction de l'influence des eaux de surface ou encore des questions liées à l'approvisionnement des nappes phréatiques. De plus, d'autres données doivent être prises en considération suite à la définition de paramètres caractérisant la zone d'études et les limites du modèle.

De manière à permettre à l'utilisateur un accès facile aux données de la zone considérée ainsi que leur usage, un système d'information géographique (GIS) a été utilisé. En conséquence du grand nombre de paramètres à prendre en compte, d'importantes quantités de données sont générées, qui doivent être enregistrées dans une base de données géographique appropriée en vue du développement et de l'interrogation ultérieures de celles-ci. Le système d'information sur la qualité de l'eau WAQIS représente exactement ce type de base de donnée. Le projet WAQIS a été développé au sein de l'Institut d'hydrologie de l'université Albert-Ludwigs, Freiburg en collaboration avec l'agence locale de l'eau, badenova-AG.

Les zones de captage de la ville de Freiburg se situent au sud au niveau du bassin du Staufener Bucht et à l'est au Zartener Becken. La modélisation de cette zone a été réalisée au moyen du simulateur FEFLOW (Finite Element Flow Simulation Program). Dans le cadre du projet WAQIS, différents travaux de fin d'études ont déjà été réalisés par le passé qui ont permis le développement de toute une série de modules et d'outils (BOMET, BOMET\_BS, OGRUT, HGEO), mais ceux-ci ne sont actuellement pas à la disposition de l'utilisateur. L'objectif de ce travail est de développer une telle interface utilisateur où les programmes existants pourraient être utilisés de manière simple et conviviale.

En plus de cela, les modifications nécessaires ont été réalisées pour permettre l'ajustement des paramètres de FEFLOW visuellement à partir du GIS. Cette visualisation des paramètres dans le GIS permet non seulement à l'utilisateur d'utiliser les fonctions existantes de FEFLOW, mais également d'obtenir une image de la distribution spatiale des valeurs de n'importe lequel des paramètres au sein du modèle considéré. Cette dernière réalisation a nécessité le développement d'un module supplémentaire appelé MODINT (INTégration MODèle) en tant qu'application de WAQIS sous SMALLWORLD GIS. De part la puissance de l'architecture de sa base de données, SMALLWORLD est particulièrement bien adapté pour cette tâche.

MODINT a été écrit en MAGIK, le langage de programmation orienté objet de SMALLWORLD. L'intégration dans FEFLOW de ce module constitue un point essentiel de ce travail. Au cours de sa réalisation, des parsers ont été créés qui sont capables de lire et d'analyser les fichiers résultant du travail de FEFLOW. D'une part on trouve les fichiers en \*.fem qui sont générés durant le prétraitement du modèle de nappe phréatique et qui contiennent toute l'information nécessaire à la modélisation (conditions limites, pas de temps, distribution spatiale des valeurs des paramètres). D'autre part résultent du traitement par FEFLOW les fichiers en \*.dac, qui contiennent, à partir des ajustements du modèle, tous les résultats pour chaque pas de temps.

La distribution spatiale des valeurs des paramètres est prise en considération par MODINT et ordonnée en fonction des éléments finis correspondants en vue de leur visualisation. Pour accomplir cette tâche, MODINT se sert d'un Style-Editor qui a été conçu dans le cadre du développement d'un module pour la description des sols destiné à BOMET et qui permet d'attribuer une couleur au choix à chaque valeur numérique d'un paramètre. À terme, l'utilisateur peut visualiser la distribution spatiale du paramètre sélectionné en tant que étendue colorée.

Avec la standardisation de quelques composants de SMALLWORLD GIS qui sont développés au sein de l'IHF et leur intégration dans FEFLOW, ce travail constitue une contribution au projet WAQIS. À ce travail de fin d'études a été rajouté un manuel grâce auquel l'utilisateur, tant le programmeur averti que le profane curieux, devrait recevoir une introduction au mode de fonctionnement de MODINT.

## Mots clés

Système d'Information Géographique (GIS)

SMALLWORLD GIS

Modélisation de la nappe phréatique

FEFLOW

Programmation orientée objet

MAGIK

Intégration modèle

Parser

MODINT

Projet WAQIS

## Shrnutí / Zusammenfassung auf tschechisch

Antropologická zátěž spodní vody vyžaduje nejenom její uvědomělé zacházení, ale také možnost jak jen nej přesněji určit opci maximálního množství odběrné spodní vody, zejména s ohledem na získávání pitné vody. Chování proudu spodní vody může být popsáno modely přibližně matematicky (číselně). Modely spočívají v k analytice důležitých inputačních datech, které jsou vyvyšovány v prostoru. Při jištění zdrojů spodní vody se podrobí velké množství problematických stanovisek, jako jištění půdy a aquiferu, kvalita vody, vliv povrchní vody nebo otázky obnovování spodní vody. Přesto vznikají další údaje přidělením oblastních parametrů a okrajových podmínek na tento model.

Aby se mohl uživatel jednoduše chopit údajů v oblasti zkoumání respektive moci zacházet a užívat informace, je používán Geografický informační systém (GIS). Ty jako následek četně polkládaných otázek velkého množství údajů, musí být ukládány v příslušných geodatabankách a jsou tímto způsobem k dispozici k dalšímu užívání a zpracování. Informační systém kvality vody WAQIS je přesně tato geodatabanka. WAQIS-projekt je zpracováván v Institutu hydrologie Univerzity Alberta Ludwiga ve Freiburgu (IHF) ve spolupráci s lokálním zásobitelem pitné vody. Zásobování badenova a.s.

Oblasti získávání pitné vody pro oblast Freiburgu jsou na jihu Stafenská zátoka a na východě Zartenská pánev. Ke konskripci modelu těchto oblastí je používán simulátor FEFLOW (Finite Element Flow Simulation Program). V rámci WAQIS projektu byla v diplomových pracích vyvinuta již řada modulů a náčiní. (BOMET, BOMET\_BS, OGRUT, HGEO), které ale nejsou uživateli centrálně k dispozici. Cíl této práce je, jeden takový povrch vytvořit a přehledně uzpůsobit.

K integrálnímu řešení byla mimochodem poskytnuta možnost, zviditelnit parametrické stanovisko FEFLOW také v GIS. S implementováním tohoto vizuálního nástroje může uživatel používat nejenom vytýkající funkčnost, ale udělat si také jak je libo obrázek o postoji základně položeného modelu. K realizaci byl vyvynut další modul MODINT (Modelintegrace) jako použití softwaru WAQIS pod SMALLWORLD GIS. K tomuto se hodí SMALLWORLD kvůli jeho silnému výkonu datenbankové –architektury obzvlášť dobře.

MODINT byl napsán v MAGIKu, objekt-orientovaná programová řeč od SMALLWORLD. Centrální význam tohoto programového modulu je navázání na FEFLOW. K jeho uskutečnění byly napsány parsy, které čtou a analyzují Outputdata z FEFLOW. Toto je k jednomu \*.fem-Datei (databáze), která je při preprocessingu modelu spodní vody generována a stojí ve společně k modelování důležité informace (okrajové podmínky, časování, prostorové rozdělení hodnoty parametru). K jiným je při procesingu od FEFLOW vyrobena \*.dac-Datei, ve které stojí mimo modelového postoje všechny výsledky v jednom okamžiku.

Prostorové rozdělení hodnoty parametru je rozuměno MONDINTem a přiděleno k vizualování příslušné Finit elementární sítě. Přitom si poslouží MODINT v rámci vývoje modelu nerostného bohatství pro BOMET vytvořeným Styl-Editorem, ve kterém je možné doložit číselné hodnoty parametrů se samostatně definovanými barvami. Na závěr uvidí uživatel prostorové rozdělení vyhledaného parametru v GIS jako barevný průběh.

Sjednocením komponentu SMALLWORLD GIS, které byly vytvořeny na IHF a integrací simulátoru FEFLOW doceluje tato práce nabídky WAQIS. Mimochodem je k této diplomové práci přiložena příručka, která má dát jak uživateli, zasvědcenému programátorovi a posléze i zvědavému lajkovi nahlédnutí do způsobu funkce MODINTu.

## **Klíčové pojmy**

Geografický informační systém  
SMALLWORLD GIS  
Modelování spodní vody  
FEFLOW  
Objekt-orientované programování  
MAGIK  
Modelová integrace  
Parser  
MODINT  
WAQIS-projekt

Übersetzerin: Šárka Najbrtová

## Zhrnutie / Zusammenfassung auf slowakisch

Antropogénne zaťaženie podzemnej vody si vyžaduje nielen uvedomeľé zaobchádzanie, ale najmä z pohľadu získavania pitnej vody aj možnosť, čo najpresnejšieho určenia množstva podzemnej vody, ktoré môže byť bez rizika čerpané. Prúdenie podzemných vôd sa dá matematicky približne (numericky) popísať pomocou modelov. Modely sa opierajú o v teréne získané vstupné dáta, ktoré sú potrebné pre analytiku. Pri zabezpečovaní zdrojov podzemných vôd je potrebné riešiť množstvo rozličných problémov ako zachovanie pôdy a vodného režimu, zachovanie kvality vody, predpokladanie vplyvu povrchových vôd alebo otázku obnovy podzemných vôd. Okrem toho ďalšie údaje vznikajú priradením parametrov terénu a hraničných podmienok k modelu.

Nato, aby bol používateľovi umožnený jednoduchý prístup k údajom skúmanej oblasti, resp. umožnené s informáciami pracovať a využívať ich, sa používa geoinformačný systém. V príslušnej geodatabáze sa ukladajú ako výsledok mnohých zvažovaných otázok veľké množstvá dát a sú takto k dispozícii pre ďalšie použitie a spracovanie. Informačný systém o kvalite vody WAQIS predstavuje presne takúto databanku. WAQIS-projekt buduje Inštitút pre hydrológiu Albert-Ludwigs-Universität vo Freiburgu (IFH) v spolupráci s lokálnym zásobovateľom pitnou vodou badenova-AG.

Staufener Bucht na juhu a Zartener Becken na východe Freiburgu sú oblasti získavania pitnej vody v priestore Freiburgu. Na modelové zachytenie týchto oblastí sa používa simulátor FEFLOW (Finite Elemente Flow Simulation Program). V rámci projektu WAQIS bol v diplomových prácach vyvinutý celý rad modulov a nástrojov (BOMET, BOMET\_BS, OGRUT, HGEO), ktoré ale neboli používateľovi centrálne k dispozícii.

Cieľom tejto práce je vytvorenie takejto plochy a jej prehľadnej organizácie. Pre integrálne riešenie bola okrem toho vytvorená možnosť, zviditeľniť nastavenia FEFLOW parametrov aj v GIS. Implementáciou tohto vizualizujúceho nástroja získava užívateľ možnosť používať nielen ponúkané funkcie, ale si tiež vytvoriť pomocou nastavení ľubovoľný priestorový obraz príslušného modelu. Pre realizáciu bol vyvinutý ďalší modul MODINT (integrácia modelu) ako softvérová aplikácia WAQISu pod SMALLWORLD GIS. SMALLWORLD sa pre tento účel zvlášť hodí pre svoju výkonnú databázovú architektúru.

MODINT bol napísaný v MAGIKu, objektovoorientovanom programovacom jazyku SMALLWORLDu. Kľúčový význam tohto programovaného modulu je pripojenie na FEFLOW. K jeho uskutočneniu boli napísané parzery (syntaktické analyzátory), ktoré čítajú a analyzujú výstupné súbory simulátora FEFLOW. Týmto sú jednak \*.fem – súbor, ktorý je generovaný pri predbežnom spracovaní modelu podzemných vôd a ktorý obsahuje všetky informácie, potrebné pre modelovanie (rámcové podmienky, časové kroky, priestorové rozmiestnenie hodnôt parametrov). Ďalej sa pri spracovaní vo FEFLOW vytvára \*.dac súbor, v ktorom sú okrem usporiadaní modelu obsiahnuté všetky výsledky ku každému časovému kroku.

MODINT rozumie priestorové rozloženie hodnôt parametrov a priradzuje ich príslušnej Finite-Element-sieti na vizualizáciu. MODINT pri tom používa pre BOMET napísaný style-editor, v ktorom je možné prideliť numerickým hodnotám parametrov samodefinované farby. Nakoniec sa používateľovi zobrazí priestorové rozmiestnenie zvoleného parametra skúmanej oblasti v GIS ako farebný priebeh.

Zjednotením komponentov tvoriacich SMALLWORLD GIS, ktoré boli vyvinuté v IHF a integráciou simulátora FEFLOW dopĺňa táto práca ponuku WAQIS. K tejto diplomovej práci je okrem toho pripojená príručka, ktorá ponúka užívateľom, šikovným programátorom, ale tiež zvedavým laikom informáciu o tom, ako MODINT pracuje.

### **Kľúčové slová**

Geografický informačný systém  
SMALLWORLD GIS  
Modelovanie podzemných vôd  
FEFLOW  
Objektovo-orientované programovanie  
MAGIK  
Integrácia modelov  
Parzer  
MODINT  
WAQIS-projekt

Übersetzerin: Zuzaná Zemberova

## 摘要: / Zusammenfassung auf chinesisch

地下水的人为的需求压力要求它不仅有意识的使用,而且要求怎么确定鉴于饮水需求量能安全地提取的水的数量的选择。地下水流动模式能由模拟的使用而近似(数字地)数学化描述。模拟基于被输入数据,这些数据在此领域的获得对于计算是很必要的。对于地下水的维护,存在着许多问题,例如怎么保存土壤和蓄水层,怎么保存水的质量,怎么预知表面水的影响,或地下水的再生问题。而且,更多的数据由于把田地分到参数和边界条件模拟产生。为了允许用户便于获得调查区域的数据并且利用信息,一个地理的信息系统(GIS)被使用。作为许多被考虑了的问题的结果,大量数据被产生,这些数据必须在各自的地理数据库被保存以便它为更进一步地应用。水质信息 WAQIS 代表这地理信息数据库。WAQIS 工程被水文研究所在 Albert-Ludwigs-福来堡大学进行,与本地的水供应商有合作。

福来堡的淡水区域位于南方的 Staufen 盆地以及东方的 Zarten 盆地。模拟程序 FEFLOW (有限单元流动模拟程序)被用来描述这一区域。在 WAQIS 工程期间,众多的调查被施行并且若干模块和工具被开发(BOMET, BOMET\_BS, OGRUT, HGEO)它从中央的表面是不可接近的。这个工程的一个目的是,创造一个有好的界面给现行的被完成的程序。除此以外, GIS 的 FEFLOW 参数调整的可视化的能力被建立。这个可视化工具的实现不仅使用户能够使用提供的方法,而且在考虑任何模型时得到任何参数值空间分布的概述。为了执行这第 2 步,另外的模块叫了 MODINT (模型集成)在 SMALLWORLD GIS 下面作为 WAQIS 的一个软件应用程序被开发。由于它的有力的数据库结构, SMALLWORLD 非常适合于这项任务。

MODINT 被 MAGIK 编辑,它是 SMALLWORLD 的面向对象的程序语言。在 MODINT 里, FEFLOW 的集成是其关键。在它的创造过程中,分析器被编辑,它能读并且分析 FEFLOW 的产量文件。在一方面,有\*.fem 文件在地下水模型的预处理期间被产生的\*.fem 文件包含了被要求计算结果的整个的信息。在另一方面,在 FEFLOW 的处理过程中产生的\*.dac 文件除了模型调整所有的结果外还包括了时阶。

参数价值的空间的分发被 MODINT 理解并且分到了各自的有限的单元网孔。这样做, MODINT 使用为 BOMET 扩展的类型编辑,它使自定义颜色的数值成为可能。最后,用户能作为颜色范围在 GIS 看见一个选择的参数的空间的分布。

使 GIS 某些 FEFLOW 集成部件标准化归功于 WAQIS。对于接下来的论文,是关于对于用户,聪明的程序员甚至好奇的新手了解 MODINT 如何工作的。



## 要旨 / Zusammenfassung auf japanisch

地下水への人為的負担は、とりわけ飲料水獲得のために、その程度を明らかにする必要がある。地下水の流動状態は、数学的に（数値として）モデルで表現できる。そのモデルは、調査地より得られた、分析に必要なデータに基づいて作られる。地下水資源の保全には、土壌、帯水層、水質、表面水の影響、地下水の生成過程など様々な要因が関わってくる。そのデータに加え、さらにその地域特有の条件が、モデル作成の際に重要となる。

GIS (Geoinformationssystem) を使うことによって、調査地のデータを簡単に扱えるようにした。結果として出た、莫大なデータは、それに属するジオデータバンクに保存され、更なる利用やデータ処理を容易にした。水質に関する情報システム (WAQIS) も、このジオデータバンクにあたる。このWAQISプロジェクトは、フライブルク アルベルト ルードヴィクス大学水文学部 (IHF) と、エネルギー・給水株式会社 - フライブルク地域飲料水 - の、共同研究である。

フライブルク地域の飲料水供給地域は、南のシュタウフェナー湾と、東のツアルテナー貯水池である。この地域のモデル化には、シミュレーション FEFLOW (Finite Element Flow Simulation Program) が使用された。WAQISプロジェクトについては、もうすでにいくつかの修士論文において、モデルやツールが開発されているが (BOMET、BOMET\_BS、OGRUT、HGEO)、一般的なレベルではまだ実用的でない。この論文の目的は、一般的なレベルでモデルを実用可能にすることにある。

総合的な解決法として、FEFLOWのパラメーターをGISで扱うことによって視覚的に分かりやすくした。このようにビジュアル化することによって、好みの方法で必要なモデルを作ることができる。WAQISのソフトウェア利用としての更なる融合モデル (Modul MODINT) は、SMALLWORLD GISによって開発された。このSMALLWORLDは、特にデータバンク構造の能率が良い。

MODINTは、対象物の方向づけをするプログラム言語、SMALLWORLDによってMAGIKで書かれた。このプログラム化されたモデルは、FEFLOWとのつながりにその意味がある。実現化のため、FEFLOW出力データを翻訳、分析処理するパーザーが書かれた。これは、\*.fem-Dateiで、地下水モデルの前過程で作成され、モデル化に必要な情報（周辺事情、時間的経過、パラメーターの価値における空間的配分）が書いてある。また、FEFLOWの過程で、モデル

の成果を除く全ての時間的過程における全ての結果が書かれている、\*.dac-Datei が作成された。

パラメーターの価値における空間的配分は MODINT によって解釈され、ビジュアル化においては、それに属する有限要素網に分類される。その際、パラメーターの数値を自分で定義した色で表すことのできるスタイル-エディター、BOMET のための地下資源モデルの発展領域における、MODINT を使用する。最終的に、空間的配分は、それぞれの地域の値が GIS において色ごとに表現することができる。

IHF で開発された SMALLWORLD GIS 構成要素と、FEFLOW シミュレーションの統一化によって、WAQIS の提案を補足した。また、この修士論文に、利用者、プログラマー、興味を持っているアマチュアのため、MODINT の機能についてのハンドブックを付加した。

## キーワード

Geoinformationssystem

SMALLWORLD GIS

地下水モデル

FEFLOW

対象物の方向づけをするプログラム

MAGIK

融合モデル

パーザー

MODINT

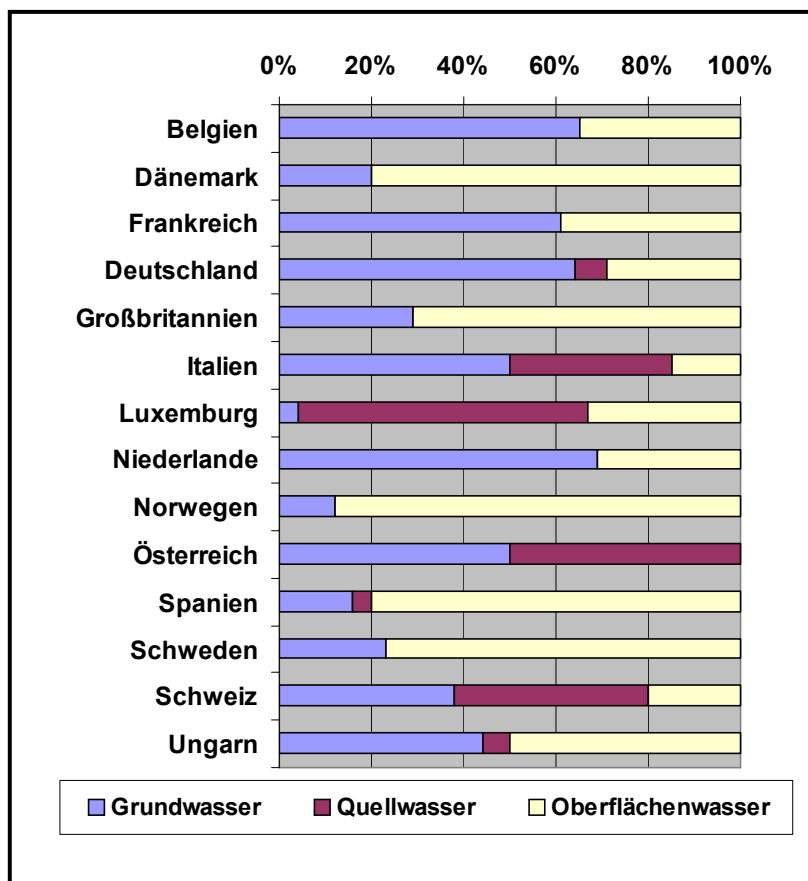
WAQIS プロジェクト

Übersetzerin: Shizu Kaga

# 1 Einleitung

Der Begriff Wasserkreislauf drückt die ständige Zustands- und Ortsveränderung von Wasser in der Erdkruste, auf der Erdoberfläche und in der Atmosphäre aus. Von der ca.  $1,39 \cdot 10^9 \text{ km}^3$  Gesamtwassermenge der Erde, sind nur rund  $4 \cdot 10^6 \text{ km}^3$ , also etwa 3 % des gesamten Wasservorrates als aktives Grundwasser verfügbar. Eine wichtige Aufgabe der Wasserwirtschaft ist die Sicherung der Nutzung von Grundwasser als Brauchwasser (Bewässerung, Industrie, Wasserkrafterzeugung) und als Trinkwasser. Abbildung 1.1 zeigt die Anteile von Grund-, Quell-, und Oberflächenwasser an der gesamten Trinkwasserentnahme im internationalen Vergleich. In Abbildung 1.1 wird eindrucksvoll die große Bedeutung des Grundwasseranteils (ca. 63 %) als Trinkwasserquelle in Deutschland illustriert. (Baumgartner und Liebscher, 1996)

Grundwasser ist in Südwestdeutschland reichlich und in guter Qualität verfügbar. Es



besteht keine unmittelbare Gefahr, dass die hiesigen Grundwasservorräte zur Neige gehen. Gleichwohl wurde in den letzten Jahren und Jahrzehnten eine erhöhte anthropogene Belastung der Umwelt aus Industrie und Landwirtschaft festgestellt. Infolgedessen wurde der Schutz von Aquifer sowie der Bodenschutz als zentrale Aufgabe angesehen, um damit der Gefährdung unseres Trinkwassers vorzubeugen. Seitens der Trinkwasserversorger sind Anstrengungen erforderlich, die eine Antwort auf den Konflikt zwischen Wassernutzung und intensiver anthropogener Nutzung gibt.

Abbildung 1.1: Trinkwasserentnahme in Deutschland im internationalen Vergleich (nach Water Services Association, 1996)

Der lokale Trinkwasserversorger *badenova-AG* bewirtschaftet die Trinkwassergewinnungsgebiete Staufener Bucht, im Süden von Freiburg, und das Zartener Becken, in Freiburgs Osten. Die *badenova-AG* hat zur Bereitstellung von Auswertewerkzeugen von wasserwirtschaftlichen Fragen die Erstellung eines *Wasserqualitäts-Informationssystems (WAQIS)* in Auftrag gegeben, das am Institut für

Hydrologie Freiburg erarbeitet wird. Erstellt werden soll eine Geodatenbank unter SMALLWORLD GIS mit Funktionalitäten, die auf wasserwirtschaftliche Daten geeignet zugreifen.

WAQIS baut auf der Anwendung LIWIS auf, welche die benötigten Kernmodule stellt. LIWIS ist ein modular aufgebautes wasserwirtschaftliches Teilinformationssystem, das auf SMALLWORLD GIS basiert. Es wurde von der Firma ERM Lahmeyer International entwickelt und bereitgestellt.

Somit ist WAQIS eine Erweiterung von LIWIS, die eigene Module enthält. WAQIS impliziert Informationen über Grundwasserdaten inklusive Qualität des Grundwassers, Bodenschutzkenngößen, meteorologische Daten sowie Aquiferparameter. In jüngster Zeit wurden am IHF zahlreiche Module und Tools entwickelt, die das WAQIS nach und nach erweitert haben. Die Parametrisierung und Übergabe von Modellkenngößen an den in WAQIS verwendeten Grundwassersimulator FEFLOW wurde schon realisiert. Was fehlt ist die Rücknahme von Modellergebnissen an das GIS. Mit dem Zugriff auf die Ergebnisse aus SMALLWORLD heraus soll eine Schnittstelle zu FEFLOW den integrativen Gedanken vervollständigen.

Im Rahmen der vorliegenden Arbeit wurde somit MODINT erstellt. Dieses Modul dient der Kommunikation zu FEFLOW sowie der Visualisierung von Modellkenngößen. MODINT baut auf dem bereits bestehenden Tool OGRUT auf. Alle OGRUT-Funktionalitäten werden in die Oberfläche des Moduls MODINT eingebaut.

Diese Arbeit liefert somit einen Beitrag zur Benutzerfreundlichkeit der Anwendungen in WAQIS sowie zur Implementierung des Simulators FEFLOW in SMALLWORLD GIS.

## 2 Problemstellung und Zielsetzung

Bereits mehrere Fachanwendungen mit integrierten Auswertewerkzeugen wurden im Rahmen des WAQIS erstellt. Einen Überblick über das WAQIS gibt Abbildung 2.1.

Das 1999 von K. Eberle entwickelte und 2001 von W. Klink um die Implementierung von Bodenschätzdaten zum BOMET\_BS erweiterte bodenkundlich-meteorologische Modul BOMET liefert bodenphysikalische und meteorologische Parameter und ist somit ein Teil des WAQIS von großer Bedeutung. Das Modul OGRUT, 2000 von J. Herrmann erarbeitet, legt die Parametrisierung von FEFLOW fest und liefert mit der mathematisch erfassten Gerinnehydraulik von Oberflächengewässern die Randbedingungen bei der Berechnung der Grundwasserneubildung und von Überflutungsflächen. Ferner liefert das Modul HGEO (J. Kiefer, 2001) einen entscheidenden Beitrag einerseits zur Verwaltung von hydrogeologischen und geologischen Sachdaten wie Bohrdaten oder Schichtgrenzen und andererseits zur Visualisierung geologischer Schichten und anderer geologischer Kenngrößen.

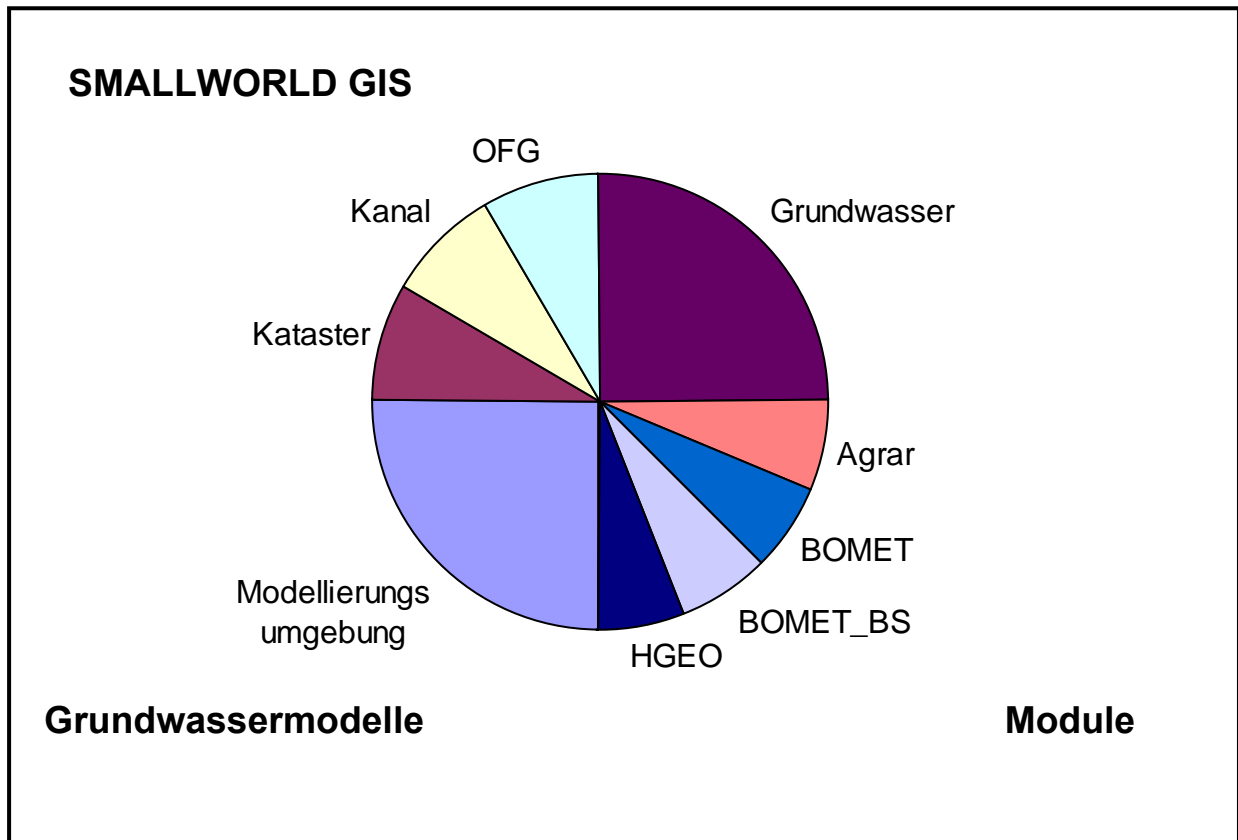


Abbildung 2.1: Überblick über das Projekt WAQIS (ergänzt nach WAQIS, 2000)

Für beide von der badenova-AG verwalteten Trinkwassergewinnungsgebiete wurden bereits bis 1993 (Zartener Becken) bzw. bis 1997 (Staufener Bucht) stationäre Grundwasserströmungsmodelle mit FEFLOW erstellt. Ziel der vorliegenden Arbeit ist es die

Kommunikation des Simulators FEFLOW mit SMALLWORLD GIS zu gewährleisten. Sind im Grundwassermodell selber Ergebnisse und Parameterverteilungen nur zur Modelllaufzeit sichtbar, soll dies unabhängig vom Modell im GIS geschehen. Ohne Aufrufen des Modells soll die Visualisierung und Verarbeitung der Modellergebnisse von der GIS-Plattform aus möglich sein. Die Schnittstelle soll dazu dienen durch den Zugriff auf alle Ergebnisse die Informationsfülle für den Anwender zu erhöhen. Die Informationsmenge wird insbesondere dann erweitert, wenn mit den Ergebnissen andere zu untersuchende Größen (z.Bsp. Schwankung des Grundwasserstandes, Minima, Maxima) erzeugt werden können.

Im Rahmen der vorliegenden Arbeit soll eine zentrale graphische nutzerfreundliche Oberfläche (graphical user interface, GUI) erstellt werden, von der aus der Anwender auf einfache Art und Weise auf die einzelnen Funktionen zugreifen kann ohne den mathematischen und programmiertechnischen Hintergrund zu kennen.

In einem ersten Schritt der Arbeit sind zur Realisierung der Modellintegration Parser zur Datei-Analyse zu programmieren. Es soll problemlos auf Outputdateien des FEFLOW-Preprozessors sowie des FEFLOW-Prozessors zugegriffen werden können. Insbesondere ist die \*.dac-Datei von zentralem Interesse, da in ihr die Modellergebnisse stehen. Die Inhalte dieser Datei sollen in Kurzform wahlweise in eine selbst deklarierte Outputdatei mit der Endung \*.eco geschrieben, auf dem Bildschirm ausgegeben und/oder in die Datenbank übernommen werden. Die \*.eco-Datei sollte in allen Fällen vor weiteren Arbeits- oder Analyseschritten geprüft werden, um sicher zu stellen, dass die eingelesene \*.dac-Datei keine Fehler aufweist.

Des weiteren ist eine zentrale Schaltstelle zu entwickeln. Sie wird repräsentiert durch einen nutzerfreundlichen Editor, von dem aus Funktionen, Hilfe, Status von Prozessen sowie ausgewählte Optionen dem Nutzer bereitstehen.

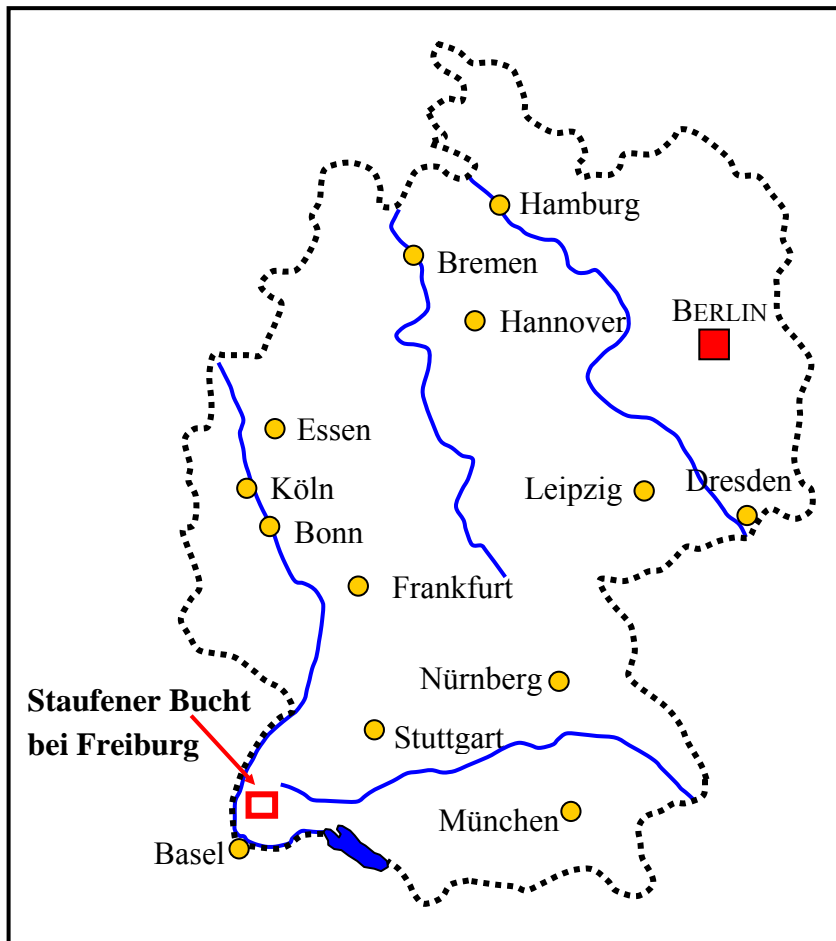
Um die Visualisierung der Modellparameter zu realisieren ist der von Herrmann (2000) entwickelte Einlesealgorithmus eines Finiten Elemente Netzes zu korrigieren und in MODINT zu implementieren. Um den Elementen dieses Netzes darzustellende Werte zuzuweisen, ist ein Code zu schreiben, der jedes einzelne Element in der Datenbank identifiziert und ihm den zugehörigen Wert zuordnet. Dies wird beispielhaft für das Trinkwassergewinnungsgebiet des Einzugsgebietes des Wasserwerkes Hausen an der Möhlin, in der die Staufener Bucht liegt, dargelegt.

Mit der Entwicklung eines Werkzeuges zur Ergebnisrücknahme wird der Gedanke der Modellintegration ergänzt. Mit dessen Realisierung und mit der Bereitstellung der Funktionalitäten über eine zentrale Nutzeroberfläche liefert diese Diplomarbeit einen Beitrag zur Weiterentwicklung des WAQIS.

### 3 Anwendungsgebiet und bisherige Arbeiten

#### 3.1 Anwendungsgebiet

Die Staufener Bucht liegt im Südwesten von Baden-Württemberg, südlich der Stadt Freiburg (Abbildung 3.1). Sie ist Teil des Trinkwassergewinnungsgebietes des Wasserwerkes



in Hausen an der Möhlin. Dieses Gewinnungsgebiet liegt auf den TK50-Blättern 7910, 7912, 8110 und 8112. Die Staufener Bucht ist keine Bucht im engeren Sinne, da das Gebiet nach Westen nicht von Berg- oder Hügelland begrenzt ist. Vielmehr handelt es sich um das Ende des Münstertales, das in Staufen ausmündet. Die Staufener Bucht unterteilt die etwa 7 km lange und 1 km breite Sulzburg-Staufener Vorbergzone. Hierzu gehören die Pultschollen des Kastelbergs und des Fohrenbergs. Bekannt ist vor allem der Staufener Schloßberg, der einen guten Überblick auf diesen Teil der Vorbergzone gestattet. (Geyer und Gwinner, 1991)

Abbildung 3.1: Lage des Untersuchungsgebietes

Die Staufener Bucht ist neben dem Zartener Becken, im Osten von Freiburg gelegen, das Trinkwassergewinnungsgebiet für die Stadt Freiburg im Breisgau. Der poröse Aquifer wird aufgebaut durch kristalline Schotter aus dem Schwarzwald und durch alpines karbonatisches Material. Hier kommt es zu einer Verzahnung der verschiedenen Schottermaterialien. Beide Schottertypen wurden hier zeitgleich am Ende der letzten Eiszeit im Jungpleistozän abgelagert, haben aber unterschiedliche Herkunftsräume.

Der lokale Trinkwasserversorger badenova-AG entnimmt am Wasserwerk Hausen an der Möhlin 6 Mio m<sup>3</sup> Wasser pro Jahr zur Versorgung des westlichen Teils von Freiburg. Der andere Teil wird vom Wasserwerk Ebnet versorgt (12 Mio m<sup>3</sup> pro Jahr), dessen Gewinnungsgebietes das Zartener Becken ist.



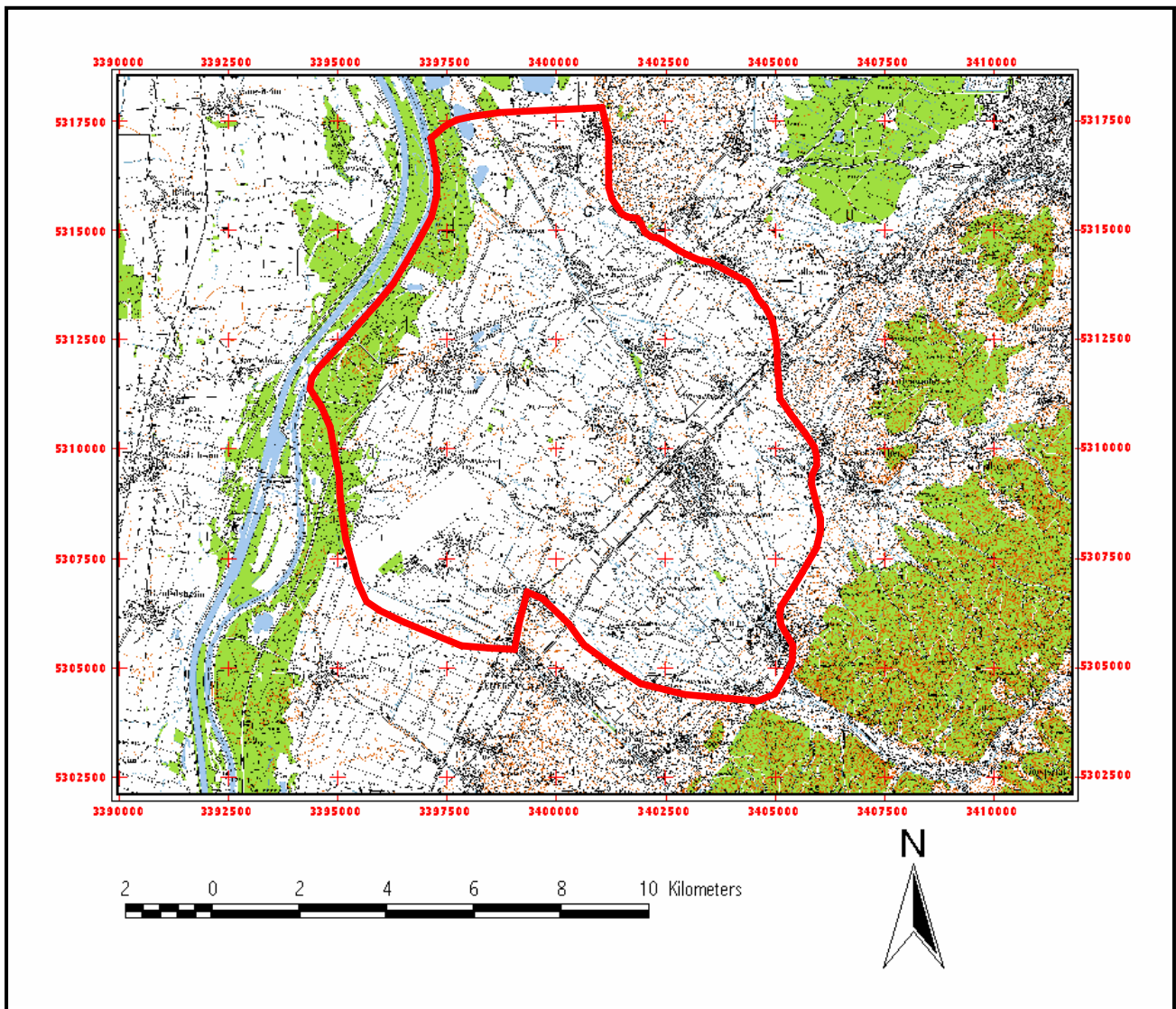


Abbildung 3.2: Das Untersuchungsgebiet südlich von Freiburg

## 3.2 Bisherige Arbeiten

### 3.2.1 Erstellung eines Modells für das Untersuchungsgebiet

Für das Einzugsgebiet des Wasserwerkes Hausen wurde im Rahmen einer anderen Diplomarbeit ein dreidimensionales, instationäres Grundwasserströmungsmodell erstellt. Verwendet wurde der Simulator FEFLOW. Mit dem kalibrierten Modell können die Strömungsverhältnisse über den Zeitraum Januar 1987 bis Oktober 1989 sehr gut nachgebildet werden. Das Modell steht somit als Werkzeug zur nachhaltigen Grundwasserbewirtschaftung zur Verfügung. (Peter, 1998)



### 3.2.2 Das bodenkundlich-meteorologische Modul BOMET

Das bodenkundlich-meteorologische Teilinformationssystem BOMET (Eberle, 1999) dient der Verwaltung aller den Trinkwasserschutz betreffenden Bodendaten in einer Geodatenbank. Außerdem stellt es Werkzeuge bereit, die den Umgang mit diesen Daten möglich machen, insbesondere das Hinzufügen von physikalischen und chemischen Eigenschaften der Pedosphäre. Von zentraler Bedeutung ist hier die Grundwasserneubildung als Eingangsgröße für Grundwassermodelle und als das Maß für die Erneuerung der Grundwasserressourcen. Auch trägt das Modul BOMET der Problematik der Nitratauswaschung ins Grundwasser Rechnung. Das Stickstoffmodell *EXPERT-N*, das von der Lehrereinheit für Ackerbau und Informatik im Pflanzenbau der TU München-Weihenstephan und dem Institut für Bodenökologie des GSF-Forschungszentrums für Umwelt und Gesundheit entwickelt wurde, wurde gewählt, um den Stickstoffkreislauf zu modellieren. Die Kopplung von *EXPERT-N* an *SMALLWORLD GIS* wurde realisiert. (Eberle, 1999)

### 3.2.3 Die BOMET-Erweiterung BOMET\_BS

In Trinkwassergewinnungsgebieten herrscht infolge doppelter Nutzung des Gebietes als landwirtschaftliche Nutzfläche und als Grundwasserentnahmegebiet grundsätzlich immer Konfliktpotenzial. Hier gewinnt der Boden als Schnittstelle zwischen beiden Nutzungen an Wichtigkeit. Bisher gab es nur die *Bodenkundlichen Kartierung* als bodenkundliche Datenquelle. Um noch mehr pedologisch relevante Informationen greifbar zu haben, wurde im Rahmen von BOMET\_BS die Möglichkeit geschaffen, bodenkundliche Informationen auch aus dem Feldschätzbuch der Reichsbodenschätzung zu erhalten. Der Übersetzungsschlüssel, der dem Feldschätzbuch zugrunde liegt, stammt allerdings aus den 30er Jahren. Daher wurden Verfahren in BOMET implementiert, die es erlauben, Bodenschätzdaten aufzubereiten um so als brauchbare Daten zur Verfügung zu stehen.

Außerdem wurde ein Visualisierungstool geschaffen, mit Hilfe dessen Elemente eines Finiten Elemente Netzes (FE-Netz) mit numerischen Werten belegt werden können. Auf diese Weise können zum Beispiel Modellergebnisse (Grundwasserstände) oder Modellparameter im GIS dargestellt werden. Im Rahmen dessen entstand das Modul *VIS\_TOOL*. (Klink, 2001)

### 3.2.4 Das Oberflächen- und Grundwassermodul OGRUT

OGRUT führt die Parametrisierung des Simulators FEFLOW durch. Grundlage dafür ist die Berechnung der Gerinnehydraulik. Mit Hilfe dessen ist die Grundwasserneubildung quantifizierbar, wobei OGRUT zweierlei Ursachen von Grundwassererneuerung berücksichtigt: zum Einen den flächenhaften Niederschlag, zum Anderen Überflutungsflächen. Dazu ist es notwendig die überfluteten Flächen zu berechnen. Es wurde das Programm *UWSP* entwickelt, welches die Wasserspiegellinie von Oberflächengewässern im ungleichförmigen Fall (ortsvariabel) berechnet.

Ferner wurde ein Parser<sup>1</sup> geschrieben, der es erlaubt ein aus zwei ASCII-Dateien<sup>2</sup> bestehendes Finites Elemente Netz einzulesen. Das FE-Netz kann in die Datenbank von SMALLWORLD GIS geschrieben werden und im GIS angezeigt werden. So wurde die Grundlage dafür geschaffen andere sich auf das eingelesene FE-Netz beziehende Daten im GIS darzustellen. (Herrmann, 2000)

Die Funktionalitäten aus OGRUT wurden in die Oberfläche von MODINT eingebaut.

### **3.2.5 Das Hydrogeologie Modul HGEO**

HGEO verwaltet sämtliche geologischen und hydrogeologischen Daten. Über eine Eingabemaske können die Geologiedaten eingegeben werden. Es wurde das Werkzeug IHF-ISOLINE entwickelt, das Isolinien erstellt und verwaltet. Die Erstellung der Gleichlinien erfolgt extern über das Programm SURFER, in welchem verschieden Algorithmen der Interpolation anwendbar sind. Das Zeichnen von Isolinien wird in HGEO exemplarisch an der Ermittlung geologische Schichtgrenzen und deren Darstellung im GIS vorgeführt. Weiterhin existiert die Möglichkeit der Visualisierung von geologischen Schnitten entlang einer an der Erdoberfläche frei wählbaren Linie. Schließlich ist in HGEO auch die Zuweisung numerischer Werte (z. B. Grundwasserstände) an Knoten eines FE-Netzes implementiert. (Kiefer, 2001)

---

<sup>1</sup> vgl. dazu auch Kapitel 6.4: Geschriebene Parser

<sup>2</sup> knoten.txt und elemente.txt

## 4 SMALLWORLD GIS

### 4.1 Das GIS und dessen Architektur

Um räumliche Geometrien realitätsnah darstellen zu können, wählt man für diese geometrischen Objekte ein Koordinatensystem, das für alle darzustellenden Objekte denselben Maßstab, dieselben Maßeinheiten und denselben Koordinatenursprung hat. Dieses Koordinatensystem wird *GIS* oder *geographische Welt* genannt. Darüber hinaus kann ein GIS zusätzliche Geometrietypen mit demselben Koordinatensystem wie Entwurfslinien und Hintergrundraster beinhalten. Der Anwender kann ferner auf eine Reihe von Koordinatensystem-Optionen zurückgreifen, die die Ansicht geographischer Objekte bearbeiten. Schließlich ist der Benutzer in der Lage von der GIS - Oberfläche aus spezifische Funktionalitäten aufzurufen und auszuführen. Bei der vorliegenden Arbeit wurde das GIS *SMALLWORLD* Version 3.1 Service Package 2 verwendet.

Das Erstellen und Laden von Objekten geschieht in *SMALLWORLD* unter Verwendung der Programmiersprache *MAGIK* im *EMACS*, der Entwicklungsumgebung von *SMALLWORLD GIS*. Gestützt wird die Objekt-Entwicklung durch das CASE-Tool (vgl. Kapitel 4.4: Das CASE - Tool). Die Möglichkeiten zur Einstellung der Konfiguration im GIS bietet das ACE (Application Configuration Environment). Hier ist das Layout des Menüs (Button- und Menüwahldarstellung), die Konfiguration der Objekte (accessible, sichtbar, fangbar, auswählbar) sowie die Anwendung einzustellen. In *SMALLWORLD* sind *Datenbanken* die entscheidende Instanz zum speichern von Objekten (vgl. Kapitel 4.5: Die Architektur der Geo - Datenbank). Ein erstelltes Objekt oder ein fertiges Datenmodell (vgl. Kapitel 4.4) kann auf eine Datenbank nach Wahl des Benutzers *appliziert*, also angewandt oder gespeichert werden.

In *SMALLWORLD GIS* liegt der Schwerpunkt auf Objekten und deren Verhalten. Daher ist *SMALLWORLD GIS* nicht *Layer*-orientiert wie z.B. *ARC/View* sondern *Objekt*-orientiert. Eine vereinfachte *SMALLWORLD GIS*-Architektur zeigt Abbildung 4.1. (GE Smallworld, 1999b)

Grundsätzlich existieren zwei verschiedene Arten von Objekten: Objekte ohne Geometrie, also ohne geometrische Zuordnung im GIS-Fenster, und Objekte mit Geometrie oder mit räumlicher Zuordnung im GIS-Fenster. Derartige geometrische Objekte heißen Real World Objects (RWOs), da sie in den meisten Fällen Objekte in der realen Welt darstellen. RWOs sind daher Benutzerobjekte mit Regeln, die das Objektverhalten sowie die Beziehung der Objekte zwischeneinander definieren.

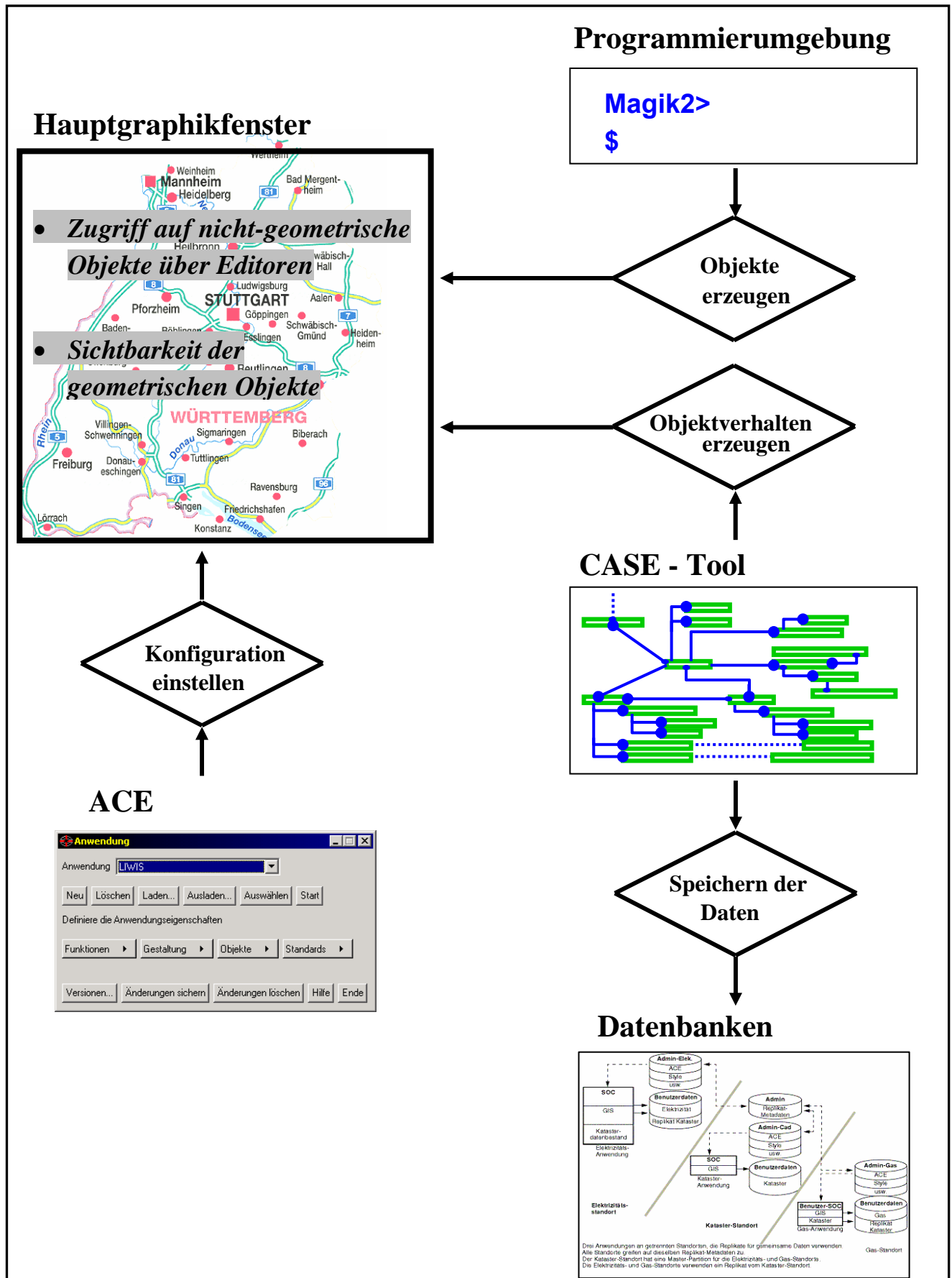


Abbildung 4.1: Überblick über die vereinfachte SMALLWORLD GIS-Architektur

## 4.2 Der EMACS: Entwicklungsumgebung in MAGIK

Eine GIS-Anwendung umfasst das Basisprodukt GIS und zusätzliche Softwarekomponenten, kundenspezifischen Anwendungscode und spezielle Objektklassen in einem Datenmodell. Darüber hinaus stellt eine GIS-Anwendung kundenspezifische Wiedergabe-Styles, Menüs und Meldungen bereit. Die Programmierung von Objekten und GIS-Anwendungen geschieht in SMALLWORLD GIS mit der objektorientierten Programmiersprache MAGIK in der Programmierumgebung EMACS. Ursprünglich wurde der EMACS als Textbearbeitungsoberfläche verwendet. Entwickelt wurde er vor ca. 30 Jahren von Richard Stallmann aus Boston.

Die Stärken von EMACS kommen erst bei der Bedienung mittels Tastenkombinationen zur Geltung. Man kann mit dem EMACS über die Tastatur schnell und elegant arbeiten. Das gilt in erster Linie für das Editieren von Programmcode, wie im vorliegenden Fall das Erstellen von MAGIK Quellcode. (Cameron und Rosenblatt, 1991; Glickstein, 1997)

EMACS (Abbildung 4.2) ist ein Werkzeug zur raschen und sicheren Durchführung auch komplizierter Textoperationen. Besonders geeignet ist es für MAGIK, aber auch für andere Skriptsprachen wie PERL, TCL oder Struktursprachen wie SGML.

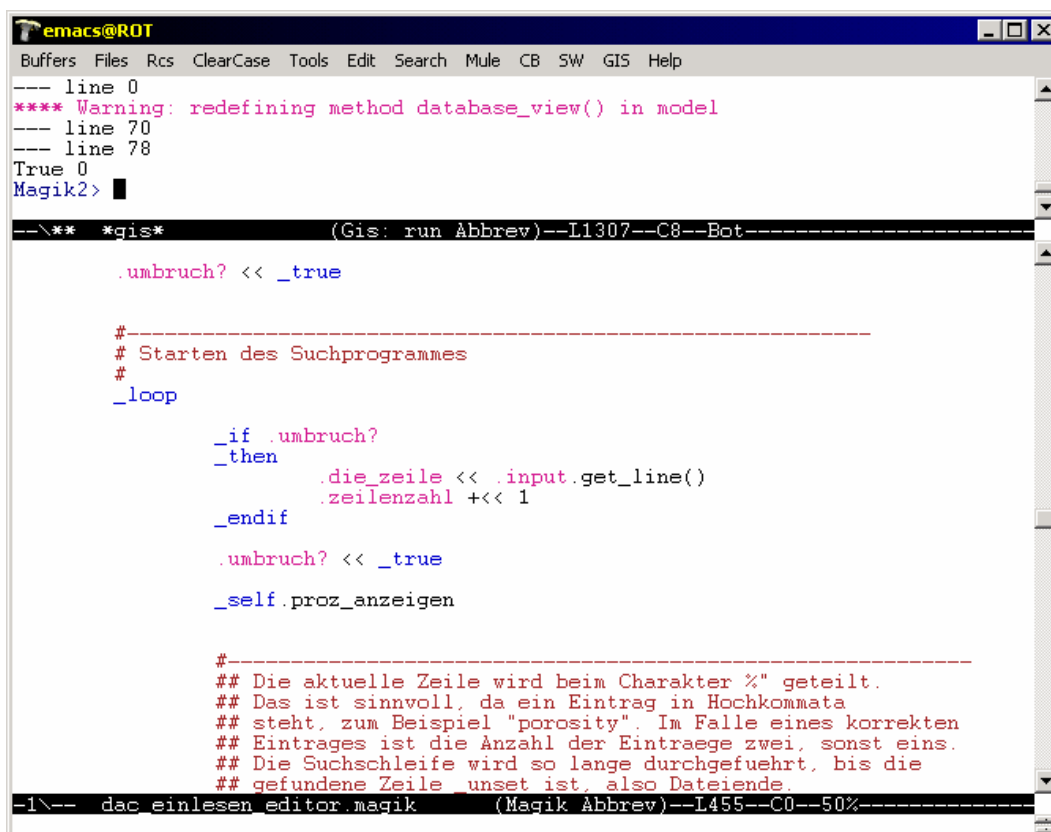


Abbildung 4.2: Die Entwicklungsumgebung EMACS

Einige spezielle Fähigkeiten von EMACS sind:

- EMACS kann über Konfigurationsdateien praktisch unbegrenzt an eigene Bedürfnisse angepasst werden. Alle Konfigurationen werden in der Programmiersprache LISP erstellt.

- EMACS bietet Makros und Tastenkürzel für oft benutzt oder oft falsch geschriebene Sequenzen.
- EMACS eröffnet im *class browser* die Möglichkeit nach Programmteilen (Methoden) zu suchen und diese anzuschauen.
- EMACS stellt den Debugger *magik studio* zur Verfügung. Damit ist das gezielte Suchen nach Programmfehlern auch während der Durchführung eines Prozesses möglich.

Der EMACS ist programmiert in der Skriptsprache LISP, man spricht daher auch von ELISP. Er läßt sich mittels LISP praktisch vollständig variieren. Man kann eigene Tastenkürzel hinzufügen oder überhaupt die Art und Weise, wie EMACS funktioniert, komplett ändern. LISP wurde im Jahre 1960 von John McCarthy geschaffen. Um darauf hinzuweisen, dass EMACS plattformunabhängig agiert, also nicht auf einem bestimmten Betriebssystem wie DOS, MACINTOSH oder UNIX basiert, nennt man EMACS auch GNU EMACS. (Cameron und Rosenblatt, 1991; Glickstein, 1997) "GNU" ist eine Art rekursives Wortspiel und bedeutet GNU is not UNIX<sup>3</sup>.

## 4.3 MAGIK

### 4.3.1 MAGIK als Sprache

MAGIK ist nicht nur eine einfache Makro-Sprache, sondern die Entwicklungssprache von SMALLWORLD GIS. Es ist eine rein *objektorientierte* Sprache, die über die Skriptsprachen anderer Systeme hinausgeht. Objektorientiert heißt, dass sowohl Variablen als auch Methoden (Funktionalitäten) bestimmten Objekten zugeordnet werden. Auf diese Weise wird sicher gestellt, dass komplexe Programmstrukturen übersichtlich bleiben. MAGIK gestattet auch die Definition von Prozeduren. Genau genommen handelt es sich also um eine hybride Programmiersprache. MAGIK agiert auf einer abstrakten Ebene, die eine Unabhängigkeit von der verwendeten Hardwareplattform gewährleistet. Zur Zeit sind MAGIK-Interpreter für nahezu alle gängigen Betriebssysteme verfügbar.

Jedes erstellte Objekt wird bei der objektorientierten Programmierung als Instanz einer Objektklasse angesehen. Das bedeutet, die Objektklasse selber definiert den Typus oder die generelle Beschreibung des zugehörigen Objektes. Alle generierten Kindobjekte verfügen über die Eigenschaften, die auf der Objektklasse (Eltern) definiert sind. Derartige Eigenschaften werden durch Vererbungsregeln (vgl. Kapitel 4.3.3: Vererbung), Methoden und Slots festgelegt. Slots sind Variablen, die unabhängig von der Methode auf der Objektklasse definiert sind. Objekte einer bestimmten Klasse können auch auf Objekte einer anderen Objektklasse zugreifen, diese manipulieren oder mit ihnen in anderer Weise interagieren. (GE Smallworld, 1997c)

---

<sup>3</sup> freundliche Mitteilung von Dipl.-Inf. Jason Bechtel, Systemadministrator bei Unique Systems Incorporated in Holland, Ohio, U.S.A.

### 4.3.2 Threads

Gestartete Prozesse können unter MAGIK in sogenannten *Threads* ausgeführt werden. Was sind Threads? Man nehme das Beispiel einer typischen Firma. In nahezu jeder Firma funktionieren mindestens drei unabhängige Abteilungen parallel: Management, Finanzwesen und Herstellung/Verkauf. Damit die Firma effizient läuft, müssen alle drei Abteilungen zur gleichen Zeit arbeiten. Wenn das Finanzwesen seine Funktionen nicht mehr ausführt, wird die Firma zahlungsunfähig. Sollte andererseits die Herstellung ihre Arbeit nicht erledigen, wird die Firma kein Geld mehr verdienen können.

Viele Software-Programme funktionieren nach demselben Schema wie dieses Firmenbeispiel. In einer Firma können mehrere Aufgaben zur selben Zeit erledigt werden, indem Sie die Aufgaben auf mehrere Mitarbeiter verteilen. Jeder Mitarbeiter beschäftigt sich mit seiner ihm zugewiesenen Aufgabe. Software verfügt in der Regel nur über einen einzigen Prozessor, und dieser Prozessor muss mehrere Aufgaben übernehmen. Um dies zu bewerkstelligen, wurde ein Konzept namens *Multitasking* eingeführt. In Wirklichkeit führt der Prozessor einen Vorgang zu einem Zeitpunkt aus und schaltet so schnell zwischen den verschiedenen Aufgaben hin und her, dass der Eindruck entsteht, er würde sie gleichzeitig erledigen. In der Betriebssystemterminologie ist Multitasking als *Multithreading* bekannt. Das Analogon kann man sich folgendermaßen vorstellen: Jedem Programm wird eine bestimmte Person zugeordnet, welche(s) mehrere Aufgaben übernehmen soll. Dies wird als Prozess bezeichnet. Die Person unterteilt ihre Zeit in sinnvolle Abschnitte – das Programm unterteilt seinen Prozess weiter in Threads.

Im vorliegenden Beispiel wurde das gegebenenfalls recht zeitintensive Einlesen langer ASCII-Dateien in Threads geschrieben. Auf diese Weise ist der MODINT-Benutzer in der Lage auf sämtliche anderen ihm zur Verfügung stehenden Tools zuzugreifen. Darüber hinaus kann er über einen Abbrechen Button den Vorgang beenden. Würde der Prozess nicht in einem Thread laufen, müsste zu dessen Beendigung das GIS beendet werden.

Das gesamte GIS ist in MAGIK geschrieben. Daraus eröffnet sich die Möglichkeit, dass die Funktionsweise einer bestimmte Anwendung an die Wünsche des Nutzers angepaßt werden kann.

### 4.3.3 Vererbung

Vererbung ist ein für objektorientiertes Programmieren typisches Kennzeichen. In jeder Objektklasse können Vererbungsregeln festgelegt werden. Diese besagen, von welcher anderen Objektklasse (Elternobjekt als eine Instanz davon) oder von welchen anderen Objektklassen geerbt werden soll. Dabei bedeutet *Erben* die Übernahme von Methoden und Slots in diese Objektklasse (Kindobjekt). Bei erfolgreichem Erben versteht eine erzeugte Instanz der Kindobjektklasse alle Methoden und Slots, die in der Elternobjektklasse definiert wurden. Auf diese Weise hat der Programmierer die Möglichkeit, eine Objektklasse mit bestimmten Eigenschaften zu kreieren und andere Klassen von dieser Elternobjektklasse erben zu lassen. Dann ist gewährleistet, dass alle Instanzen der Kindobjektklasse mit denselben Funktionalitäten ausgerüstet sind. Die Vererbungsregeln werden in der Exemplardefinition (vgl. Kapitel 4.3.4: Definition von Exemplaren und Methoden) festgelegt,

im vorliegenden Beispiel erbt `dac_einlesen_editor` von `ihf_menu_basis` und `engine_model`.

```
def_slotted_exemplar(:dac_einlesen_editor,
                    {
                      { :slot1,      _unset4,      :writable},
                      { :slot2,      _unset,          :writable}
                    },
                    { :ihf_menu_basis, :engine_model })
$
```

#### 4.3.4 Definition von Exemplaren und Methoden

Objektklassen werden definiert wie auf Seite 14 im obigen Beispiel gezeigt wird. Das Definieren einer Objektklasse heißt in MAGIK *Exemplardefinition*. Auf diese folgen die Slots und Vererbungsregeln der definierten Objektklasse. Danach werden die *Methoden* festgelegt, die diese Klasse verstehen soll, was im Code wie unten dargestellt aussehen kann. Vor dem Methodenname steht ein sogenanntes *pragma-statement*. Es definiert den Level der Methode, worunter man die Stufe der Autorisierung dieser Methode versteht. Pragmastatements sollten bei einer Klassen- und Methodendeklarationen immer dabeistehen. Im Anhang in Tabelle A.1 sind die möglichen Levels aufgelistet.

Eine Methode wird nach der allgemeinen Struktur definiert:

```
_pragma(classify_level = < Level >, topic = < Liste der Topics > )
_method < Klasse > . < Methodenname > ( < Übergabeparameter > )
    ##
    ## < Beschreibung der Methode >
    ##
    < strukturieter und auskommentierter Quellcode >
_endmethod
$
```

#### 4.3.5 Editoren

Editoren sind Nutzeroberflächen, von denen aus vordefinierte Funktionen oder andere Auswahlmenüs nach Wunsch des Anwenders aufrufbar sind. Es gibt in MAGIK grundsätzlich zwei Möglichkeiten, einen Editor zu erstellen. Einmal erlaubt das CASE-Tool die Möglichkeit, einer erstellten Objektklasse (Tabelle) einen bestimmten Editor zuzuordnen. Zur Auswahl stehen dann vordefinierte Editoren. Über die Methode

<sup>4</sup> Vor MAGIK-eigene Befehle wird ein Tiefstrich gesetzt



`define_application_buttons()`, die auf der Objektklasse definiert sein muss, lassen sich weitere Einträge in der Option `Extras` einfügen. Alle gängigen Werkzeuge wie Ändern, Holen, Einfügen sind in einem derartigen Editor bereits implementiert. Sollen Oberflächen kreiert werden, deren Oberfläche gemäß des von SMALLWORLD gegebenen Defaults aussehen dürfen, ist dies die einfachste Möglichkeit. Möchte man dagegen das Layout des Editors selber festlegen, ist eine Nutzeroberfläche auch "von hand" in MAGIK durchführbar. Hier steht dem Programmierer völlig frei, wie der Editor auszusehen hat und welche Funktionen aufrufbar sein sollen. Geöffnet wird eine solche Oberfläche mit der Methode `open()`. Wie der Editor aussehen soll, ist in der Methode `activate_in()` bestimmbar. Ein Beispiel dafür steht im Anhang.

#### 4.3.6 Speichereinheiten von SMALLWORLD GIS

Die Programmiersprache von SMALLWORLD GIS ist die komplett objektorientierte Sprache MAGIK. Jedwede Operation in SMALLWORLD schließt das Erstellen von Objekten ein. Dies geschieht beispielsweise im MAGIK Code beim Aufrufen der `new()` Methode. Alle Objekte werden in einer Sitzung in einem Arbeitsspeicherraum gespeichert, der aus zwei Teilen besteht: der statische Speicher (der für relativ langlebige Objekte verwendet wird) und der Ephemeral-Speicher (wo alle Objekte erzeugt werden, ausgenommen sehr große Objekte). (GE Smallworld 1999a)

Der Kernel entspricht dem nicht-änderbaren Betriebssystem.

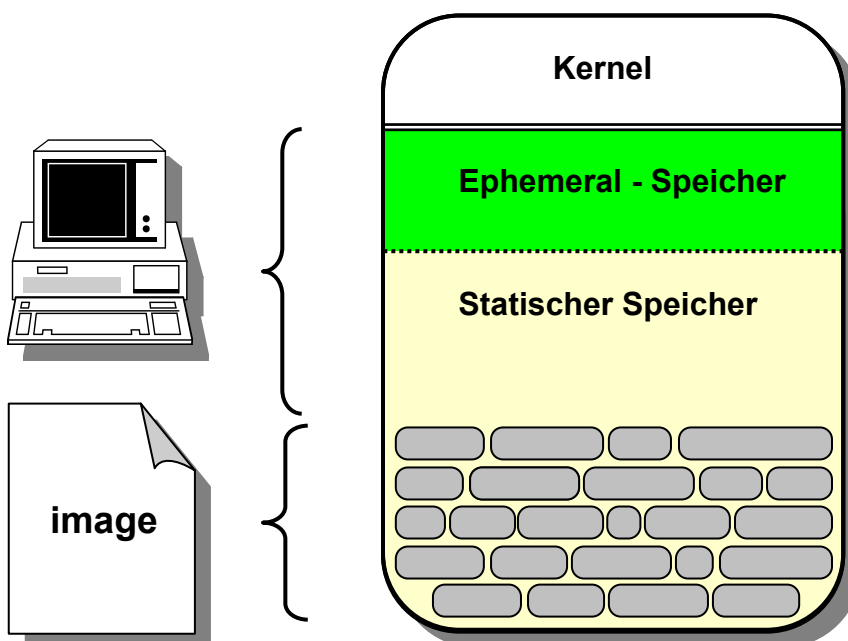


Abbildung 4.3: Verwendung von Speicherplatz in SMALLWORLD GIS (verändert nach GE Smallworld, 1999a)

Durch die Ausführung von MAGIK-Programmen werden immer neue Objekte erzeugt, die anfänglich im Ephemeral-Speicher sind. Nach einiger Zeit werden diese Objekte, sofern sie noch verwendet werden, in den statischen Speicher verschoben. Wenn man eine Sitzung beginnt, wird die Größe des statischen Speichers in der Kommandozeile als -sds -Zahl (in Kilobyte) angegeben. Der gesamte Arbeitsspeicher ergibt sich aus dieser Angabe und aus der Größe des Images.

Mit Fortschreiten von Prozessen im GIS wird der Ephemeral-Speicher und der statische Speicher immer kleiner. Wenn sie ganz erschöpft sind, werden automatisch die Objekte gelöscht, die nicht mehr benötigt werden. Dies wird Garbage Collection genannt. Wenn ein Image gespeichert werden soll, ist zunächst eine Garbage Collection durchzuführen, um die Größe des Images zu verkleinern. Durch eine zweite Garbage Collection wird noch mehr statischer Speicher frei. Einige Objekte, die bei der ersten Garbage Collection entfernt werden, verweisen auf andere Objekte. Wenn nun kein Objekt mehr auf sie verweist, werden diese Objekte von der zweiten Garbage Collection entfernt.

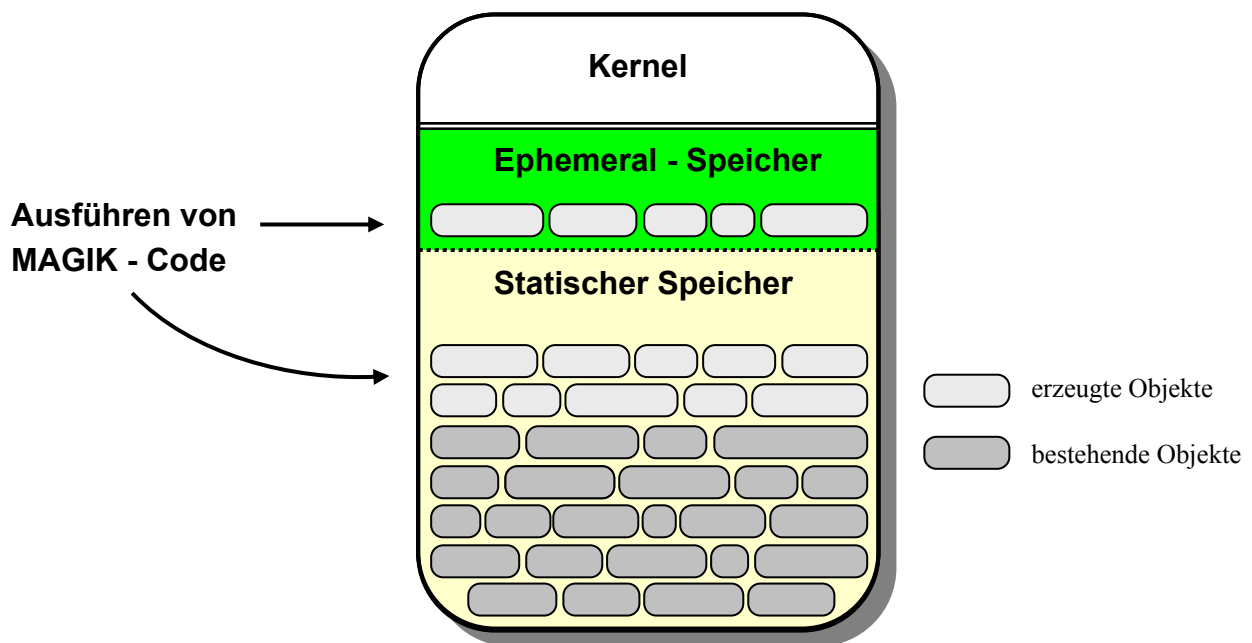


Abbildung 4.4: Verwendung von Speicher beim Erzeugen und Speichern neuer Objekte (verändert nach GE Smallworld, 1999a)

#### 4.4 Das CASE - Tool

In SMALLWORLD ist das CASE-Tool (Computer Aided Software Engineering) ein interaktives Entwicklungssystem zur bequemen Erzeugung von Objekten und Anwendungen. Das strukturierte Entwickeln von Objekten im CASE-Tool wird verwendet, um die Programmierung zu erleichtern und um Konsistenz zu gewährleisten. Inkonsistenz tritt auf, wenn ein Objekt unter verschiedenen Namen mehrmals auf einer Datenbank gespeichert ist.

Das mehrmalige Speichern von Daten (Redundanz) führt sowohl zum fehlerhaften Umgang mit diesen Daten als auch zu einem unnötigen Verbrauch von Speicherkapazität.

Objekten sind beliebig viele Felder zuordenbar. Dabei versteht man unter *Feldern* Eigenschaften oder Attribute des jeweiligen Objektes. Das Verhalten zum Beispiel des Objektes *Haus* kann definiert sein durch die Feldtypen:

- physikalisches Feld                      z.Bsp. *name*
- geometrisches Feld                      z.Bsp. *lage*

Ein typisches Objekt besteht aus einem oder mehreren physikalischen Feldern, die alphanumerische Daten wie Name, Länge, Kapazität, Beschreibung usw. enthalten und einem oder mehreren geometrischen Feldern für eine Geometrie oder die Position eines Objektes.

Objekte, die im CASE-Tool erzeugt werden, sind Tabellen. Einträge in eine derartige Tabelle nennt man *Record*. Records werden im Standardeditor, der auf jedem Objekt definiert ist, automatisch erstellt. Records lassen sich aber auch in der Programmierumgebung eingeben. Dies geschieht über sogenannte *Templates*. Templates kann man als vorläufigen Record ansehen, vor dessen entgültigem Eintrag in die Tabelle dessen Felder gefüllt werden müssen. Wie ein Template erstellt werden kann, erklärt ein Beispiel im Anhang.

Jeder Record (Eintrag) in der Tabelle muss von einem anderen Record in derselben Tabelle unterschieden werden. Dies geschieht über sogenannte Schlüsselfelder oder kurz Schlüssel. Schlüssel sind eindeutige Bezeichner für einen Eintrag in eine Tabelle. In vielen Fällen ist ein physikalisches Feld mit der Bezeichnung *Name* als Schlüssel sehr geeignet. Beispielsweise kann der Name eines Hauses *Rathaus* oder *Post* sein, was eine eindeutige Zuordnung zulässt. Im Falle keiner eindeutigen Zuordnung mit nur einem Schlüssel, können auch zwei oder mehrere Schlüsselfelder verwendet werden.

Beziehungen zwischen Objekten heißen *Verknüpfungen*. Verknüpfungen sind wichtig, um eine Beziehung zwischen zwei verschiedenen Objekten zu charakterisieren. Diese Idee heißt *Entity-Relationship-Modell*. Dabei sind Entities die Objekte, zwischen denen eine Verknüpfung erstellt werden soll während Relationships die Art der Beziehung zwischen Objekten darstellt. Alle möglichen Verknüpfungen zeigt Tabelle 4.1.

*Tabelle 4.1: Beispiele zum Entity-Relationship-Modell*

Beziehung	Beispiel
0 : 0	0 oder 1 Flüsse fließen durch 0 oder 1 Städte
1 : 0	1 Frau hat 0 oder 1 Ehemann
1 : 1	1 Land hat 1 Hauptstadt
0 : n	0 oder 1 Zug fährt durch 1 oder mehrere Bahnhöfe
1 : n	1 Diplomand arbeitet an 1 oder mehreren Rechnern
m : n	1 oder mehrere Planeten werden von 1 oder mehreren Astronomen beobachtet

Das Entity-Relationship-Modell wird praktiziert, um Inkonsistenzen zu vermeiden. Liegt eine 1 : 1 - Beziehung zugrunde, ist die Erstellung eines zweiten Objektes *Hauptstadt* und der 1 : 1 - Verknüpfung nicht erforderlich. Es kann auch die Spalte *Hauptstadt* im Objekt *Land* geben. Bei 0/1/m : 0/1/n - Verknüpfungen erstellt SMALLWORLD sogenannte *intermediate tables*. Das sind interne Tabellen, die zwei Spalten enthalten: die eine zeigt auf die erste Tabelle und die andere zeigt auf die zweite Tabelle in der Verknüpfung.

Was mit dem CASE-Tool auf einer graphischen Benutzeroberfläche recht benutzerfreundlich erzeugt wird, kann auch programmiert werden. Es ist aber einfacher, ein Datenmodell zu erstellen als den source code dafür zu schreiben. So besehen hat sich die Verwendung des CASE-Tools als Oberfläche für Entwickler bewährt. Nichtsdestotrotz muss für jedes im CASE-Tool erstellte Objekt bereits eine Exemplardefinition existieren. Das heißt, möchte man ein Objekt im CASE-Tool einfügen, muss man davor dieses Objekt als Exemplar definiert haben und in den EMACS laden. Dann kann man die Tabelle im CASE-Tool einfügen und komplexen Funktionen zuordnen. Im CASE Objekt Editor unter *Eigenschaften* findet sich die Option *Speicher-Objektklasse festlegen*. Hier muss der Name des bereits definierten Exemplars angegeben werden. Ferner findet sich bei den *Eigenschaften* die Möglichkeit einen Editor für das Objekt festzulegen. Dies ist der Editor, der bei Doppelklick auf das Objekt im Objektklassenmenü aktiviert wird.

Ein im CASE-Tool erstelltes Netz von Objekten nennt man Datenmodell. Ein Modell erfaßt immer nur die innere Struktur eines Systems. Somit ist auch das im CASE-Tool konstruierte Datenmodell eine reine Metainformation, es enthält noch keinerlei Daten! Möchte man bereits bestehende Datenmodelle ins CASE-Tool importieren, ist das über ein *Reverse Engineering* (Zurückentwickeln) möglich. (GE Smallworld, 1997b)

## 4.5 Die Architektur der Geo - Datenbank

Ein Hauptgedanke einer Datenbank ist es gewonnene Daten strukturiert zu verwalten, um so weiterverarbeitet werden zu können. Speicherinstanz von verwalteten Daten ist somit eine Datenbank. Die Datenbank in SMALLWORLD enthält die Beziehungen des Datenmodells zu den Anwendungsdaten, die Art, in der die Daten strukturiert werden und die Mechanismen, mit denen von den Benutzern auf die Daten zugegriffen werden kann.

Man unterscheidet bei den Datenbeständen, ob sie Daten aus der CASE-Partition oder aus der GIS-Partition enthalten. Den Umgang mit Datenbeständen kontrolliert der `manage_soc`. Ein soc ist ein Spatial Object Controller. Der `manage_soc` verwaltet alle Datenbestände.

Die komfortable Datenbank-Umgebung von SMALLWORLD gestattet es, dass mehrere Benutzer Daten einer Datenbank gleichzeitig bearbeiten können. Sinn dieser sogenannten Versionsverwaltung unter SMALLWORLD ist es, dass jeder Benutzer in der Lage ist, sich eine eigene Version anzulegen. Die objektrelationalen, versionsverwalteten Datenbanken des SMALLWORLD GIS heißt VMDS (Version Managed Data Store). Beim Erstellen einer sogenannten Alternative wird eine Kopie des Hauptdatensatzes (Hauptalternative) erzeugt. Alternativen werden erstellt, damit verschiedene Bearbeiter zur gleichen Zeit Änderungen des Hauptdatensatzes vornehmen können. Man kann beliebig viele Unteralternativen einer Alternative erstellen. Auf diese Weise entsteht eine umgekehrt-baumartige Hierarchie. In

dieser Struktur ist es möglich, eine in einer Alternative durchgeführte Änderung auf die nächsthöhere Alternative zu übertragen, zu *senden*. Änderungen, die in der nächsthöheren Alternative gemacht wurden, können *geholt* werden, das entspricht der Übertragung der Änderung auf die nächstuntere Alternative. Dieser elegante hierarchische Aufbau von Datensatz-Kopien zeichnet SMALLWORLD als leistungsstarkes GIS aus. Abbildung 4.5 illustriert diese Struktur bildlich. (GE Smallworld, 1999b)

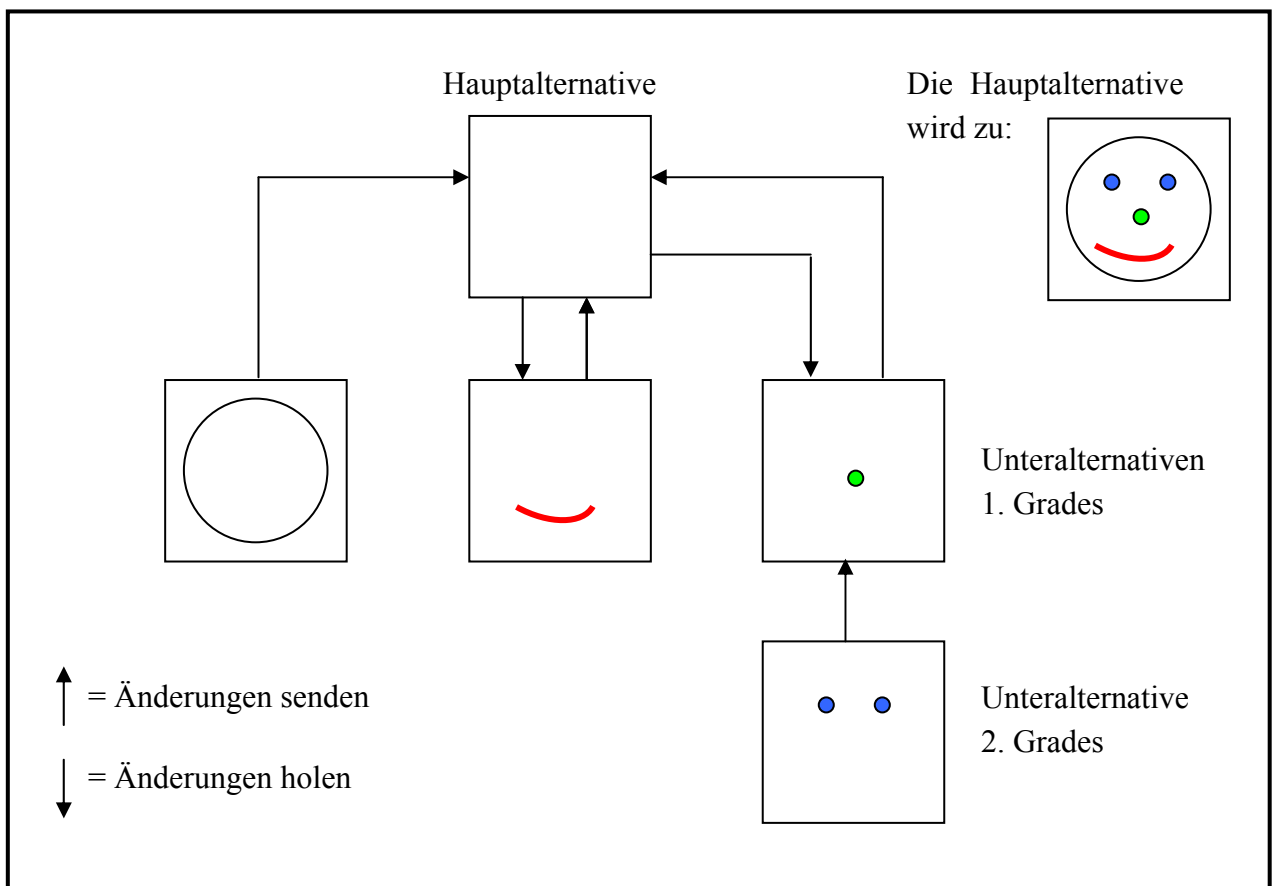


Abbildung 4.5: Alternativenhierarchie unter SMALLWORLD GIS (verändert nach GE Smallworld, 1999b)

Eine Alternative ist somit als Glasplatte zu verstehen, in der ein Datenmodell definiert und durch die auf Datenbestände anderer Alternativen geschaut werden kann.

Die Datenbanken in SMALLWORLD GIS enthalten die im CASE-Tool erstellten Tabellen (beschrieben in Kapitel 4.4: Das CASE - Tool). Tabellen kann man als Relationen auffassen. Somit sind die Datenbanken unter SMALLWORLD GIS relationale Datenbanken. Alle Daten sind in Relationen, also in Tabellen gespeichert. In den Datenbankpartitionen von SMALLWORLD GIS werden folgende Informationen gespeichert:

- Objekte, die im CASE-Tool definiert werden. Struktur des Datenmodells.
- Benutzerkonfiguration

- Tabellenzugriffsrechte
- Topologie

(GE Smallworld, 1999b)

## 4.6 Module und Fachschalen

Module oder auch Teilinformationssysteme bilden in einem GIS die Basis für weitere Entwicklungen und Anwendungen. Es sind weiterentwickelte Fachschalen, die auf einer Fachschale im engeren Sinne basieren. Module sind eigenständige Funktionalitäten, die gegebenenfalls auf andere Modulen angewiesen sind. Ist letzteres der Fall, wird bei der Definition eines Moduls dessen Abhängigkeit von anderen Modulen angegeben, was bei der Installation dazu führt, dass alle notwendigen Module automatisch auch geladen werden. Auf diese Weise wird Eigenständigkeit der einzelnen Module und einwandfreies Funktionieren parallel gewährleistet.

Bei Fachschalen handelt es sich um GIS-Module, welche angepasst an die jeweiligen Anforderungen Datenmodelle und entsprechende Bearbeitungsfunktionen zur Verfügung stellen. Dem Benutzer ist hier also die Überlegung zur optimalen Datenmodellierung abgenommen. Allerdings passen diese Modelle nicht immer und auf jeden Anwender, so dass eine Flexibilität zur Anpassung der Modelle gefordert werden muss.

Der entscheidende Punkt von Fachschalen sind also die vorbereiteten Datenmodelle. Während die Kernsysteme eine Grundfunktionalität zur Erfassung und Auswertung von Linien, Flächen, Punkten, Annotation (freie Graphik und Textelemente) und Topologie bereitstellen, sind in Fachschalen zusätzlich vorbereitete Datenstrukturen für die zu erfassenden Sachdaten und Beziehungen enthalten. Erfassung und Auswertung wird zumeist ebenfalls durch menügesteuerte Funktionen unterstützt. Im Gegensatz zu den Kernsystemen werden von den Fachschalen fast immer angepaßte Benutzeroberflächen zur Verfügung gestellt.

Zur individuellen Anpassung von Fachschalen werden Werkzeuge benötigt, die neben den Datenmodellen auch die zugehörigen Arbeitsabläufe für Erfassung, Verwaltung etc. anpassen können. CASE-Tools können in der Regel lediglich die Datenmodelle anpassen, für den gesamten Anpassungsprozess werden daher spezielle Konfigurations- oder Makrosprachen benötigt. In diesem Bereich kommen zunehmend objektorientierte Sprachen zum Einsatz, welche sich zwar an Standards wie C++ oder SMALTALK anlehnen, diese aber um raumbezogene Funktionen erweitern (z.B. AVENUE von ESRI, AMBOSS von ADASYS, MAGIK von SMALLWORLD). (Schilcher, 1996)

Im Zuge dieser Diplomarbeit ist das Modul MODINT entstanden. Jedes Modul wird definiert durch eine sogenannte *module.def* Datei, in der u.a. der Modulname, eine Beschreibung sowie eine Liste der benötigten Module steht. Beim Laden dieses einen Moduls ins GIS werden alle erforderlichen Module (unter *requires*) gleichzeitig auch geladen. Für MODINT sieht die Moduldefinition wie folgt aus:

```

modint 1

description
    Integration von FEFLOW, zentrale Schnittstelle fuer alle Tools
end

requires
    ogrut 1
    ihf_vistool 1
    bomet 1
end

```

## 4.7 Fazit

Im Rahmen des WAQIS-Projektes wurden bislang eine Reihe von Auswertewerkzeugen und Tools bereitgestellt, mit Hilfe derer Boden und Aquifer im Sinne der Trinkwassersicherung beschreibbar sind. Verwendet wurde das GIS Smallworld der Firma GE Smallworld.

In dieser Arbeit soll eine Oberfläche geschaffen werden, die die bestehenden Werkzeuge zentral vorhält. Die Bewegung des Wassers in der gesättigten und ungesättigten Bodenzone wird im WAQIS durch das Grundwassersimulationsprogramm FEFLOW (Finite Element Subsurface FLOW System) berechnet. Als weitere Aufgabe ist somit die Integration von FEFLOW in SMALLWORLD GIS derart zu realisieren, dass die Werte der Modellparameter im GIS flächenhaft dargestellt werden können. Wegen seiner leistungsstarken Datenbankstruktur eignet sich SMALLWORLD GIS besonders gut für dieses Problem. Schließlich liefert FEFLOW große Datenmengen, die in geeigneten Datenbanken verwaltet und zu weiterer Verarbeitung gespeichert werden müssen. Geschaffen werden diese Datenstrukturen im CASE-Tool zur Vermeidung von Fehlern, zur Verhinderung von Redundanz und zur Übersichtlichkeit über das Projekt. Entwickelt wird in sogenannten Alternativen, das sind Kopien des Datenmodells in der Hauptalternative des CASE-Tools in diese Unteralternative. Diese Versionsverwaltung von SMALLWORLD GIS erlaubt es mehreren Programmierern gleichzeitig zu entwickeln.

Als Programmiersprache steht die objektorientierte Sprache MAGIK zur Verfügung. *Obektorientiert* heißt, dass Methoden bestimmten Objektklassen zugeordnet werden. Wird eine Instanz einer Klasse erzeugt, versteht diese dann die auf ihrer Klasse definierten Methoden. Wird eine Kindobjektklasse erstellt, versteht die Instanz der Unterklasse die Methoden der Elternobjektklasse auch. Diese für das objektorientierte Programmieren kennzeichnende Eigenschaft nennt man Vererbung. Programmiert wird in der Umgebung EMACS, einem Editor zur Bearbeiten von Textdateien. MAGIK ist eine der Entwicklersprachen, in denen Prozesse in Threads ablaufen können. Threads sind eine Art separate Prozessoren, in denen Vorgänge unabhängig vom Hauptprogramm gestartet werden

können. Die Integration von FEFLOW ist mit der Durchführung zeit- und speicheraufwendiger Prozesse verbunden, weswegen das Aufrufen von Threads unumgänglich ist.



## 5 Der Simulator FEFLOW

### 5.1 Einleitung

FEFLOW ist ein interaktives, voll menügesteuertes und graphikorientiertes 2D- und 3D-Grundwassermodellierungssystem zur Simulation von Strömungs-, Stoff- und Wärmetransportprozessen in gesättigten und ungesättigten porösen Medien. Es wurde 1979 von H.-J. Diersch (WASY GmbH) entwickelt. Verwendet wird die Methode Finites Elemente, das zugrunde liegende Netz besteht somit aus Tetraedern. Dieses Konzept erlaubt die einfache Verdichtung des Netzes zur Erhöhung der numerischen Stabilität an Stellen, an denen der Potenzialgradient groß ist, was insbesondere an Brunnen (als blaues Symbol in Abbildung 5.1) der Fall ist. FEFLOW beinhaltet Netzgeneratoren, die komplexe Strukturen mit Hilfe von Superelementen effektiv diskretisieren. Bei der Erzeugung der Diskretisierung wird der Anwender durch hinterlegte Karten seines Berechnungsgebietes auf der Basis von Gauss-Krüger-Koordinaten unterstützt. Graphisch orientierte Editoren ermöglichen die Zuordnung von Parametern als Anfangs- und Randbedingungen zu Knoten und Elementen. Dabei können interaktiv mit der Maus am Bildschirm einzelne Knoten oder Knotengruppen sowie Elemente oder Elementgruppen ausgewählt werden. Die erforderlichen Parameter lassen sich dem FE-Gitter auch über Dateien zuordnen.

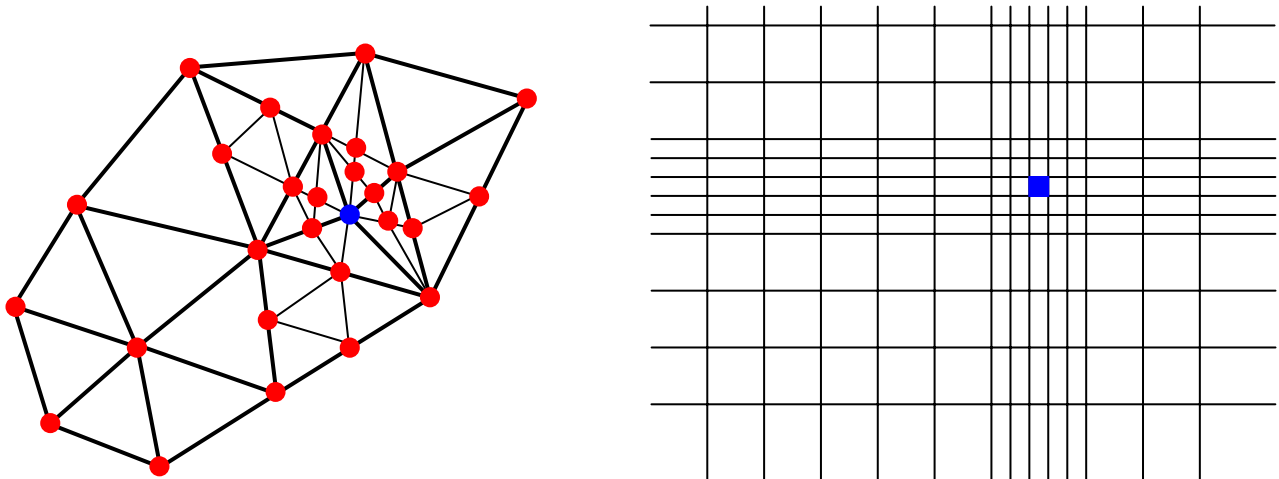


Abbildung 5.1: Finite Elemente Netz (FEFLOW) und Finite Differenzen Netz (MODFLOW)

Gegenüber dem Netz aus Tetraedern erstellen Modelle, die die Methode Finites Differenzen verwenden, ein Netz aus Quadern. MODFLOW ist ein Beispiel dafür. MODFLOW ist ein zweidimensionales Grundwassermodell, das seit 1984 existiert. Es hat einen modularen Aufbau, der einfache Änderungen erlaubt, sodass der Code an eine bestimmte Anwendung angepasst werden kann (McDonald und Harbaugh, 1988). Ein Finites

Differenzen Netzes kann nicht in der gleichen Einfachheit verdichtet werden wie ein Finites Elemente Netz, da die lateralen Bereiche des Netzes um den Brunnen auch verdichtet werden, obwohl dort keine Verdichtung erforderlich ist.

Bemerkenswerterweise gibt es seit 1995 den Simulator FRAC3DVS, in welchem der Modellierer die Option hat auf **beide** Methoden (FE und FD) zuzugreifen. Es wurde von R. Therrien und E.A. Sudicky in der Waterloo Group in Canada entwickelt. FRAC3DVS ist ein numerisches Grundwasserströmungs- und Stofftransportmodell zur Modellierung der gesättigten und ungesättigten Zone. Geschaffen wurde dieses Modell insbesondere zur Modellierung des Kluftaquifers. Es kann als bislang einziges Modell die stoffliche und hydraulische Interaktion zwischen Matrix und beliebig vielen (horizontalen oder vertikalen) Klüften nachbilden. Die poröse, gering durchlässige Matrix ist repräsentiert in drei Dimensionen und Klüfte hoher Durchlässigkeit werden durch zweidimensionale Ebenen wiedergegeben. Bei der Lösung wird advektiver Fluss und Stofftransport in Kluft und Matrix, molekulare Diffusion entlang des Konzentrationsgradienten, mechanische Dispersion und radioaktiver Zerfall berücksichtigt. Für weiter Information sei auf Therrien und Sudicky (1995) verwiesen.

Schließlich existiert neben der FE- und FD-Methode, der Algorithmus Finites Volumina (FV). Dessen bedient sich das aus den U.S.A. stammende Modell SUTRA, das zur Simulation von Dichteströmungen verwendet wird. Simmons und Narayan (1998) haben dieses Modell intensiv bei australischen Salzseen angewandt.

## 5.2 Relevante Dateitypen in FEFLOW

Ziel von MODINT ist es, eine Schnittstelle von SMALLWORLD GIS zu FEFLOW zu schaffen. Dabei sollen die Ergebnisse von FEFLOW als Dateien eingelesen und im GIS zur Visualisierung verfügbar gemacht werden. Die Kommunikation zwischen dem Modell und GIS kann nur über FEFLOW-Outputdateien stattfinden. So besehen stellt sich die Frage nach den Dateitypen, die zur Modellintegration von FEFLOW in SMALLWORLD GIS von Bedeutung sind.

In FEFLOW existieren einige Dateien, in denen die für Preprocessing, Processing und Postprocessing erforderlichen Daten stehen. Maßgeblich für die Funktionalität von MODINT sind die Dateien \*.fem und \*.dac, gegebenenfalls auch \*.dar. Die ersten beiden sind variabel im binären oder ASCII - Format in FEFLOW speicherbar.

Die \*.fem - Datei beinhaltet die komplette Beschreibung des Strömungs- und Stofftransportproblems. Sie wird beim Preprocessing von FEFLOW erzeugt. In dieser Datei stehen die Werte aller Material- und Stofftransportparameter, die der Prozessor beim Ausführen benötigt sowie die Elemente, denen die Parameter zugewiesen sind. Darüber hinaus enthält sie alle Informationen über die Potenzialränder und Randbedingungen.

Die \*.dac - Datei wird beim Processing erzeugt. Sie kann als Ergänzung der \*.fem - Datei angesehen werden. In ihr stehen neben der gesamten Problemdefinition auch alle Modellergebnisse, also die hydraulischen Potenziale auf allen Knoten zu jedem Zeitschritt. \*.dac-Dateien werden in der Regel sehr groß. Die im Zusammenhang mit dieser Arbeit

untersuchte Datei umfasst die Ergebnisse von 12 Zeitschritten und hat bereits über 600.000 Zeilen.

Die \*.dar - Datei wird benötigt, sofern detaillierte numerische Werte an Beobachtungspunkten (observation points) zur Kalibrierung herangezogen werden sollen. (WASY, 1998)

### **5.3 In FEFLOW verwendete Parameter**

In FEFLOW können alle Parameter, die zur Beschreibung eines gesättigten oder ungesättigten Fließproblems erforderlich sind, graphisch editiert werden. Die Werte können global oder lokal zugewiesen werden. Ein globaler Parameter nimmt im gesamten Untersuchungsgebiet denselben Wert an während die Werte lokaler Parameter einzelnen Bereichen zugeordnet sind.

Die räumliche Verteilung der Stofftransport- und Materialparameter für 3D Lösungen ist in dieser Arbeit von besonderem Interesse. Im GIS soll diese Verteilung dargestellt werden. Im Anhang befinden sich Tabellen über die betrachteten Modellkenngrößen (vgl. Tabelle A.2 und Tabelle A.3 im Anhang).

### **5.4 Fazit**

FEFLOW ist das im WAQIS-Projekt angewendete Grundwassermodellierungssystem. Es bedient sich der Methode Finiter Elemente, neben der es auch die Algorithmen Finite Differenzen (z.Bsp. in MODFLOW) und Finite Volumina (z.Bsp. in SUTRA) gibt. In FEFLOW wird die Outputdatei vom Typ \*.fem beim Preprocessing erzeugt und enthält alle Modelleinstellungen. Beim Processing wird die Datei \*.dac generiert, in der außer den Modelleinstellungen die Modellergebnisse stehen.

Die in FEFLOW zur Modelllaufzeit erstellten Dateien sollen verwendet werden, um die Schnittstelle zwischen SMALLWORLD GIS und FEFLOW zu verwirklichen. Damit eröffnen sich zwei verschiedene Wege zur Integration des Modells. Zu unterschiedlichen Zeitpunkten wird auf eine unterschiedliche Fülle von Information zugegriffen.

## 6 Das Modul MODINT zur Modellintegration

### 6.1 Vorgehensweise und Umsetzung

Eine Schnittstelle ist der Anschluß einer externen Funktionalität an ein System, auf dem gearbeitet wird. Dabei umfasst die Steuerung der Funktionalität das Zuweisen des erforderlichen Inputs, Kommunikation während des Prozesses sowie die Rücknahme des Outputs. Im vorliegenden Fall ist die Funktionalität der Simulator FEFLOW, der in das System SMALLWORLD GIS integriert wurde.

Das Zuweisen des Inputs, bzw. die Parametrisierung von FEFLOW geschieht über das bereits existierende Modul OGRUT. Zur Laufzeit des Modells kann der Simulator über den *Interface Manager* (IFM) Daten mit anderen Programmen über externe Dateien auszutauschen. Zur Vervollständigung war daher Ziel dieser Arbeit auch die Rücknahme des Outputs, also der Ergebnisse umzusetzen, um damit den integrativen Gedanken der interface zu verwirklichen. Dazu wurden Algorithmen entwickelt, die die Ergebnisdatei (\*.dac) des Modells einliest.

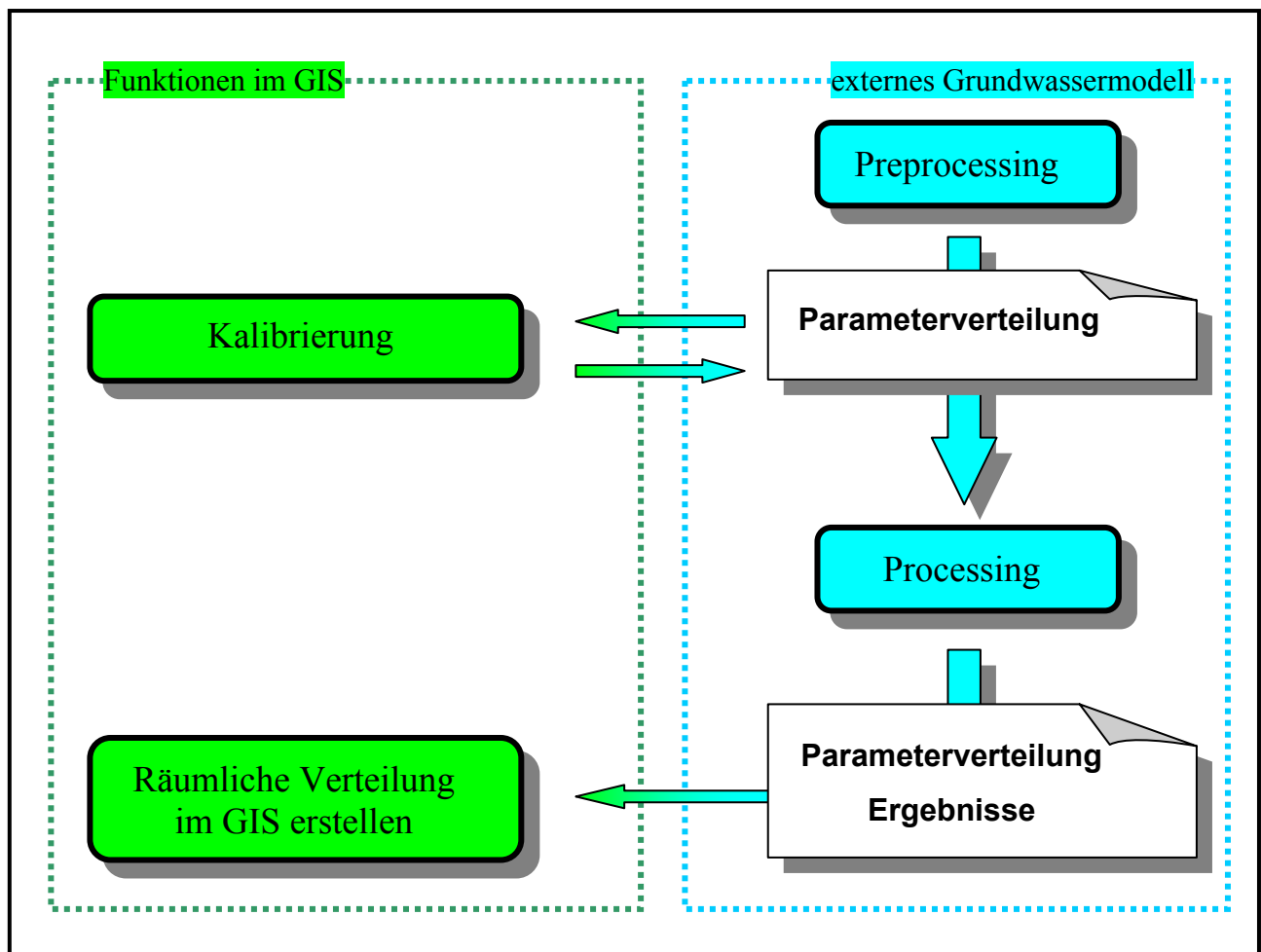


Abbildung 6.1: Wirkungsweise der Schnittstelle (←→) an den Simulator FEFLOW

Bisher kann bei der Integration von FEFLOW in SMALLWORLD GIS zu verschiedenem Prozessfortschritt das Modell gesteuert werden. Mit dem Modell wird bereits nach dem Preprocessing, also in der Phase vor der eigentlichen Simulation kommuniziert. Hier wird auf die FEFLOW-Steuerdatei (\*.fem) zugegriffen. Bisher können die Werte globaler Parameter in dieser Datei verändert werden. Damit ergibt sich in beschränktem Maß (da nur Variierbarkeit globaler Parameterwerte) eine Zugriffsmöglichkeit auf FEFLOW vor der Simulation. Dieser erste Teil der Schnittstelle stellt somit die Option zur Anbindung an Kalibriermechanismen, wie etwa die Auto-Kalibrierung dar.

Der zweite Teil der Schnittstelle ist repräsentiert durch das Einlesen der Ergebnisdatei (\*.dac) nach dem Processing. Die darin enthaltene Verteilung der Modellparameter kann analysiert und dargestellt werden, jedoch nicht die Verteilung der Ergebnisse selber. Die Analyse der Verteilung der Ergebnisse kann hier auch stattfinden, wurde aber im Rahmen dieser Arbeit noch nicht realisiert. Mit der Kenntnis über die flächenhafte Verteilung und mit dem Finiten Elemente Netz als Abbildungsbasis kann ein räumliches Abbild der Parameterverteilung im GIS erzeugt werden. Zu welchen Prozessfortschritten die Schnittstelle ansetzt zeigt Abbildung 6.1.

Die bereits existierenden Module von SMALLWORLD GIS sind somit in der Lage die Art und Weise der flächenhaften Anordnung von Parameterwerten des Modells in eine flächenhafte Anordnung im GIS zu überführen.

## 6.2 Umbau und Integration von OGRUT

Ein wesentlicher Punkt dieser Arbeit war das Aufarbeiten der OGRUT-Sourcen, um sie in MODINT einbauen zu können. OGRUT ist ein Oberflächen- und Grundwasser-Teilinformationssystem für SMALLWORLD GIS. Es liefert das Finite Elemente Netz aus FEFLOW für MODINT. Das enge Zusammenspiel zwischen OGRUT und MODINT war der Anlass dafür, dass die Funktionalitäten beider Module in einer gemeinsamen Datenbank und auf einer zentralen Oberfläche vereinigt wurden. Diese sind unter dem Menüpunkt *OGRUT* und *MODINT* von der Oberfläche aus verfügbar (vgl. Abbildung 6.3).

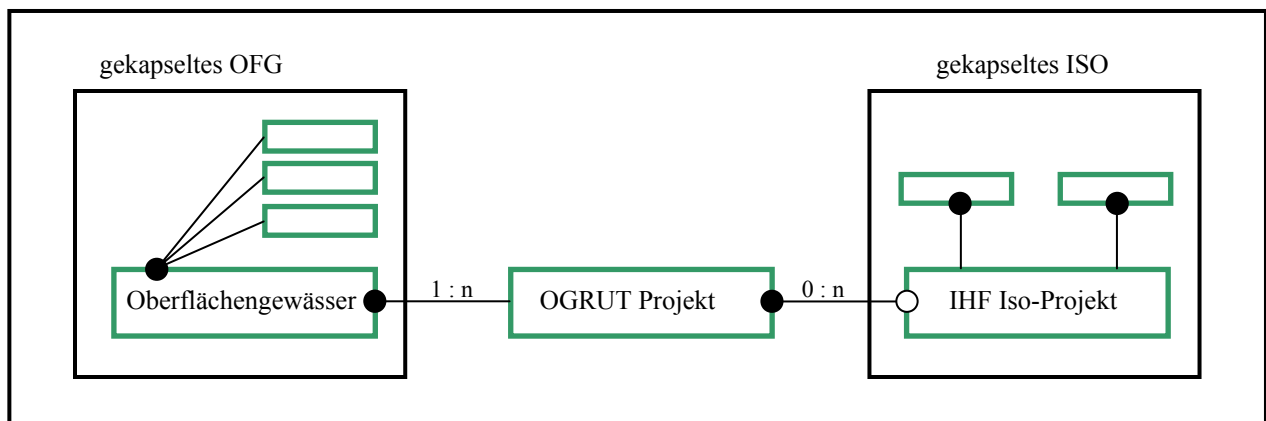


Abbildung 6.2: Struktur der gekapselten Module in der Entwicklerdatenbank

Zunächst wurde im CASE-Tool das Datenmodell von OGRUT in eine separate Datenbank überführt, auf der entwickelt wurde. OGRUT greift auf das OFG- und ISO- Modul von LIWIS zu, die Datenmodelle dieser Module wurden auch auf die Entwicklungsdatenbank übertragen. Dadurch erfolgte eine Kapselung von OFG, das ISO- Modul wurde bereits im Zuge einer anderen Diplomarbeit (Kiefer, 2001) gekapselt. Durch diese Kapselung ist eine Anpassung des Datenmodells in der neuen Entwicklerdatenbank möglich (vgl. Abbildung 6.2).

In einem weiteren Schritt wurden alle Sourcen von OGRUT so organisiert, dass die Funktionalitäten von der MODINT-Oberfläche aus angewählt werden können. Dabei mussten die Methoden (vgl. Kapitel 4.3.4) auf die neuen Objekte des OFG- und des ISO- Moduls umgeschrieben werden.

### 6.3 Die "neue" Oberfläche von MODINT

MODINT ist repräsentiert durch eine benutzerfreundliche Oberfläche, in der verschiedene bereits existierende Auswertewerkzeuge aus dem WAQIS-Projekt bereitgestellt werden. Insbesondere sind die Funktionalitäten des Moduls OGRUT (Herrmann, 2000) von diesem Menü aus erreichbar. Ferner besteht auf dem zentralen Editor (Abbildung 6.3) die Möglichkeit FEFLOW-Dateien einzulesen bzw. zu manipulieren, was die Schnittstelle zum Grundwassermodell FEFLOW repräsentiert.

Das zentrale Steuermenü von MODINT ist so gestaltet, dass es in beliebiger Form erweiterbar ist. Die Comic-Symbole repräsentieren den Urheber und verweisen auf MODINT- (Lucky Luke) bzw. OGRUT- (Geier) Sourcen.

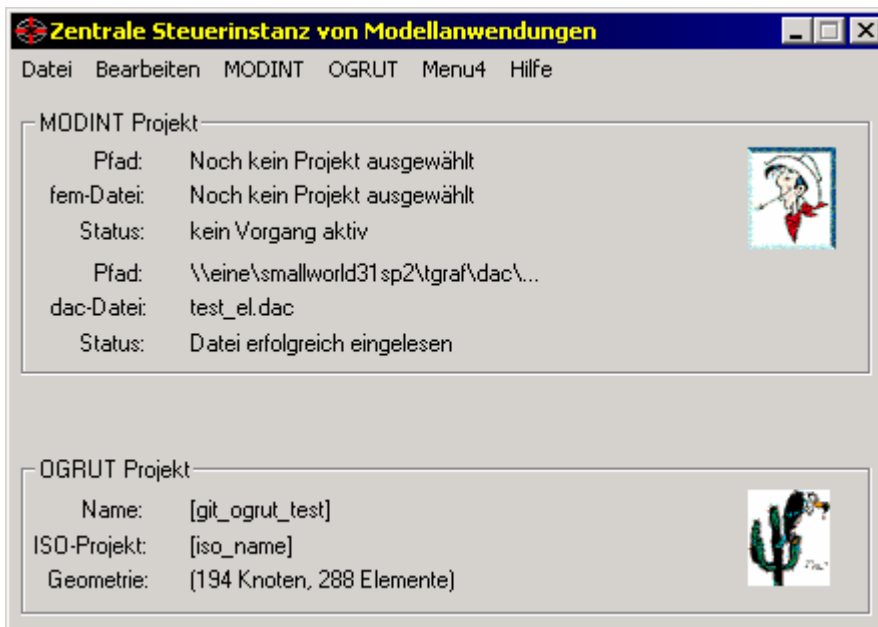


Abbildung 6.3: Der zentrale Editor von MODINT

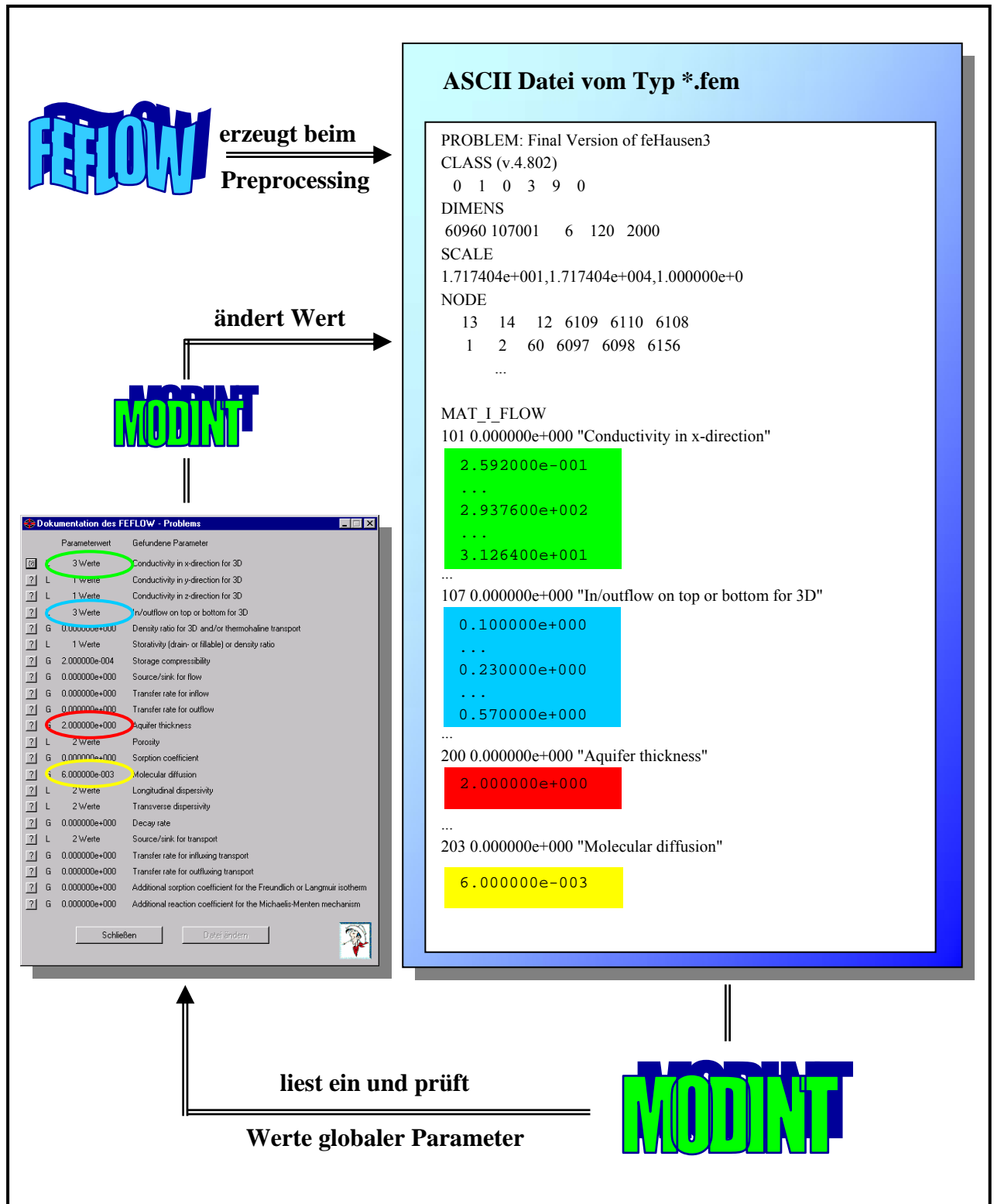


Abbildung 6.4: Modifikation einer in FEFLOW generierten \*.fem-Datei durch MODINT

## 6.4 Geschriebene Parser

Damit die Informationen, die in den FEFLOW-Dateien stehen, erkannt werden und weiterverarbeitet werden können, müssen für die Dateitypen \*.fem und \*.dac Parser geschrieben werden, der diese Prüfung vornimmt. Ein Parser ist ein Analysewerkzeug der Syntax einer Sprache. Solche Sprachanalysatoren werden u.a. als Programmteile von Compilern oder Übersetzern eingesetzt. Er führt eine syntaktische Prüfung eines in einer höheren oder dateispezifischen Programmiersprache geschriebenen Computerquellprogrammes durch.

Im vorliegenden Fall handelt es sich nicht um eine Programmiersprache sondern um eine gesetzmäßige Abfolge von Zahlen (Parameterwerte, Knoten- und Elementenummern), die die flächenhafte Verteilung der Modellparameter beschreiben.

### 6.4.1 Der \*.fem - Parser

Aus den Angaben im Fenster des Modells FEFLOW erstellt das Preprocessing des Simulators eine \*.fem-Datei. In ihr stehen alle zur Berechnung notwendigen Daten wie Randbedingungen, Zeitschritte und flächenhafte Verteilung der Parameter sowie insbesondere deren Werte. Das entwickelte Werkzeug analysiert die Datei und erkennt die Parameterwerte. Bei globalen Parametern kann der Nutzer deren Wert ändern. Auf diese Weise wird bewerkstelligt, dass \*.fem-Dateien in beschränktem Umfang manipuliert und ins Processing des Simulators FEFLOW als Input eingehen. Somit hat der Anwender auf die Modelleinstellungen zu einem Zeitpunkt vor der Simulation direkten Zugriff. Abbildung 6.4 illustriert die Funktionsweise.

### 6.4.2 Der \*.dac - Parser

In der \*.dac-Datei stehen sowohl die flächenhaften Verteilungen aller Parameterwerte als auch die Ergebnisse der Modellrechnung zu allen Zeitpunkten. Bei dieser Datei ist die flächenhafte Verteilung der Parameterwerte von Interesse. Zur Darstellung der Ergebnisse (hydraulische Potenziale) und der Parameterverteilungen muss die Datei analysiert und eingelesen werden<sup>5</sup>. Zur korrekten Zuweisung der Geometrie wird ein FE-Netz benötigt, welches von OGRUT eingelesen und bereitgestellt werden kann. Dabei werden zwei ASCII-Dateien des Netzes gelesen und die geometrische Lage von Knoten und Elementen verstanden (vgl. Abbildung 6.5).

Da die FEFLOW bürtigen \*.dac-Dateien sehr groß sind, benötigt das GIS zum erfolgreichen Einlesen einen hohen Arbeitsspeicher. Dieser ist als -sds -Zahl beim **Starten** des GIS zuzuweisen. Bei Testdurchläufen hat sich bei \*.dac-Dateien, die um 600.000 Zeilen lang sind, ein -sds von 250.000 kB als erforderlich und ausreichend herausgestellt. Abbildung 6.7 zeigt, wie MODINT die Zuordnung von allen Werten der Parameter zu deren jeweiligen geometrischen Lage versteht und unter Zuhilfenahme des von OGRUT gelieferten Netzes die

<sup>5</sup> Die Erzeugung flächenhafter Verteilungen von FEFLOW-**Ergebnissen** wurde im Rahmen dieser Arbeit noch nicht implementiert.



flächenhafte Zuweisung vornehmen kann, um die Werteverteilung als farblichen Verlauf im GIS darstellt.

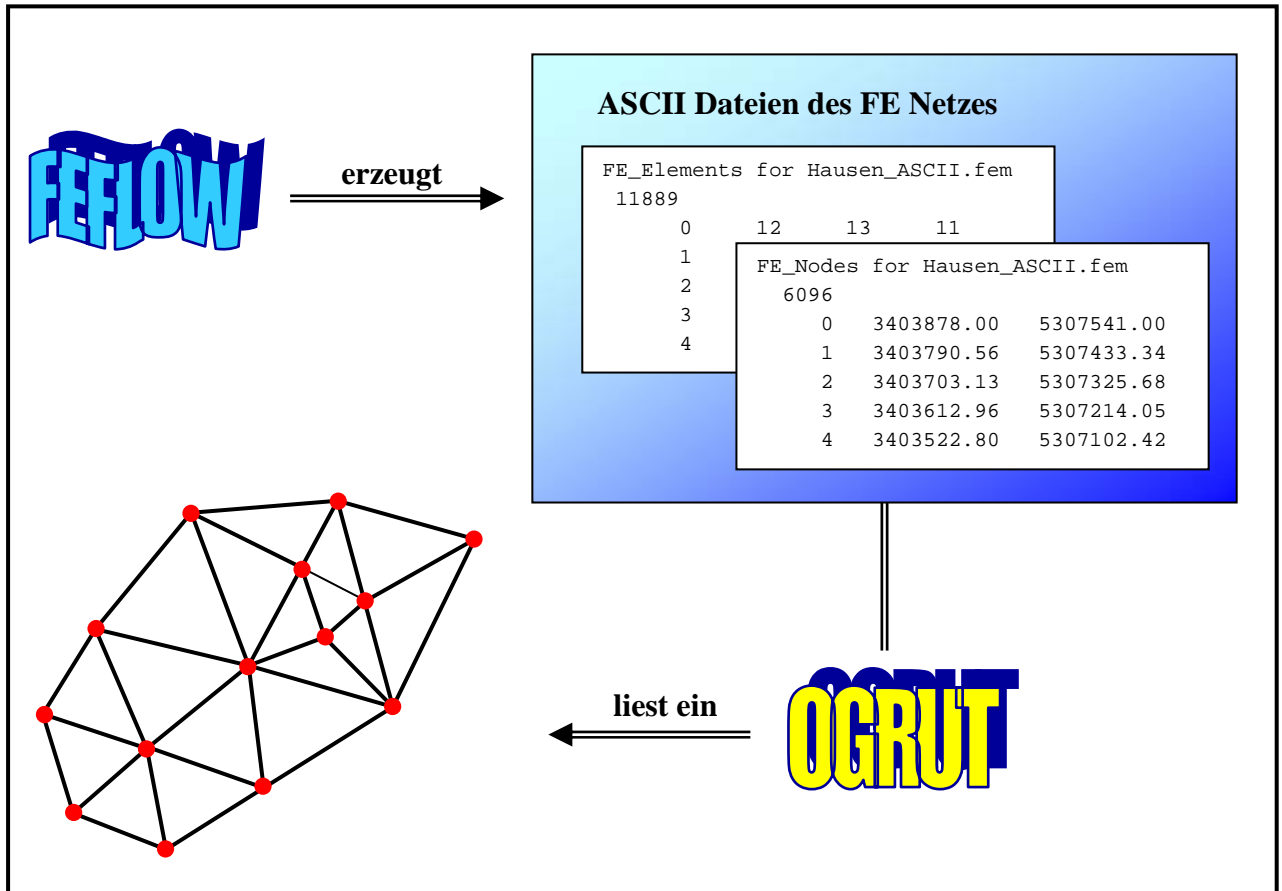


Abbildung 6.5: Einlesen eines FE-Netzes von OGRUT

Die eingelesenen Parameterdaten werden in jedem Fall auf dem Monitor ausgegeben. Wahlweise ist darüber hinaus sowohl das Anlegen einer Ausgabedatei als auch das Schreiben der Daten in die Datenbank ermöglicht. Wie in Abbildung 6.8 dargelegt, wird bei Wählen der letzteren Option das Löschen alter Datenbestände empfohlen. Die folgende Abbildung gibt einen Überblick über die möglichen Ausgabeformen.

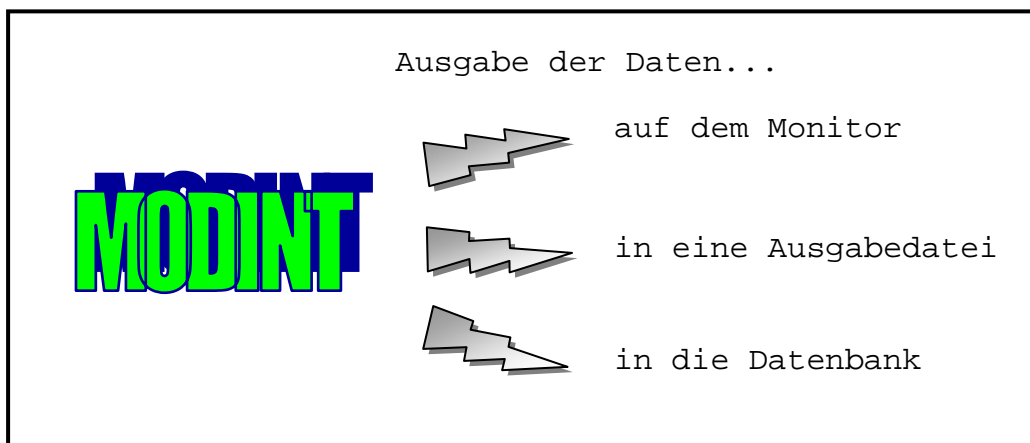


Abbildung 6.6: Ausgabeformen von MODINT beim Einlesen einer \*.dac-Datei

Beim Einlesen wird für jeden gefundenen Parameter eine Variable vom Typ `parameter` erzeugt. Dies ist ein Zwischenobjekt, das erstellt wird, um Daten darauf zu speichern und weiter verarbeitbar zu halten. Ein `parameter` hat die in Tabelle 6.1 ausgestellten Slots (vgl. Kapitel 4.3.1). Die Slots werden beim Prüfen der \*.dac-Datei mit den Informationen der Modellparameter gefüllt. Wenn die Parameterdaten auch auf die Datenbank geschrieben werden sollen, wird auf die Inhalte der Slots von `parameter` zugegriffen.

*Tabelle 6.1: Beschreibung der Slots von `parameter`*

Slot	Beschreibung	Variablentyp
<code>name</code>	Name des Parameters.	<code>string</code>
<code>wert</code>	Alle Werte, die der Parameter im Modellgebiet annimmt.	<code>hash_table</code>
<code>werte_zeile</code>	Alle Zeilen, in denen Werte des Parameters in der Datei stehen.	<code>hash_table</code>
<code>einheit</code>	Einheit des Parameters.	<code>string</code>
<code>global?</code>	<code>_true</code> , wenn es sich um einen globalen Parameter handelt, sonst <code>_false</code> (lokaler Parameter).	<code>boolean</code>
<code>bereicheanzahl</code>	Anzahl der Wertebereiche je Parameter. Bei globalen Parameter eins, sonst gleich der Größe von <code>werte_zeile</code> oder von <code>wert</code> , also gleich <code>werte_zeile.size</code> .	<code>integer</code>
<code>elemente_liste</code>	Liste mit alle Elemente eines Parameters als Nummern. <ul style="list-style-type: none"> <li>Bei globalen Parametern ist diese Liste eindimensional und entspricht der globalen Elementeliste<sup>6</sup></li> <li>Bei lokalen Parametern wird in der Methode <code>lokale_parameter_einlesen</code> eine zweidimensionale <code>hash_table</code> erzeugt. Dann kann auf ein einzelnes Element zugegriffen werden mit: <code>elemente_liste[.bereich<sup>7</sup>][.lauf_el<sup>8</sup>]</code>.</li> </ul>	<code>hash_table</code> 1D oder 2D
<code>zeile</code>	Die Zeile, in der der Name des Parameters steht.	<code>integer</code>

<sup>6</sup> Dies ist einfach eine Liste der Struktur  $\{1, 2, 3, \dots, n-1, n\}$  mit  $n$  = Gesamtzahl Elemente im Modellgebiet

<sup>7</sup> Laufvariable über die Bereiche

<sup>8</sup> Laufvariable über die Elemente je Bereich

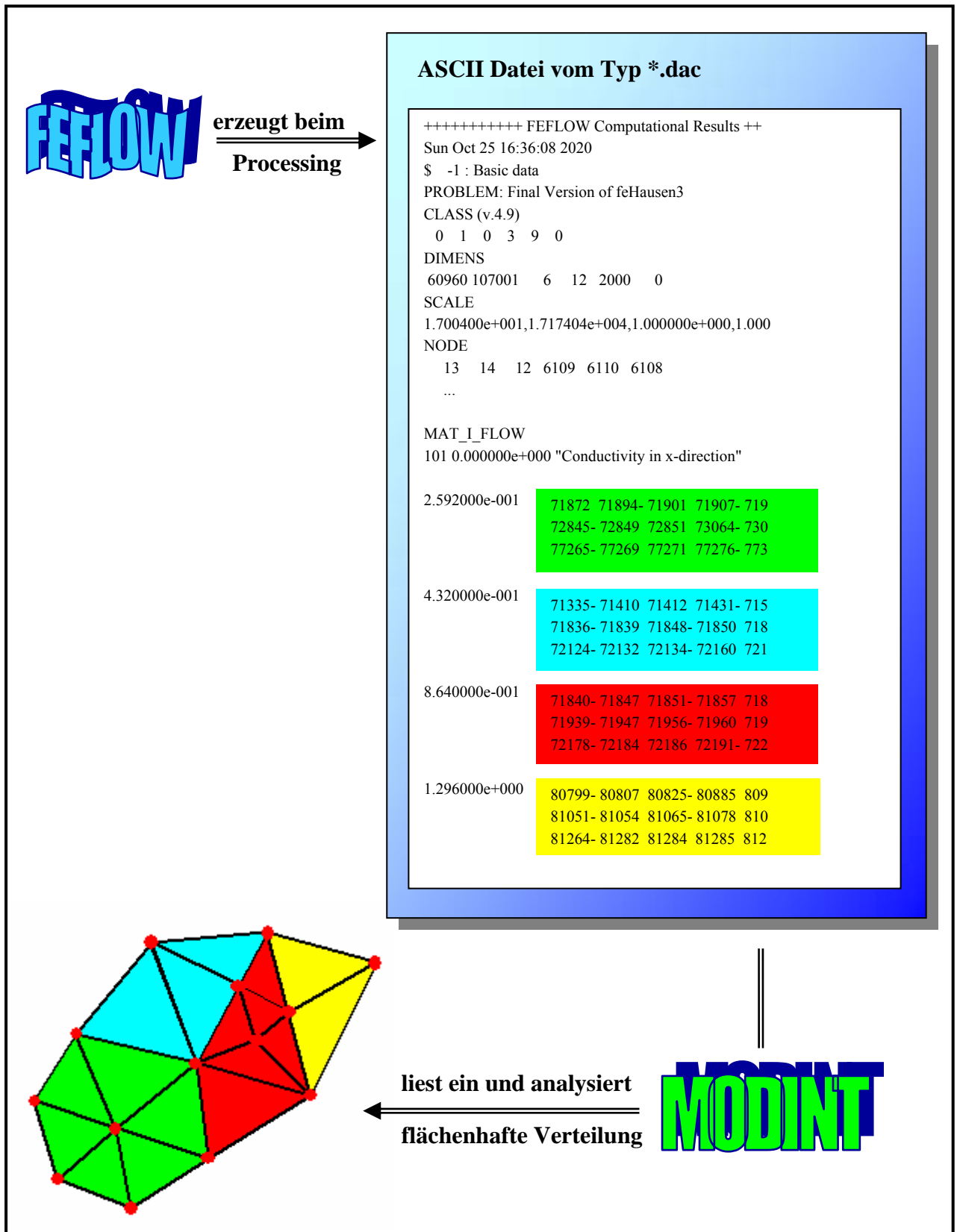


Abbildung 6.7: Funktionsweise zur Erzeugung flächenhafter Parameterverteilungen in MODINT

## 6.5 Die Output - Datei \*.eco

Beim Einlesen der \*.dac-Datei erhält der Anwender die Möglichkeit eine Outputdatei vom Typ \*.eco zu erzeugen (vgl. Abbildung 6.8). Vor dem Einlesevorgang der \*.dac-Datei erscheint ein Menü, in dem der Pfad der erstellten \*.eco-Datei angegeben werden kann. Als Dateiname wird der Stamm der einzulesenden \*.dac-Datei verwendet und ein 'o' für Output zu dem Stamm hinzugefügt. Das Schreiben der Parameterdaten in Slots (vgl. Tabelle 6.1 und Kapitel 6.4.2) erlaubt das einfache Zugreifen auf die Daten und ein einfaches Anlegen der \*.eco-Datei. Die Struktur dieser Outputdatei ist in Abbildung 6.9 wiedergegeben.

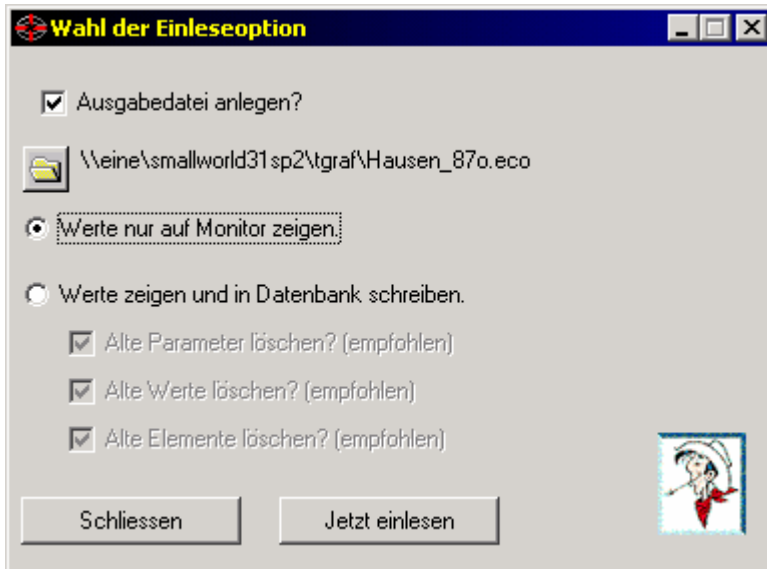


Abbildung 6.8: Menü zur Auswahl der Einleseoptionen

Große Ergebnisdateien wie die \*.dac-Datei können immer fehlerbehaftet sein. Die \*.eco-Datei dient daher der Kontrolle über die eingelesenen Daten. Zweck dieser Outputdatei ist es außerdem, dass der Anwender nach dem Processing in FEFLOW eine Kontrolle über die Einstellungen des Modells selber erhält. Falls es zu einer fehlerhaften Zuweisung kam, sind Parameter dem Modell neu zuzuweisen. Darüber hinaus ist eine knappe Form der von MODINT interpretierten Modelldefinition immer eine sehr geeignete komprimierte Form für eine weitere Verarbeitung der Daten. Die Nachricht `--- normal exit ---` am Dateiende informiert den Anwender darüber, dass die Datei erfolgreich erzeugt und der Einlesevorgang korrekt beendet wurde.

Der Eintrag eines *lokalen* Parameters in die \*.eco-Datei verläuft nach dem Schema:

(i) < Parametername >

FEFLOW unit of found parameter: < Einheit >

The parameter was found in line < Zeile des Parameternamens > of the searched file.

In the considered problem, this is a local parameter.

Here's an overview about the assigned values which were found in the searched file:

```
< Wert 1 > was found in line < Zeile 1 >
< Wert 2 > was found in line < Zeile 2 >
.....
< Wert n > was found in line < Zeile n >
```

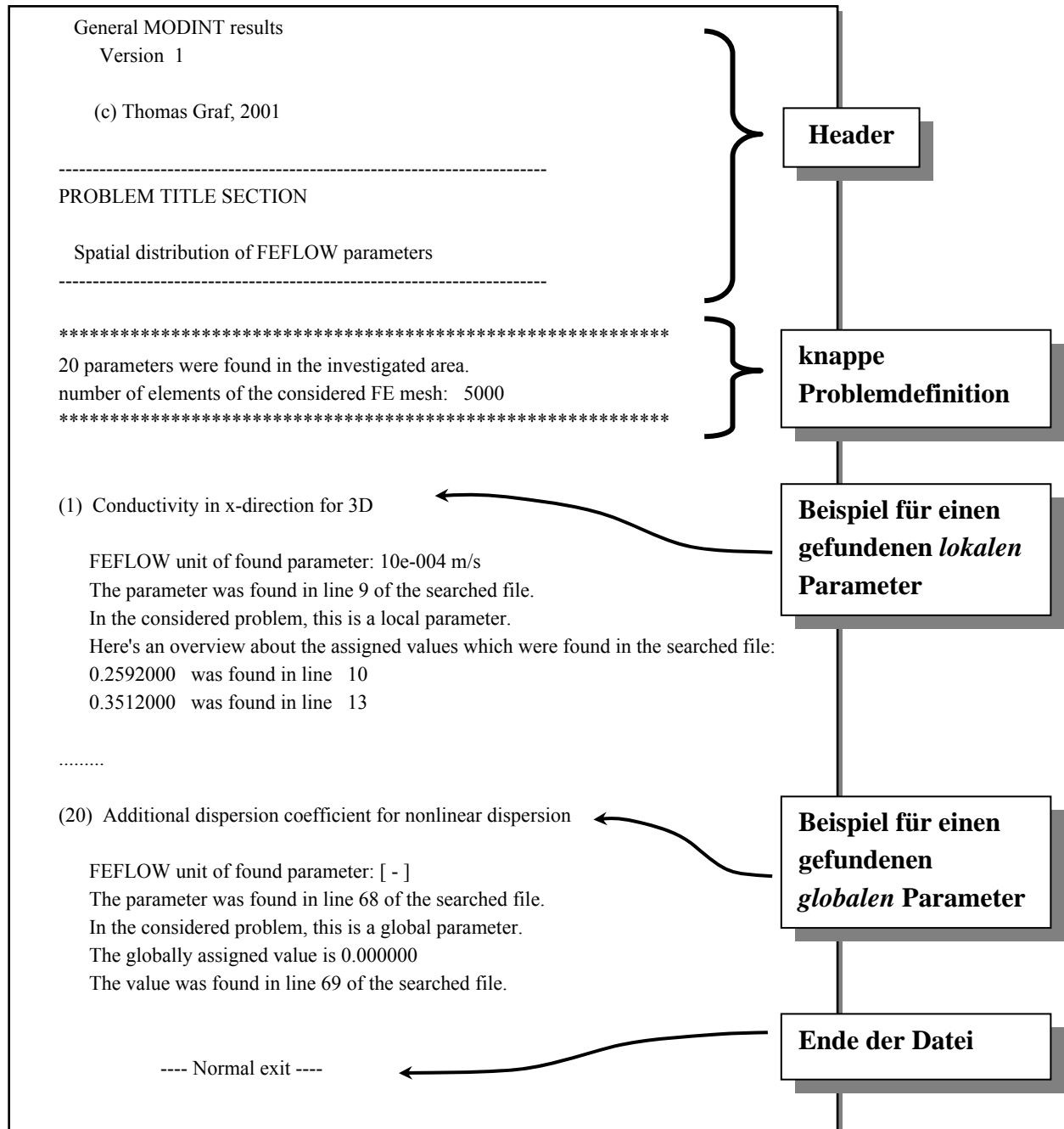


Abbildung 6.9: Struktur der \*.eco-Datei

Der Eintrag eines *globalen* Parameters in die \*.eco-Datei verläuft nach dem Schema:

(i) < Parametername >

FEFLOW unit of found parameter: < Einheit >

The parameter was found in line < Zeile des Parameternamens > of the searched file.

In the considered problem, this is a global parameter.

The globally assigned value is < Parameterwert >

The value was found in line < Zeile des Parameterwertes > of the searched file.

## 6.6 Auswahl zur Visualisierung von Modellparametern

Wenn die \*.dac-Datei erfolgreich eingelesen wurde, erreicht der Anwender von der zentralen Oberfläche (vgl. Abbildung 6.3) aus eine graphische Oberfläche (dargestellt in Abbildung 6.10), in der der Parameter zu wählen ist, dessen Werte flächenhaft verteilt angesehen werden sollen. Darin steckt hinter dem Button Parameter den Elementen zuordnen die Schnittstelle GIS - FEFLOW, wiedergegeben durch die vom GIS erkannte Georeferenz des Modells. Der Button Style zuweisen repräsentiert die Verbindung von MODINT zum Visualisierungstool VIS\_TOOL, in dem die Art des farblichen Verlaufes gewählt wird.

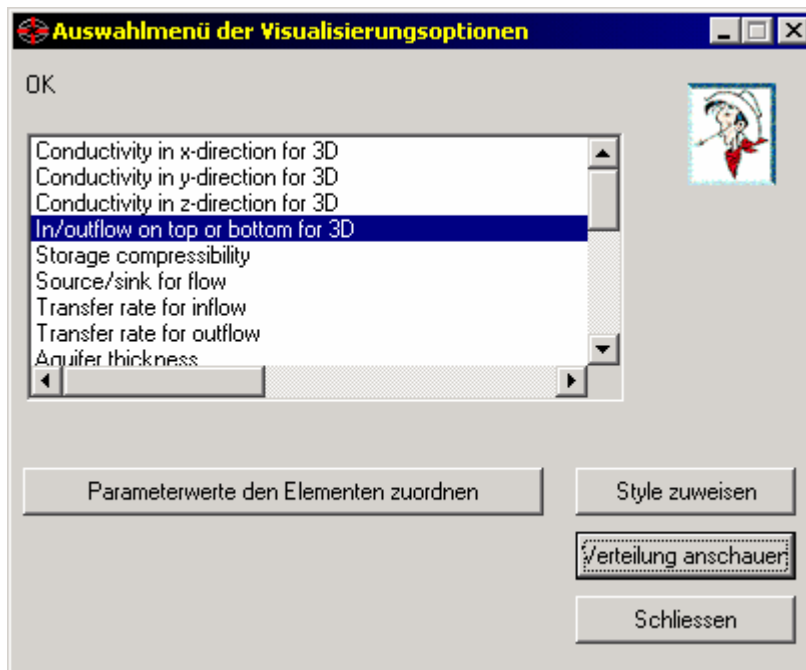


Abbildung 6.10: Menü zur Auswahl des darzustellenden Parameters

## 6.7 Datenmodell von MODINT

In dem in Abbildung 6.8 gezeigten Menü besteht die Möglichkeit Parameterdaten während des Einlesevorganges der \*.dac-Datei in die Datenbank zu übernehmen. Dafür wurde ein

passendes Datenmodell im CASE-Tool geschaffen. In die Erzeugung des Modells gingen folgende Überlegungen ein:

- (1) Ein Parameter besitzt immer die eindeutig zuweisbaren Attribute
  - Name
  - Einheit
  - global oder lokal
  - Anzahl der Wertebereiche
  - Zeile, in der der Parametername steht
- (2) Ein Parameter besitzt immer 1...n Werte (mit  $n \geq 1$ )
  - Daher ist eine 1:n -Beziehung eines Parameters zu seinen Werten erforderlich
  - Jeder Wert steht in genau einer Zeile
- (3) Eine Parameterwert ist immer 1...n Elementen zugeordnet (mit  $n \geq 1$ )
  - Daher wieder eine 1:n -Beziehung zu der Tabelle, die die Elemente enthält
  - Im Falle eines globalen Parameters entspricht die Größe der Elementetabelle der Gesamtzahl Elemente des Modellgebietes. Dann verweisen alle Elemente auf einen Wert, den globalen Parameterwert.

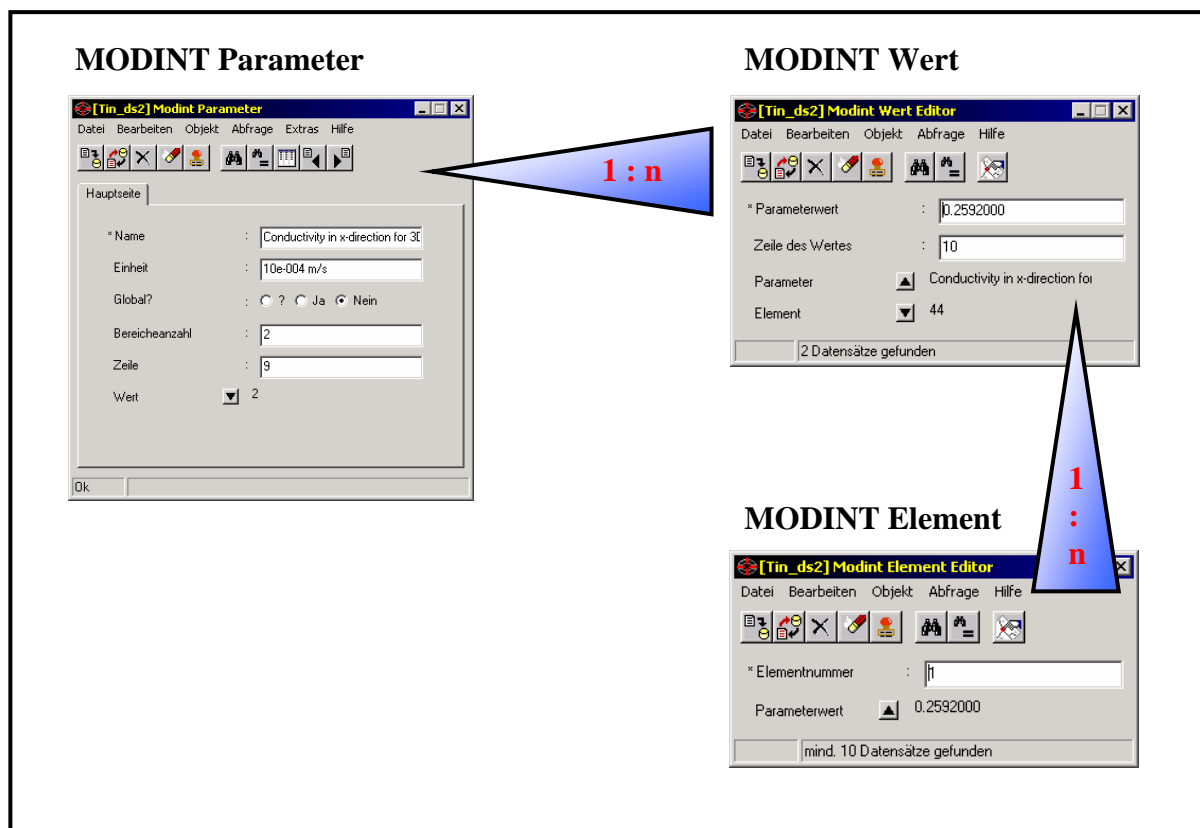


Abbildung 6.11: Die Editoren des Datenmodells von MODINT

So entstand das in Abbildung 6.11 durch die Editoren der Tabellen beschriebene Datenmodell. Für weitere Informationen sei verwiesen auf Graf (2001).

## 6.8 Fazit

Das Modul MODINT schafft die Möglichkeit der Integration des Simulators FEFLOW in SMALLWORLD GIS. Es wurde eine Schnittstelle zwischen dem Simulator FEFLOW und dem GIS erstellt. Wiedergegeben wird diese Schnittstelle zum Einen über das im Modul OGRUT realisierte Einlesen des in FEFLOW generierten Finiten Elemente Netzes. Zum Anderen stellt MODINT die Möglichkeit bereit Outputdateien (\*.dac und \*.fem) von FEFLOW in das GIS einzulesen und die Art der Parameterverteilung zu prüfen. Dabei wird die \*.fem-Datei verwendet, um globale Parameterwerte zu manipulieren, während die \*.dac-Datei die flächenhafte Verteilung der Parameter liefert. Es wurde damit also die Möglichkeit geschaffen vom GIS aus zu zwei verschiedenen Zeitpunkten mit FEFLOW zu kommunizieren. In einem weiteren Schritt werden die Werte eines ausgesuchten Parameters unter Berücksichtigung der Geometrie den Elementen des FE-Netzes zugeordnet. Das Visualisierungstool VIS\_TOOL liefert den Style, mit Hilfe dessen die Werte als farblicher Verlauf im GIS darstellbar sind. Wählbar sind im VIS\_TOOL Verlauf der Farbe, Minimalwert, Maximalwert und die Anzahl der Farbschritte. Ausgeben lassen sich eingelesene Daten auf dem Monitor, in einer Outputdatei bzw. in der Datenbank. Per Default werden alte Datenbankeinträge gelöscht.

Darüber hinaus wurde ein zentrales Steuermenü erstellt, in das die Funktionen von MODINT und die aufgearbeiteten Sourcen von OGRUT eingearbeitet sind.

Die beiden FEFLOW-Outputdateien unterscheiden sich inhaltlich nur dadurch, dass in der \*.dac-Datei auch die Modellergebnisse (hydraulische Potenziale) zu allen gerechneten Zeitpunkten stehen. Obzwar diese zusätzliche Information nicht explizit eingelesen und in eine flächenhafte Darstellung umgewandelt wurde, wurde auch für die \*.dac-Datei ein separater Parser geschrieben. Dies eröffnet die Möglichkeit im Rahmen einer weiteren Arbeit die flächenhafte Darstellung auch der Ergebnisse zu realisieren und so MODINT zu erweitern. Angesichts der zahlreichen Kommentare im MODINT-Quellcode dürfte diese Erweiterung kein programmiertechnischer Aufwand sondern nur eine Frage der Zeit sein.

Es stehen also nun die Werkzeuge zur Parametrisierung (Zuweisung von Parametern und Randbedingungen an das Modell) sowie die Rücknahme der vorgenommenen Parameterverteilungen zur Verfügung. In einem weiteren Schritt kann noch die Ergebn isrücknahme realisiert werden.



## 7 Ergebnisse

Ziel dieser Arbeit war es eine Schittstelle zwischen dem Simulator FEFLOW und dem GIS zu programmieren und damit die Modellintegration zu schaffen. Hilfreich hierfür war die Möglichkeit ein Finites Elemente Netz ins GIS zu laden, die im Zuge der Erstellung des Modules OGRUT erzeugt wurde. Auf dieses Netz lassen sich räumliche Verteilungen von beliebigen Größen abbilden, im vorliegenden Fall sind dies die Modellparameter von FEFLOW. Die genaue Verteilung der Parameterwerte steht in \*.dac-Dateien, das sind Ausgabedateien, die beim Processing des Grundwassermodells erstellt werden. In MODINT wurde die Möglichkeit geschaffen derartige Dateien einzulesen und die flächenhafte Verteilung zu analysieren. Die erkannten Parameterdaten können wahlweise auf dem Monitor ausgegeben, in eine Ausgabedatei oder in die Datenbank geschrieben werden. Hier wurde die leistungsstarke Datenbank-Architektur von SMALLWORLD GIS ausgenutzt, um Modellergebnisse geeignet zu speichern und so vorzuhalten.

Herangezogen wurde außerdem das Visualisierungstool VIS\_TOOL. Es erlaubt einfaches Zuweisen von Farbstyles an die Elemente des FE-Netzes. Wie OGRUT wurde auch VIS\_TOOL in MODINT integriert, so dass sich mit Hilfe des Visualisierungstools von MODINT aus die Werteverteilungen als farblichen Verlauf im GIS darstellen lassen.

Außerdem wurde eine zentrale Nutzerumgebung erstellt, von der aus alle zur Modellintegration notwendigen Funktionen sowie die OGRUT Werkzeuge zugänglich sind.

### 7.1 MODINT als GIS - Werkzeug

Zur Realisierung der Modellintegration von FEFLOW wurde das Modul MODINT erstellt. Die Schnittstelle ist repräsentiert durch programmierte Algorithmen, die von FEFLOW generierte Dateien (\*.fem und \*.dac) lesen, analysieren und gefundene Daten anzeigen und/oder in die Datenbank schreiben. Auf diese Weise stehen die Daten späterer Verarbeitung (Visualisierung) zur Verfügung. Die Kommunikation mit FEFLOW wurde durch folgende Arbeiten geschaffen:

- Programmieren von Parsern zur Analyse von FEFLOW - Dateien
- Implementierung dieser Einlesefunktionalitäten in eine GUI
- Aufarbeiten, Korrigieren und Integrieren des Programmcodes des bereits bestehenden Moduls OGRUT in die GUI
- Schaffung einer Auswahlmöglichkeit von visualisierbaren Größen zur Darstellung im GIS.
- Visualisierung der eingelesenen Daten aus FEFLOW - Dateien

MODINT ist nicht nur eine benutzerfreundliche Oberfläche sondern auch ein GIS-Werkzeug, das die geforderte Aufgabe von Verwaltung und Sichtbarmachung von Modellparametern verwirklicht. Dabei liegt besonderes Augenmerk darauf, dass in MODINT verschiedene WAQIS-Module ineinandergreifen. Abbildung 7.1 illustriert, welche Module welche Aufgaben übernehmen. Die Nutzerfreundlichkeit kommt durch folgende Implementierungen zum Ausdruck:

- Durch Einstellungen der Selektierbarkeit von Buttons und Menüfunktionen werden Fehlbedienungen von vorne herein verhindert. Außerdem führt so der Anwender die Funktionen in der richtigen Reihenfolge aus.
- Statusanzeigen von Prozessen informieren ständig über den Stand eines Vorganges und darüber, wie weit der Prozess fortgeschritten ist. Der Prozessfortschritt wird in Prozent angegeben und nicht in verbleibender Zeit. Durch letzteres würde der Rechenaufwand und damit die benötigte Zeit unnötig in die Höhe getrieben werden. Das Einlesen einer großen \*.dac-Datei nimmt ohnehin schon bis zu 45 Minuten in Anspruch.
- Zeitintensive Einleseprozesse laufen in Threads (vgl. dazu Kapitel 4.3.1) ab. Das bewirkt, dass der Anwender gleichzeitig auf andere Funktionalitäten zugreifen kann. Außerdem ist dann der Prozeß über eine entsprechende Schaltstelle beendbar.
- Beim Aufruf von bedeutenden Funktionen wie Zurücksetzen der Einstellungen, Schließen eines Editor oder Ansehen der produzierten flächenhaften Verteilung im GIS (Durchführung eines `hard_reset`), erhält der Nutzer Warnungen, die ihn über die Relevanz der gewünschten Funktion informieren, ihn über dessen Konsequenz in Kenntnis setzen und eine Abbruch-Option offenhalten.

Im Rahmen von MODINT wurde eine Schnittstelle zwischen dem Grundwassermodell FEFLOW und SMALLWORLD GIS geschaffen. Dabei kann zu verschiedenen Zeitpunkten (nach dem Preprocessing und nach dem Processing) mit FEFLOW interagiert werden. Die Visualisierungsoption des entwickelten Moduls MODINT ist somit als eine Art *Postprocessing* aufzufassen, da es gewonnene Daten nach der eigentlichen Modellrechnung verarbeitet. Abbildung 7.2 zeigt nochmal das Zusammenspiel des Simulators mit dem GIS.

## 7.2 Ergebnisplots

Dem in Abbildung 3.2 gezeigten Untersuchungsgebiet liegt ein Netz zugrunde, das die Fläche diskretisiert. Da der Simulator FEFLOW verwendet wurde, ist dieses Netz ein Finites Elemente Netz (Abbildung 7.3). Es bildet die geometrische Basis für das Abbilden der Werteverteilung ausgesuchter Parameter. Als Beispiel hierfür wird in Abbildung 7.4 die flächenhafte Verteilung des Speicherkoeffizienten aufgeführt, weitere Ergebnisse finden sich im Anhang in Abbildung A.1 bis Abbildung A.4.

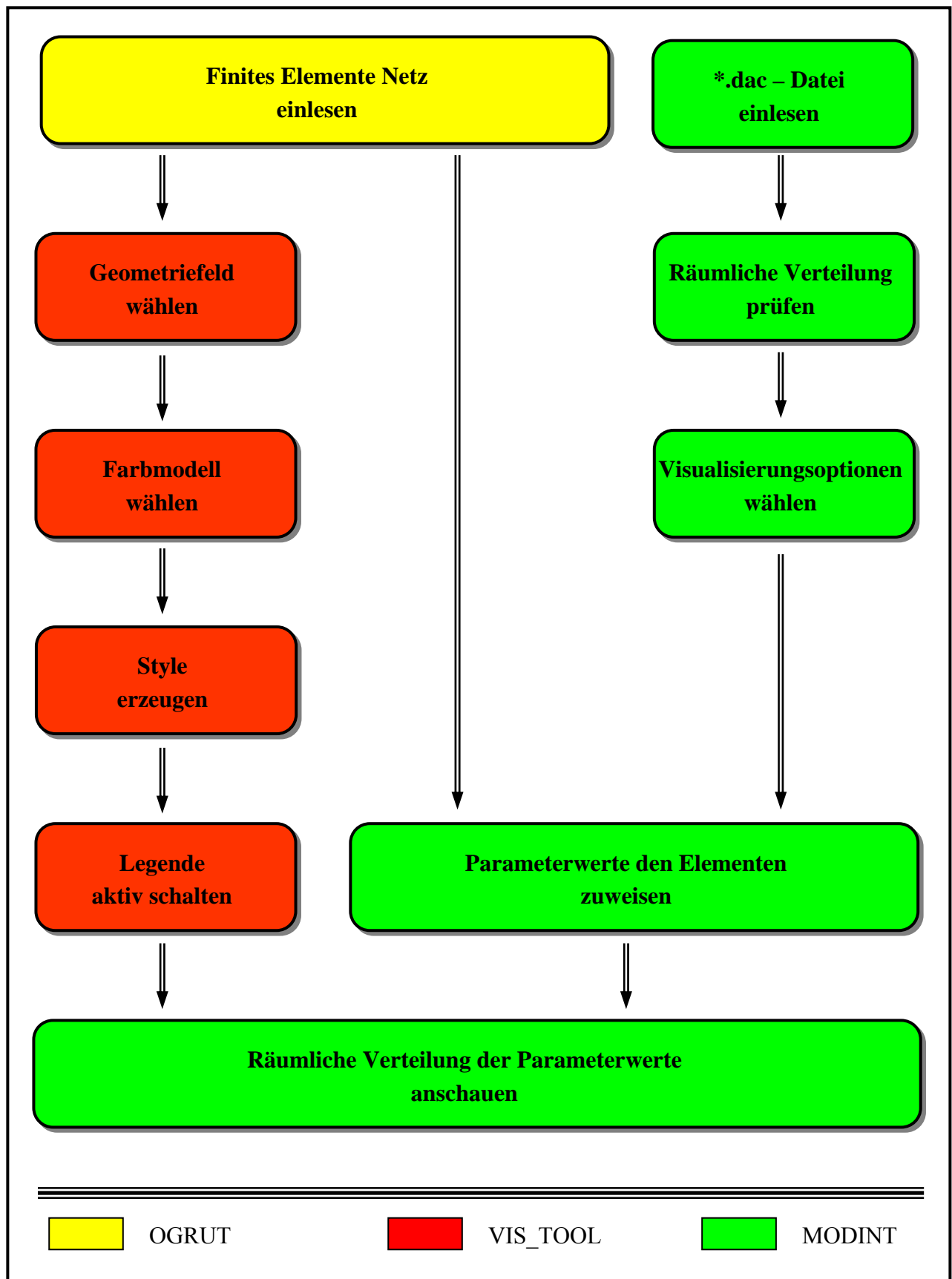


Abbildung 7.1: Erzeugung einer räumlichen Verteilung der FEFLOW-Parameterwerte

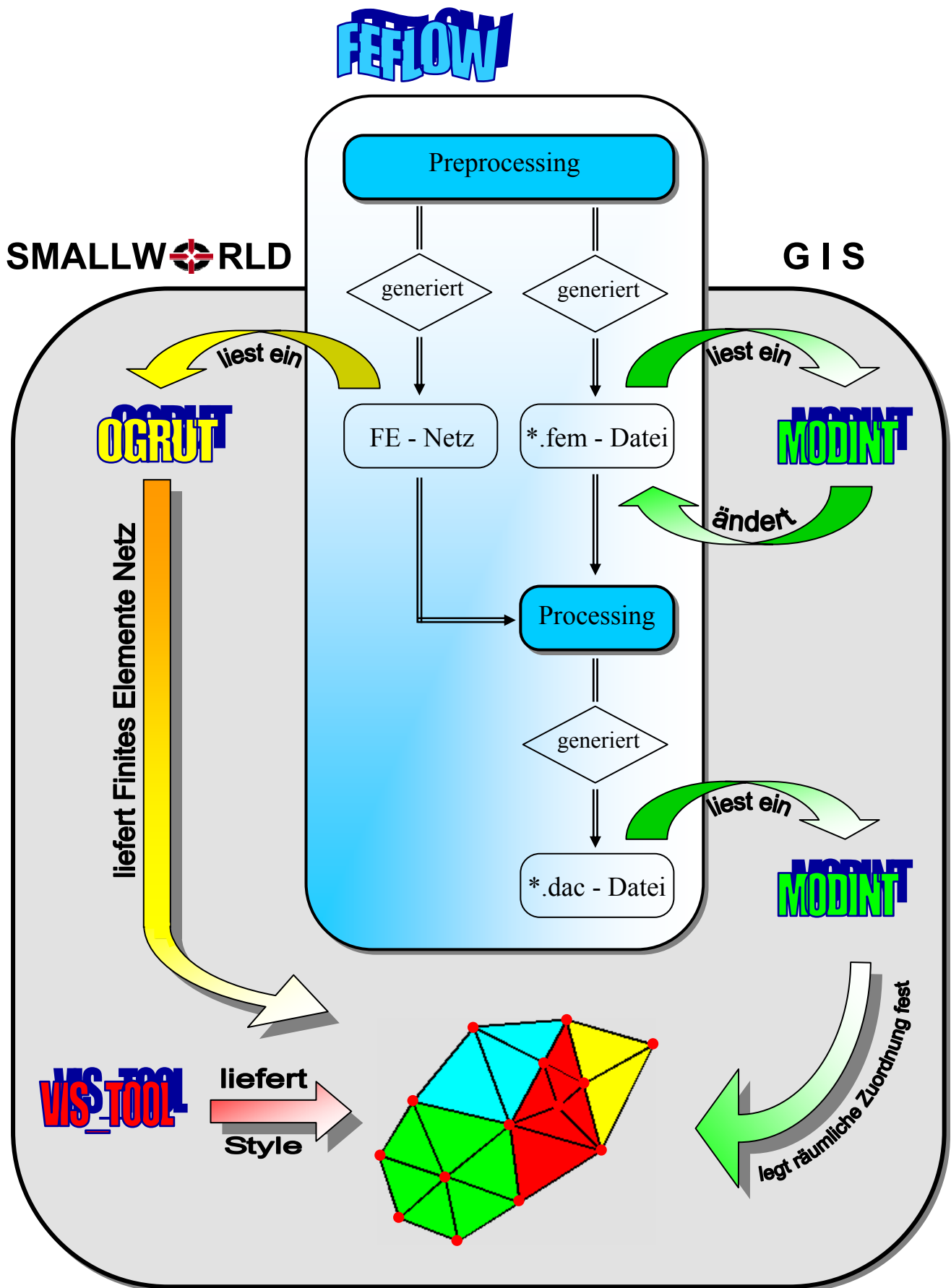


Abbildung 7.2: Zusammenspiel der verschiedenen Module von SMALLWORLD GIS und dem Simulator FEFLOW

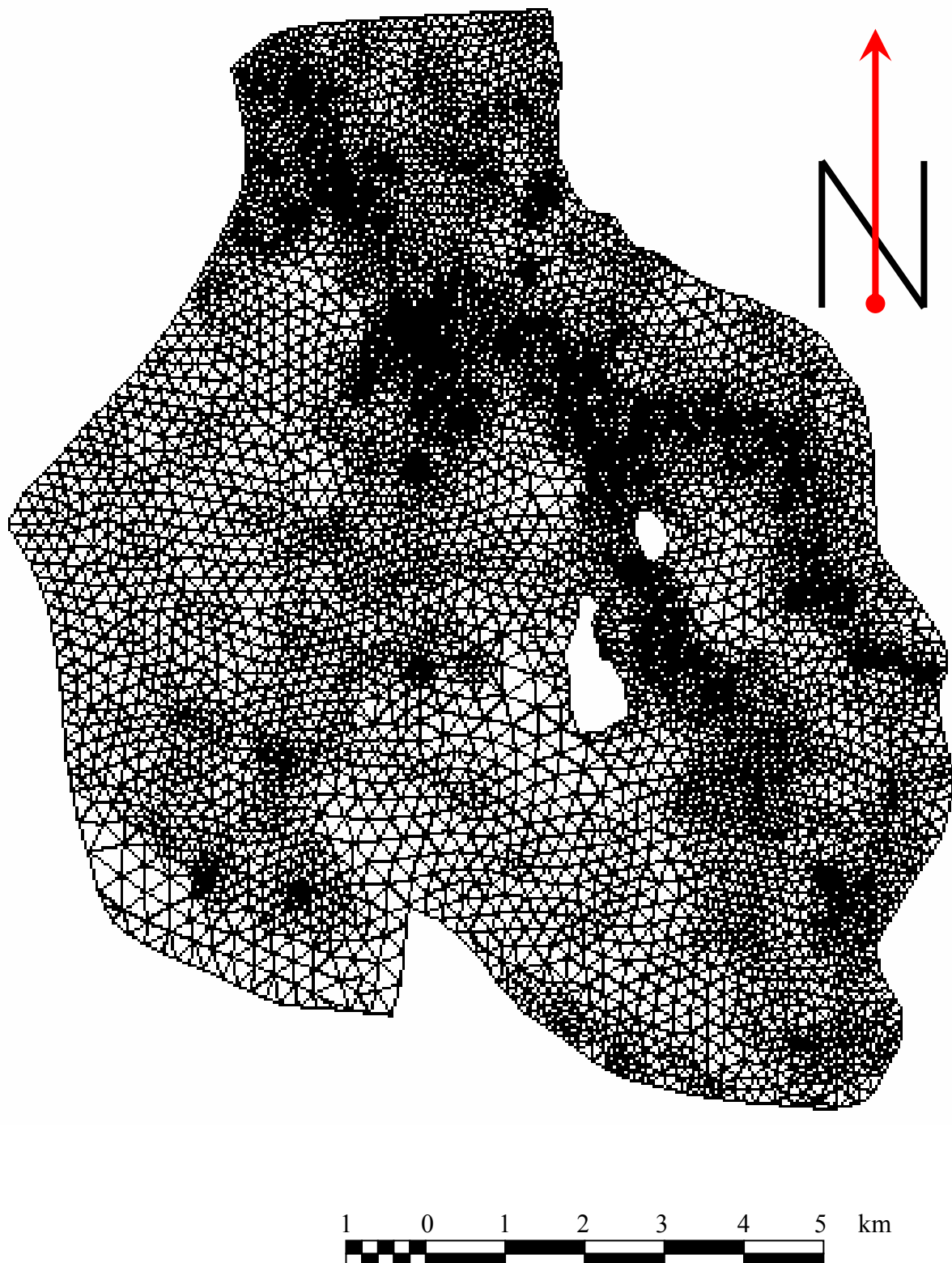


Abbildung 7.3: Das zugrunde liegende Finite Elemente Netz

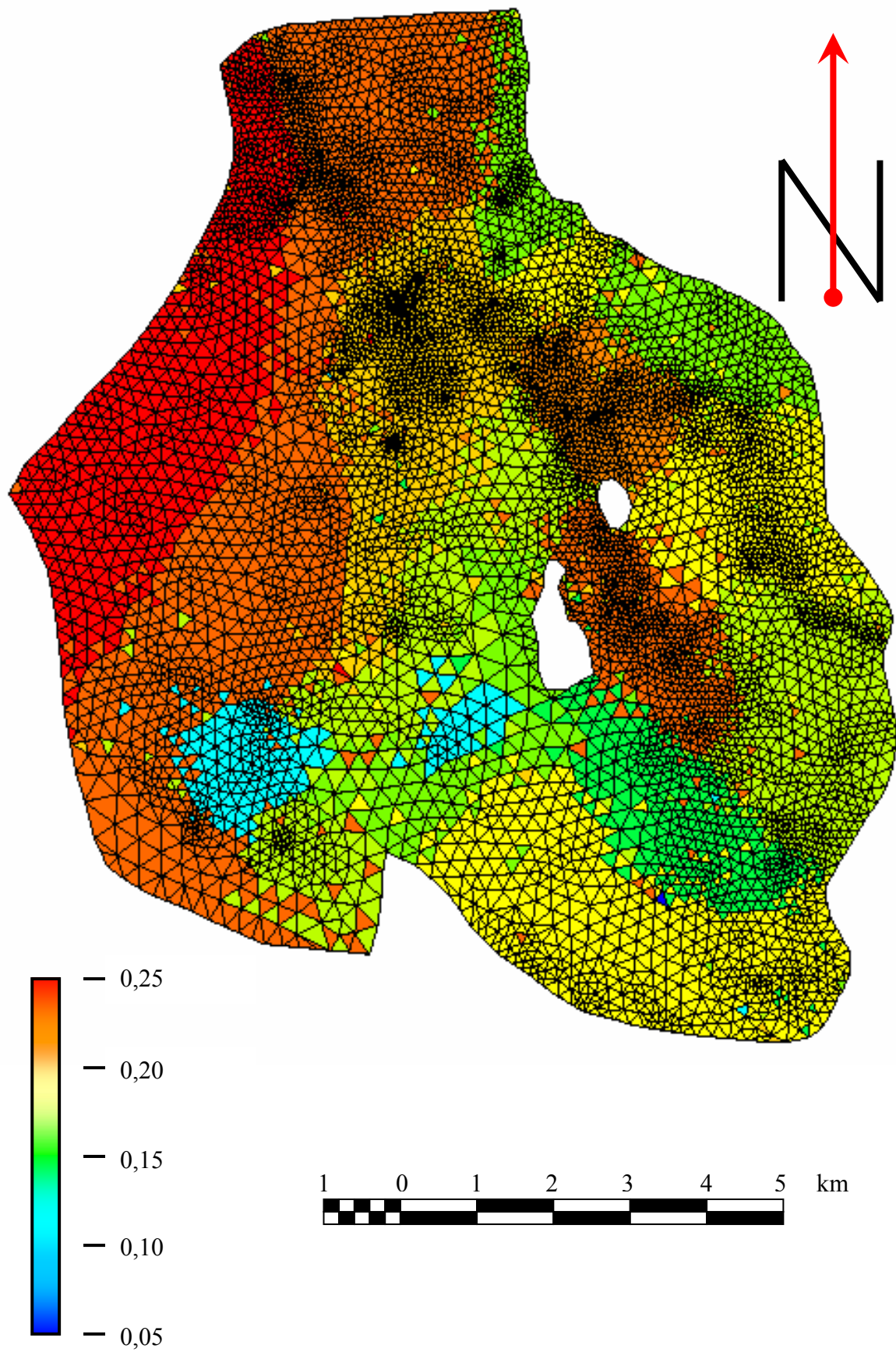


Abbildung 7.4: Verteilung des Speicherkoefizienten im Untersuchungsgebiet

## 8 Schlussbetrachtung und Ausblick

Das Modul MODINT zur Modellintegration ist ein weiteres unter SMALLWORLD GIS entwickeltes Auswertewerkzeug im WAQIS-Projekt. Es stellt die Schnittstelle zum Simulator FEFLOW (Finite Element Subsurface FLOW System) dar. Im Zuge des WAQIS wurden bereits zahlreiche Tools erstellt. Ein im Rahmen von MODINT erstelltes Auswertemenü eröffnet die Möglichkeit auf eine Reihe von Funktionalitäten zuzugreifen. Dieser zentrale Editor ist so eingerichtet, dass er bei Bedarf in beliebiger Weise erweitert und ergänzt werden kann. Auch wurde auf Anwenderfreundlichkeit Wert gelegt, was sich in ständig aktualisierten Statusanzeigen, Ausgrauen (nicht-Selektierbarkeit) von Buttons und Menüeinträgen sowie Warnmeldungen vor relevanten Prozessen ausdrückt.

Schließlich bildet den Kern von MODINT die interface zu FEFLOW. Als farblichen Verlauf lassen sich die Werte von Modellparametern aus FEFLOW räumlich verteilt darstellen. Auf diese Weise erhält der Nutzer einen raschen Überblick über vorgenommene Einstellung bzw. flächenhafte Verteilung eines ausgesuchten Parameters. Er kann dadurch sein regionales Wissen über die Fließ- und Transportprozesse im Untergrund erweitern. Die erwünschte Rückgabe der Modellergebnisse war Ziel dieser Arbeit, wurde aber in MODINT nicht integriert. Dies kann Teil einer weiteren Arbeit sein.

Anwender von SMALLWORLD GIS sind in aller Regel mit Grundwassermodellen wie FEFLOW weniger vertraut. Durch die Integration des Modells und durch die Schaffung einer nutzerfreundlichen Oberfläche, von der aus auf Modell-Outputs zugegriffen werden kann, ist dieser Stein aus dem Weg geräumt, wodurch die Menge auswertbarer Information erheblich steigt. Neben dieser Arbeit wurde ein **MODINT-Handbuch** erstellt, das eine umfassende Bedienungsanleitung liefert und einen Einblick in die Programmierstruktur von MODINT gibt.

Mit der Erzeugung von MODINT wurde die Kommunizierung zwischen den Modell FEFLOW und der Datenbank von SMALLWORLD GIS komplettisiert. Die Georeferenz der Elemente im Modell wird vom GIS verstanden und kann daher abgebildet und die Elemente können mit Daten belegt werden. Somit existiert nun der letzte Baustein in der Reihe GIS → Parametrisierung des Modells → FEFLOW → Rückgabe der Ergebnisse und Parameter → Darstellung im GIS.

Beispielhaft wurde die Integration des Simulators FEFLOW in SMALLWORLD GIS am Einzugsgebiet des Wasserwerkes Hausen an der Möhlin mit Erfolg getestet. Abbildung 7.4 sowie die Abbildungen im Anhang zeigen die flächenhaften Verteilungen der Modellparameter.

Die in dieser Arbeit vorgestellte Integrationsmöglichkeit des Grundwassermodells FEFLOW ist als Ansatz zu verstehen. Im folgenden sollen Vorschläge für weitere Arbeiten gegeben werden.

- Insbesondere kann noch die Rücknahme und Darstellung von **Modellergebnissen** realisiert werden. Die Grundlagen wurden in MODINT bereits geschaffen: Ergebnisdateien können schon von MODINT eingelesen werden.
- Auch sind die Ergebnisse zu verschiedenen **Zeitpunkten** von Interesse. Es wären daher noch Methoden zu programmieren, mit Hilfe derer hydraulische Potenziale zu ausgesuchten Zeitpunkten gelesen und dargestellt werden.
- Zurückgegebene Ergebnisse sollen weiter verarbeitbar sein. Dazu wäre noch ein Werkzeug zu erstellen, was **Bearbeitungsoptionen** wie die Ermittlung eines Maximums oder Minimums, die Schwankungsbreite bzw. ein zeitliches oder räumliches Mittel der Grundwasserstände bereithält.
- Im vorgestellten Fall wird eine \*.dac-Datei dreimal geprüft. Zunächst wird die Größe als Zeilenzahl ermittelt, dann eine zweidimensionale Matrix erstellt und schließlich werden die Parameterdaten selber ermittelt. Es kann untersucht werden, ob es möglich ist das Analysieren auf einen oder zumindest **zwei Einlesevorgänge** zu beschränken.
- Die **Art der Simulation** wurde noch nicht berücksichtigt. Gegebenenfalls möchte der Anwender Auskunft darüber erhalten, ob es sich um ein reines Grundwasserfließ- oder etwa ein Stofftransportmodell handelt.



## 9 Literaturverzeichnis

- Baumgartner, A. und Liebscher, H.-J., 1996: Allgemeine Hydrologie - Quantitative Hydrologie. Gebrüder Borntraeger, Berlin/Stuttgart, 694 Seiten.
- Cameron, D. und Rosenblatt, B., 1991: Learning GNU EMACS. O'Reilly UK, UK, 144 Seiten.
- Domenico, P.A. und Schwartz, F.W., 1998: Physical and chemical hydrogeology. John Wiley & Sons, New York, 506 Seiten.
- Eberle, K., 1999: BOMET, Aufbau eines bodenkundlich - meteorologischen Teilinformationssystems im GIS SMALLWORLD. Unveröffentlichte Diplomarbeit, Institut für Hydrologie, Albert-Ludwigs-Universität Freiburg im Breisgau, 60 Seiten.
- Frieg, B., 1987: Hydrogeologie und Grundwasserhydraulik des Einzugsgebietes des Wasserwerkes Freiburg-Ebnet, Inaugural-Dissertation, Naturwissenschaftlich-Mathematische Gesamtfakultät, Ruprecht-Karls-Universität Heidelberg.
- GE Smallworld, 1997a: MAGIK: SMALLWORLD MAGIK style guide. GE SMALLWORLD, United Kingdom, 13 Seiten.
- GE Smallworld, 1997b: SMALLWORLD GIS 2 Case Tutorial. GE SMALLWORLD, United Kingdom, 20 Kapitel.
- GE Smallworld, 1997c: SMALLWORLD GIS 2 Magik. GE SMALLWORLD, United Kingdom, 45 Kapitel.
- GE Smallworld, 1999a: SMALLWORLD 3 Application Development Overview, United Kingdom, 10 Kapitel.
- GE Smallworld, 1999b: SMALLWORLD GIS 3 Overview. GE SMALLWORLD, United Kingdom, 78 Seiten.
- Genser, H., 1959: Stratigraphie und Tektonik der Vorbergzone am südwestlichen Schwarzwald zwischen Staufen und Badenweiler. Ber. Naturf. Ges. Freiburg i.Br., 49: 59-112, Freiburg i.Br.
- Geyer, O. F. und Gwinner, M. P., 1991: Geologie von Baden-Württemberg. E. Schweizerbart'sche Verlagsbuchhandlung (Nägele und Obermiller), Stuttgart, 482 Seiten.

- Glickstein, B., 1997: Writing GNU EMACS Extensions - Editor customizations and creations with LISP. O'Reilly UK, UK, 233 Seiten.
- Graf, T., 2001: MODINT - Handbuch zur Integrationsmöglichkeit von FEFLOW in SMALLWORLD GIS. Unveröffentlichtes Handbuch zur Diplomarbeit, Institut für Hydrologie, Albert-Ludwigs-Universität Freiburg im Breisgau, 39 Seiten.
- Herrmann, J., 2000: OGRUT, GIS-Integration von FEFLOW in das Freiburger Wasserqualitätsinformationssystem WAQIS. Unveröffentlichte Diplomarbeit, Institut für Hydrologie, Albert-Ludwigs-Universität Freiburg im Breisgau, 90 Seiten.
- Kiefer, J., 2001: HGEO & IHF-ISO - Integration der Grundwassermodellierungsumgebung FEFLOW in das Freiburger Wasserqualitäts-Informationssystem WAQIS (2). Unveröffentlichte Diplomarbeit, Institut für Hydrologie, Albert-Ludwigs-Universität Freiburg im Breisgau, 66 Seiten.
- Kinzelbach, W., 1992: Numerische Methoden zur Modellierung des Transports von Schadstoffen im Grundwasser. R. Oldenbourg Verlag, München/Wien, 343 Seiten.
- Klink, W., 2001: Weiterentwicklung und Einsatz des Bodenkundlich - Meteorologischen Moduls „BOMET“. Unveröffentlichte Diplomarbeit, Institut für Hydrologie, Albert-Ludwigs-Universität Freiburg im Breisgau, 69 Seiten.
- Mäckel, R. und Metz, B., 1997: Schwarzwald und Oberrheintiefland - Eine Einführung in das Exkursionsgebiet um Freiburg im Breisgau. Selbstverlag des Institutes für Physische Geographie der Albert-Ludwigs-Universität Freiburg i.Br., Freiburg, 334 Seiten.
- Maniak, U., 1997: Hydrologie und Wasserwirtschaft - Eine Einführung für Ingenieure. Springer, Berlin/Heidelberg/New York, 644 Seiten.
- McDonald, M.G. und Harbaugh, A.W. (1988): A modular three-dimensional finite-differenceground-water flow model, Techniques of Water-Resources Investigations.- Book 6, U.S.G.S., Reston, VA.
- Peter, A., 1998: Dreidimensionale, instationäre Grundwasserströmungsmodellierung für das Einzugsgebiet des Wasserwerkes Hausen an der Möhlin. Unveröffentlichte Diplomarbeit, Institut für Hydrologie, Albert-Ludwigs-Universität Freiburg im Breisgau, 133 Seiten.
- Schilcher, M., 1996: Geo-Informatik – Anwendungen, Erfahrungen, Tendenzen. Siemens AG / Publicis MCD, Deutschland, 620 Seiten.

- Therrien, R. and Sudicky, E.A., 1995. Three-dimensional analysis of variable-saturated flow and solute transport in discretely-fractured porous media. *Journal of Contaminant Hydrology*, 23: 1-44.
- Simmons, C. T., and Narayan, K.A., 1998. Modelling density dependent flow and solute transport at the Lake Tutchewop saline disposal complex, Victoria. *Journal of Hydrology*, 206(3-4), 219-236.
- Water Services Association (WSA), 1996: Waterfacts '96. UK, 74 Seiten.
- WAQIS, 2000: <http://www.ruf.uni-freiburg.de/hydrology/forsch/waqis>. Institut für Hydrologie, Albert-Ludwigs-Universität Freiburg im Breisgau, 1 Seite.
- WASY (Institute for Water Resources Planning and System Research Ltd.), 1998: FEFLOW User's Manual. WASY GmbH Berlin, Berlin, 177 Seiten.

## Anhang

Tabelle A.1: Für das *pragma-statement* verwendete Levels

Level	Beschreibung
<b>Basic</b>	Für gewöhnlichen Zugriff. Alle Programmierer dürfen diese Methode oder Klasse verwenden. Normale Kenntnisse reichen aus. Falls unsachgemäßer Gebrauch dennoch zu einem Fehler führt, ist es unwahrscheinlich, dass man größeren Schaden anrichtet.
<b>Advanced</b>	Für außergewöhnlichen Zugriff. Programmierer sollten diese Methode oder Klasse nur dann verwenden, wenn <i>basic</i> Methoden nicht ausreichend sind. Falls diese Objekte fälschlich verwendet werden, kann dies zu unbedeutenden Fehlern führen.
<b>Restricted</b>	Nur für Zugriff des Eigentümers des Code. Für das Kernprodukt von SMALLWORLD ist das die Entwicklergruppe von SMALLWORLD GIS.
<b>Debug</b>	Darf von jedem Programmierer während des Entwickelns verwendet werden, insbesondere zur Fehlersuche. Diese Methoden sollen aber nicht im fertiggestellten Produkt enthalten sein.
<b>Deprecated</b>	Diese Methoden und Klassen waren allgemein verfügbar, werden aber bald <i>restricted</i> , verändert oder sogar gelöscht werden.
<b>Unclassified</b>	Für diese Methode oder Klasse wurde entweder kein <i>pragma-statement</i> oder kein Level definiert.

## Erstellen eines Tabelleneintrages mit Hilfe eines Templates

```

template << v.collections[:haus].new_template_record()

*template.name << "Post"
*template.stadtteil << "Wiehre"
template.strasse << "Poststraße"
template.hausnummer << 20
template.offizielles_gebaeude? << _true
template.stockwerke << 3
template.mitarbeiter << die_mitarbeiter_liste
template.eigentuemer << "Axel Sundermann"

record << v.collections[:haus].insert(template)

```

## Starten eines Threads

```

_method dac_einlesen_editor.einlesen
  ##
  ## Startet die Methode do_einlesen in einem eigenen Thread.
  ##
  _global thread
  _self.run_engine(thread.low_background_priority,:do_einlesen)

_endmethod
$

```

## Beenden eines Threads

```

_method dac_einlesen_editor.abbrechen(status_proz)
  ##
  ## Beendet den Thread.
  ##

  _self.interrupt_engine()
_endmethod
$

```

## Erstellen eines Editors

```

_method der_editor.open(grs)
  ##
  ## Öffnet und aktiviert einen neuen der_editor.
  ## grs ist an die new() Methode weitergeleitet.
  ##
  _local title << "Dokumentation des FEFLOW - Problems"
  _local position << coordinate.new(20,20)
  _local width << 400
  _local height << 200

  >> _self.new(grs).activate(title,position,width,height)
_endmethod
$

```

```

_method projektinfo_editor.new(grs)
  ##
  ## Erstellt einen Klon des Editors
  ##

  >> _clone.init(grs)
_endmethod
$

```

```

_method projektinfo_editor.activate(_optional title,position,width,
                                     height)

  ##
  ## Erstellt und zeigt die Oberfläche des Editors. Wenn das
  ## Fenster noch nicht erzeugt wurde, wird die activate_in()
  ## Methode aufgerufen, mit dem Rahmen als Parameter, und kreiert
  ## das Layout des Editors.
  ##

  _if .top_frame _isnt _unset
  _then _self.reactivate(title,position,width,height); _return _self
  _endif

  _if title _is _unset _then title << _self.title _endif
  _if .parent_model _is _unset

```

```

        _then t << .top_frame <<
            _self.frame_class.new(title,agent.new(_self))
        _else t << .top_frame << dialog_frame.new_with_parent(
            .parent_model.top_frame, title, agent.new(_self))
        _endif

        ok? << _false
        _protect
            _self.activate_in(t)
            ok? << _true
        _protection
            _if _not ok? _then .top_frame << _unset _endif
        _endprotect

        _self.reactivate(_unset,position,width,height)

    >> _self
_endmethod
$

```

```

_method projektinfo_editor.activate_in(frame)
    ##
    ## liefert das Layout des Menus
    ##

    < Definitionen der Items >

_endmethod
$

```

Im folgenden sind zwei Tabellen aufgeführt, die Name, Symbol und Einheit von Material- und Stofftransportparametern enthalten, welche in FEFLOW verwendet werden.

*Tabelle A.2: Materialparameter für 3D Lösungen (WASY, 1998)*

Parameter	Symbol	Einheit
Conductivity in x-direction for 3D	$K_{11}$	$10^{-4} \text{ m} \cdot \text{sec}^{-1}$
Conductivity in y-direction for 3D	$K_{22}$	$10^{-4} \text{ m} \cdot \text{sec}^{-1}$
Conductivity in z-direction for 3D	$K_{33}$	$10^{-4} \text{ m} \cdot \text{sec}^{-1}$
In/outflow on top or bottom for 3D	$P_o$	$10^{-4} \text{ m} \cdot \text{d}^{-1}$
Density ratio for 3D and/or thermohaline transport	$\alpha$	$10^{-4}$
Storativity (drain- or fillable) or density ratio	$\varepsilon_e$	1
Storage compressibility	$S_o$	$\text{m}^{-1}$
Source/sink for flow	$Q_p$	$10^{-4} \text{ d}^{-1}$
Transfer rate for inflow	$\Phi_h^{\text{in}}$	$10^{-4} \text{ d}^{-1}$
Transfer rate for outflow	$\Phi_h^{\text{out}}$	$10^{-4} \text{ d}^{-1}$
Aquifer thickness	$b$	m

*Tabelle A.3: Stofftransportparameter (WASY, 1998)*

Parameter	Symbol	Einheit
Porosity	$\varepsilon$	1
Sorption coefficient	$\kappa$	1
Molecular diffusion	$D_d$	$10^{-9} \text{ m}^2 \cdot \text{sec}^{-1}$
Longitudinal dispersivity	$\beta_L$	m
Transverse dispersivity	$\beta_T$	m
Decay rate	$\vartheta$	$10^{-4} \text{ sec}^{-1}$
Source/sink for transport	$Q_C$	$(\text{mg/l}) \text{ d}^{-1}$
Transfer rate for influxing transport	$\Phi_C^{\text{in}}$	$\text{m d}^{-1}$
Transfer rate for outfluxing transport	$\Phi_C^{\text{out}}$	$\text{m d}^{-1}$
Additional sorption coefficient for the Freundlich or Langmuir isotherm	$k_1$	1
Additional reaction coefficient for the Michaelis-Menten mechanism	$k_2$	1



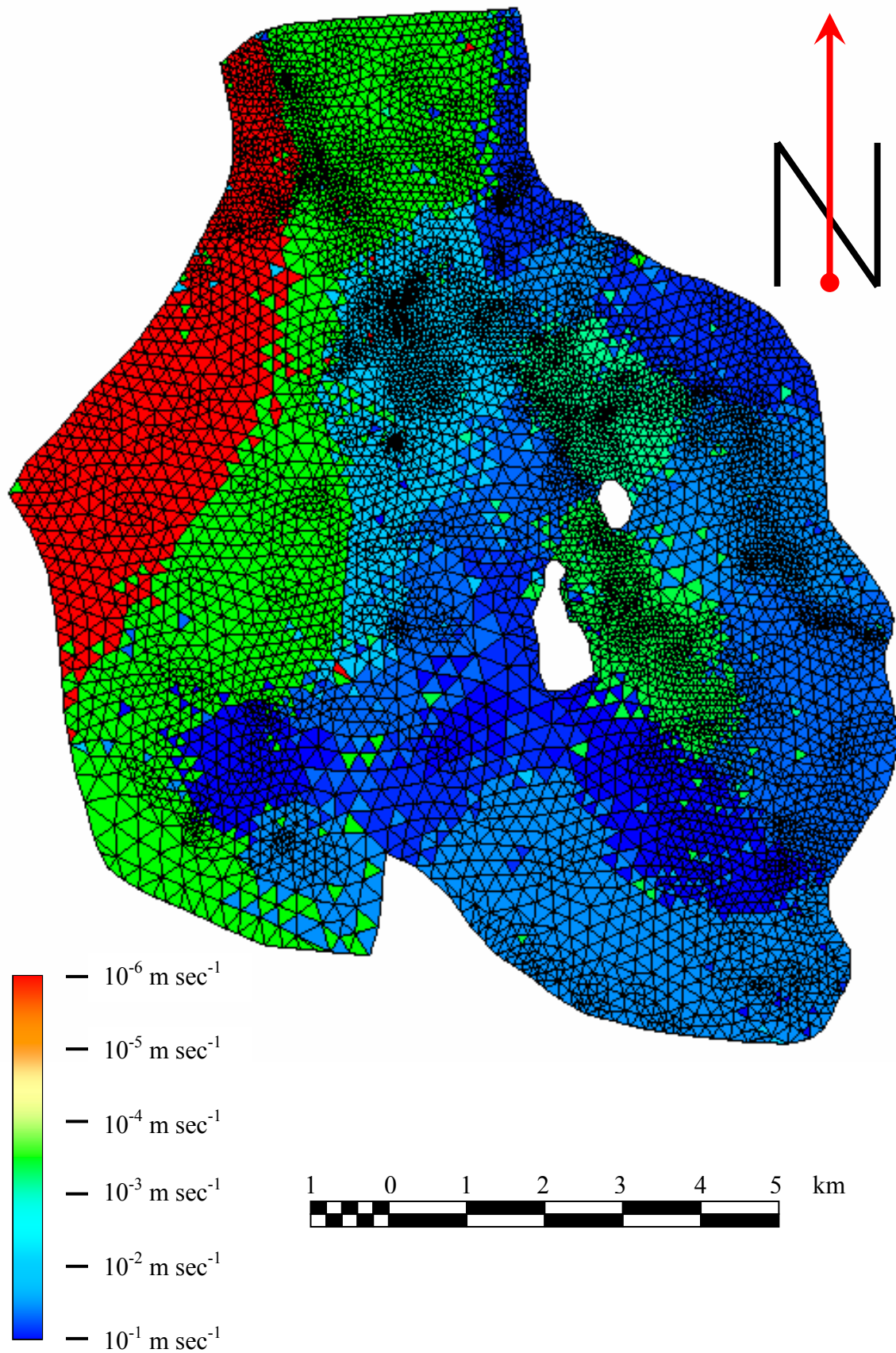


Abbildung A.1: Verteilung der hydraulischen Leitfähigkeit im Untersuchungsgebiet

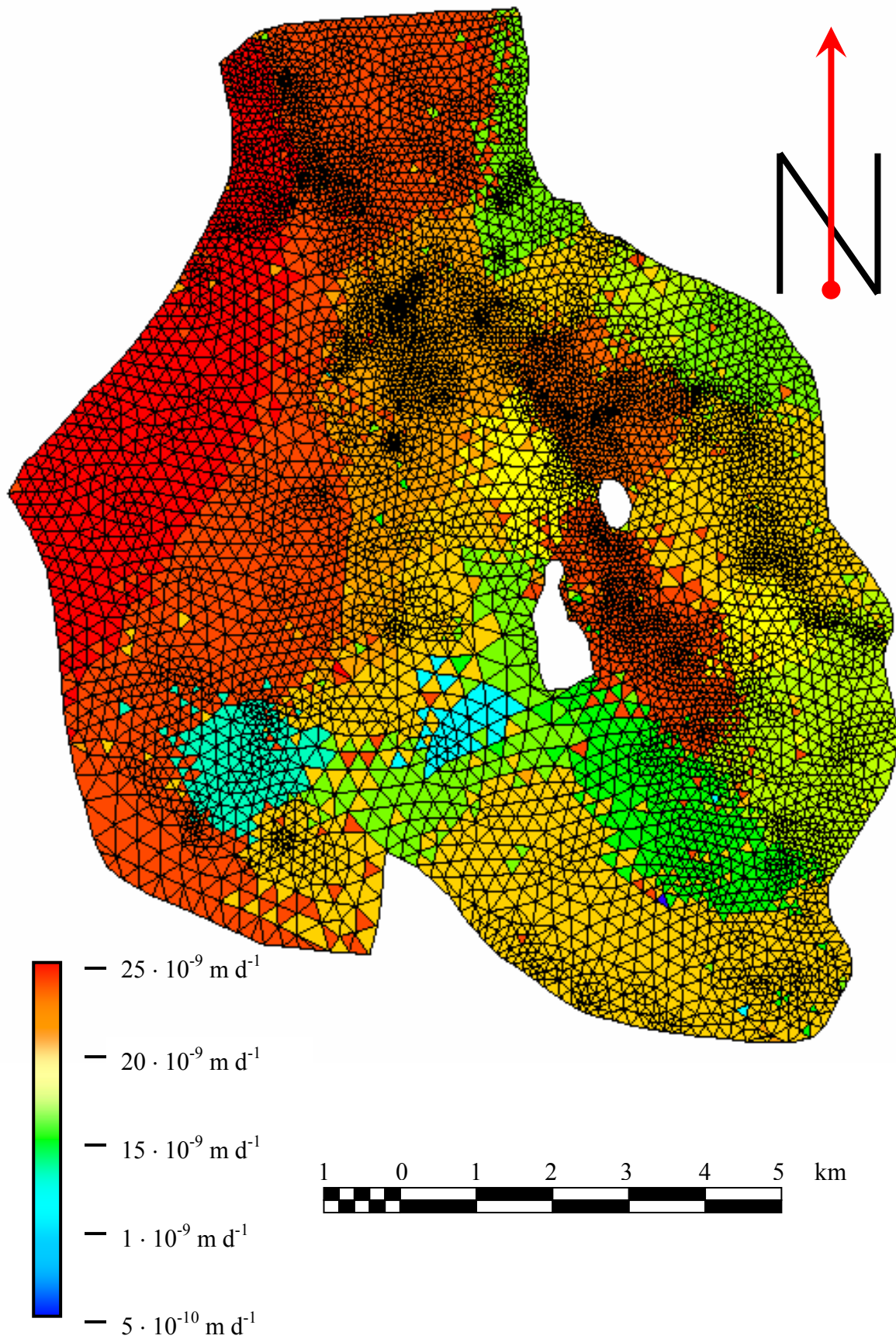


Abbildung A.2: Verteilung der Grundwasserneubildung im Untersuchungsgebiet

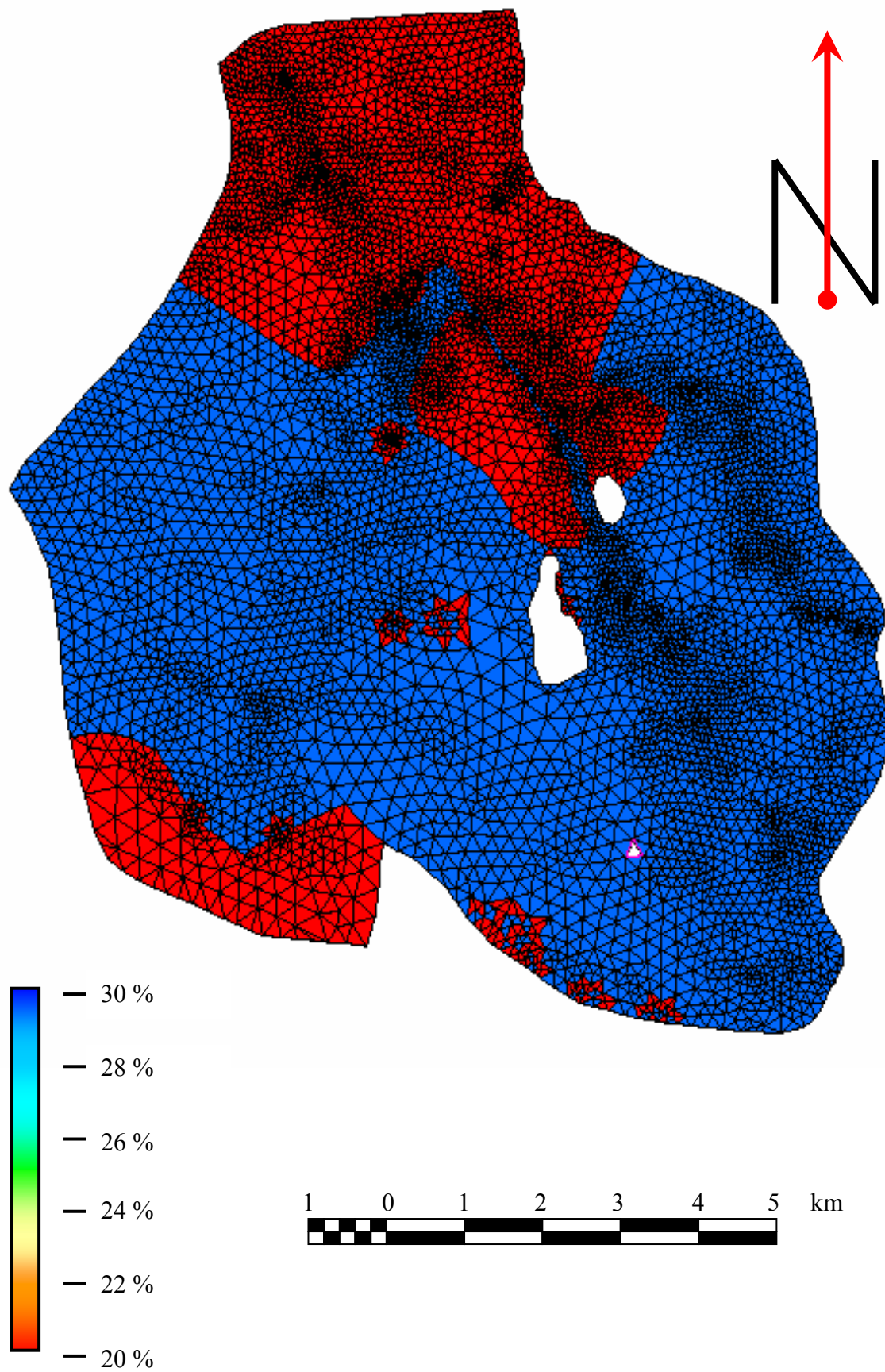


Abbildung A.3: Verteilung der Porosität im Untersuchungsgebiet



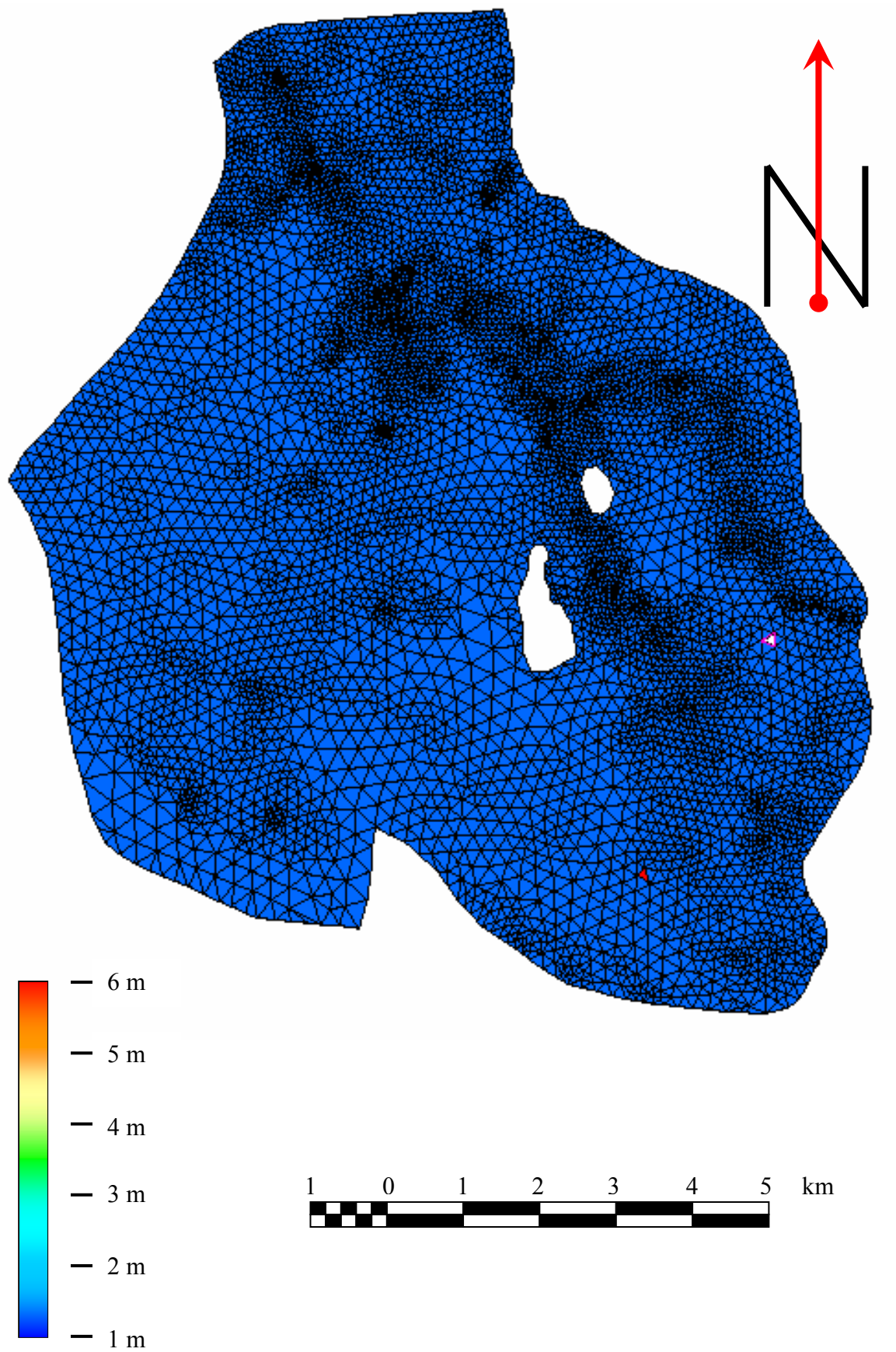


Abbildung A.4: Verteilung der Aquifermächtigkeit im Untersuchungsgebiet

# Index

- \*
  - \*.dac-Datei ..... XIII, 4, 30, 31, 34, 36, 38, 40, 46
  - \*.eco-Datei ..... 34, 35, 36
- A**
  - ACE ..... 1-X, 9
  - Alternative ..... 18, 19
  - Aquifer ..... X, XII, 1, 5, 21, 54
  - ASCII ..... X, 8, 13, 24, 30
  - auswählbar ..... 9
  - Autorisierung ..... 14
- B**
  - Bodenschätzdaten ..... 3, 7
  - BOMET.. X, XII, XIII, XIV, XV, XX, XXI, 3, 6, 7, 47
  - BOMET\_BS ..... XII, XIV, XX, 7
- C**
  - CASE ..... X, 9, 14, 16, 17, 18, 19, 20, 21, 36
  - CASE-Tool ..... 16, 18
- D**
  - Datenbank-Architektur ..... XII, 39
  - Datenmodell ..... 9, 11, 18, 19, 36, 37
  - Datenmodellierung ..... 20
  - Diskretisierung ..... 23
- E**
  - Editoren ..... 13, 14, 23, 37
  - Elemente ..... XIII, 7, 8, 23, 25, 32, 37, 38, 39, 40
  - Elternobjektklasse ..... 13, 21
  - EMACS ..... X, 9, 11, 12, 18, 21, 47, 48
  - Entity-Relationship-Modell ..... 17
  - Ephemeral-Speicher ..... 15, 16
  - ERM Lahmeyer International ..... 2
  - EXPERT-N ..... 7
- F**
  - Fachschalen ..... 20
  - fangbar ..... 9
  - FD ..... 1-X, 24
  - FE ..... X, 4, 7, 8, 23, 24, 30, 31, 38, 39
  - FEFLOW ..... X, XII, XIII, XIV, XV, XX, XXI, 2, 3, 4, 21, 22, 23, 24, 25, 27, 28, 29, 30, 34, 36, 38, 39, 40, 41, 42, 45, 48, 49, 52, 54
  - Feld ..... 17
  - Feldschätzbuch ..... 7
  - Finite Differenzen ..... 1-X, 23, 24, 25
  - Finite Elemente ..... X, XII, XIV, XX, 23, 43
  - Finite Volumina ..... X, 25
  - Finites Elemente Netz ..... 8, 39, 40
  - FRAC3DVS ..... X, 24
  - FV ..... X, 24
- G**
  - Garbage Collection ..... X, 16
  - Geodatenbank ..... XII, 2, 7
  - GNU ..... X, 12, 47, 48
  - graphical user interface ..... 4
  - Grundwasserneubildung ..... X, XII, 7
  - GUI ..... X, 4, 34, 39
- H**
  - Hauptalternative ..... 18, 21
  - HGEO ..... XI, XII, XIV, XX, 3, 8, 48
- I**
  - IHF-ISOLINE ..... 8
  - Image ..... 16
  - Inkonsistenz ..... 16
  - Instanz ..... 9, 12, 13, 21
  - interface ..... 45
  - Isolinien ..... 8
- K**
  - Kernel ..... 15
  - Kindobjektklasse ..... 13, 21
  - Konfiguration ..... 9
  - Konsistenz ..... 16
  - Koordinatensystem ..... 9
- L**
  - Layout ..... 9, 15, 52, 53
  - Level ..... 14, 50
  - LISP ..... 1-XI, 11, 12, 48
  - LIWIS ..... 1-XI, 2
- M**
  - MAGIK ..... IV, XII, XIII, XIV, XV, XXI, 9, 11, 12, 13, 14, 15, 16, 20, 21, 47
  - manage\_soc ..... 18
  - Materialparameter ..... 25, 54
  - Methoden ..... 12, 13, 14, 21, 24, 46, 48, 50

- 
- Modellintegration .....XI, XII, XIII, XV, 2, 24,  
26, 39, 45
- MODFLOW .....XI, 23, 25
- MODINT.....XI, XII, XIII, XIV, XV, XX,  
XXI, 2, 4, 8, 13, 20, 24, 26, 28, 29, 30, 31, 33,  
34, 36, 37, 38, 39, 40, 45, 46, 48
- Module ..... 2, 20, 40, 42
- module.def.....20
- Monitor.....31, 38, 39
- Multitasking ..... 13
- Multithreading..... 13
- N
- Nitratauswaschung .....7
- O
- Objektklasse ..... 12, 13, 14, 18
- Objektorientiert ..... 12
- OGRUT .....XI, XII, XIV, XX, 3, 7, 8, 28, 30, 31,  
38, 39, 48
- OOP.....XI
- Outputdatei ..... 4, 25, 34, 38
- Outputdateien ..... XIII, 4, 24, 38
- P
- Parser.....XII, XIII, XV, 4, 8, 30, 38
- Pedosphäre .....7
- Postprocessing.....24, 40
- pragma-statement ..... 14, 50
- Preprocessing ..... XIII, 24, 25, 30, 40
- Processing..... XI, XIII, 24, 25, 30, 34, 39, 40
- R
- Randbedingungen.....XII, XIII, 23, 24, 30
- Real World Object.....XI
- Record .....17
- Redundanz..... 17, 21
- Reichsbodenschätzung .....7
- Reverse Engineering.....18
- Richard Stallmann .....11
- RWO .....XI
- S
- Schlüssel.....17
- Schlüsselfelder ..... 17
- Schnittstelle ..... 7, 21, 24, 25, 28, 38, 40, 45
- Slots.....12, 13, 14, 32, 34
- Spatial Object Controler.....XI
- Spatial Object Manager.....XI
- statische Speicher ..... 15, 16
- Staufener Bucht..... 1, 2, XII, XX, 1, 3, 4, 5
- Stickstoffmodell ..... 7
- Stofftransportparameter.....24, 54
- Superelementen .....23
- SURFER.....8
- SUTRA.....24, 25
- T
- Tabelle.....14, 17, 18, 25, 32, 34, 37, 50, 54
- Template.....17
- Threads.....13, 21, 40
- Trinkwasser .....1
- U
- Unclassified.....50
- UWSP.....1-XI, 7
- V
- Vererbung.....13, 21
- Verknüpfungen.....17, 18
- Versionsverwaltung.....18, 21
- VIS\_TOOL.....XI, 7, 38, 39
- Visualisierung ..... XIII, 2, 3, 4, 8, 24, 36, 39
- W
- WAQIS.....XI, XII, XIII, XIV, XV, XX, XXI,  
1, 2, 3, 4, 21, 25, 28, 39, 40, 45, 48, 49
- Wasserwirtschaft.....1, 48
- WASY .....XI, 23, 25, 49, 54
- Z
- Zartener Becken .....XII, 1-XX, 1, 3, 5
- Zeitschritte.....XIII, 30
- Zurückentwickeln.....18

## **Ehrenwörtliche Erklärung**

Hiermit erkläre ich, dass ich diese Arbeit selbständig und nur unter Verwendung der angegebenen Hilfsmittel angefertigt habe.

Freiburg, den 20.12.2001

---

Thomas Graf

