DESIGN AND DEVELOPMENT OF A HYDROLOGICAL DECISION SUPPORT FRAMEWORK

Report to the
Water Research Commission

by

DJ Clark[1], JC Smithers[1], DA Hughes[2], KB Meier[3], MJ Summerton[3] and AJE Butler[4]

[1]School of Bioresources Engineering and Environmental Hydrology
University of KwaZulu-Natal
Pietermaritzburg, South Africa

[2]Institute for Water Research
Rhodes University
Grahamstown, South Africa

[3]Umgeni Water
Pietermaritzburg, South Africa

[4]Partners in Development
Pietermaritzburg, South Africa

WRC Report No 1490/1/09
ISBN 978-1-77005-811-8

April 2009

DISCLAIMER

EXECUTIVE SUMMARY

Introduction

The National Water Act (NWA, 1998) of South Africa (Act 36 of 1998) specifies that Government, as the public trustee of the nation's water resources, must ensure that water is protected, used, developed, conserved, managed and controlled in an equitable and sustainable manner for the benefit of all people. The NWA has introduced new paradigms and significant challenges in the management of water resources in South Africa. These include the provision of reserves to meet human and ecological requirements as primary priorities, the devolution of the responsibility for the management of water from a national to a regional level and the requirement that stakeholders participate in the management of water resources.

The NWA calls for the equitable, efficient and sustainable allocation of water resources. This will require decision support at temporal and spatial scales appropriate to the demands of the NWA and which utilises state-of-the-art hydrology, in addition to social, economic and environmental needs which have to be considered. A process-based hydrological model, which can have meaningful links with socio-economic models, is a logical framework on which to build the Decision Support System (DSS).

The NWA requires that the Department of Water Affairs and Forestry (DWAF), as custodian of South Africa's water resources, should provide a National Water Resource Strategy (NWRS) as a framework for the management of water resources in South Africa. The draft version of the NWRS provides the implementation framework for the NWA and identifies 19 Water Management Areas (WMAs) to be managed by Catchment Management Agencies (CMAs).

It was stated in Clark and Smithers (2006) that as the demand on South Africa's water resources increases it is necessary for water to be managed and utilised with greater efficiency. The water management paradigm has changed from a focus on meeting demands from surface water supplies to a more holistic approach which also considers other aspects such as water quality, the environment as a water user, surface water-groundwater interactions, and social and economic impacts There has been an international trend in recent years towards this new water management paradigm known as Integrated Water Resources Management (IWRM) (Havnø et al., 2002; Schulze, 2002). In South Africa this new paradigm has been captured in Section 3 of the National Water Act (NWA, 1998) of South Africa (Act 36 of 1998) as follows:

"*3. (1) As the public trustee of the nation's water resources the National Government, acting through the Minister, must ensure that water is protected, used, developed, conserved, managed and controlled in a sustainable and equitable manner, for the benefit of all persons and in accordance with its constitutional mandate.*

(2) Without limiting subsection (1), the Minister is ultimately responsible to ensure that water is allocated equitably and used beneficially in the public interest, while promoting environmental values.

(3) The National Government, acting through the Minister, has the power to regulate the use, flow and control of all water in the Republic." (NWA, 1998).

As explained in Clark and Smithers (2006), IWRM requires a greater understanding of complex hydrological systems. Since the 1980s advances in computer science combined with improved understanding of hydrological processes has resulted in the development of computer simulation models that encapsulate understanding of hydrological systems. However, many of the models were developed for a particular domain within the hydrological system, such as surface water, groundwater or ecology, by experts in that domain. IWRM requires that the models for each domain be integrated and that scientific experts collaborate in the holistic assessment and management of water resources.

As discussed in Clark and Smithers (2006), the need to integrate water resources models has led to the recognition of the need for and development of modelling frameworks to facilitate integrated water resources modelling. Traditionally models have each been run within their own modelling systems, each consisting of similar tools to prepare model input data, write model input files and analyse model output. Modelling frameworks seek to reduce duplication of effort and facilitate model use and development by providing common data preparation and post-processing tools, including a Geographic Information System (GIS) and database management, for use with several models. From the literature reviewed it is apparent that there is a growing realisation of the need for hydrological modelling frameworks and that there are a number of different frameworks under development internationally (HarmonIT, 2002; Marston et al., 2002; Argent and Rizzoli, 2004).

The aims of the project, as stated in the contract were to:

- Provide hydrological decision support for the implementation of the NWA (1998)
- by identifying a suite of suitable and appropriate modules for use by the CMAs
- the linking/integration of selected modules within a Hydrological Decision Support Framework (HDSF), which could include:
  - a generic user interface
  - integrated GIS
  - generic, extensible spatial and temporal databases with seamless input/output to/from models
- The HDSF should be structured to enable inclusion of additional modules
- Build capacity through the implementation of the project.

The aims of the project were translated into the following objectives:

- The primary objective was the development of a Hydrological Decision Support Framework (HDSF) and integration of relevant modules into framework
- The second objective was to extend the capabilities of the some of the modules so as to enable the assessment of water resources and the allocation of water use licences at the level of CMA and to consolidate and encapsulate existing relevant research findings into the selected simulation models
- The third objective was to provide user support and documentation.

The primary objective of this project was the development of a Hydrological Decision Support Framework (HDSF) which can incorporate relevant and appropriate simulation models linked by a common flexible and extensible database and integrated with a GIS for use at a planning and operational level by Catchment Management Agencies (CMAs) at spatial scales ranging from point of use to the entire Water Management Areas (WMA) and at temporal time scales of one day. The first step towards this objective was to identify the requirements for the HDSF. Several existing hydrological modelling frameworks and framework development tools were reviewed and evaluated relative to the identified requirements. Based on this evaluation, it was decided not to develop a new framework and one of the existing hydrological modelling frameworks was selected for adoption and further development within the project to form the foundation of the HDSF. In addition, a framework for linking hydrological models was selected for evaluation within the project. The project then entered a design phase where enhancements to the selected hydrological modelling framework and additional modelling tools were conceptualised. The implementation phase of the HDSF then took place and resulted in the HDSF software presented in this report.

A secondary objective of this project was to provide for the modelling requirements of Catchment Management Agencies (CMAs) for the assessment and management of available water resources and their dynamics within a catchment by extending the capabilities of the some of the modules well as consolidating and encapsulating existing relevant research findings into selected simulation modules in order to refine the simulation of hydrological processes. It was recognised that no single model would be suitable to meet all the modelling requirements necessary for IWRM and that also the HDSF should not be restricted to a fixed set of models, but rather provide a framework within which any compatible model could be run. It was decided that a physical conceptual model operating at a daily time-step would be necessary to provide for the modelling requirements of CMAs and should be included in the HDSF as part of this project.

An additional objective of this project was to compile a collection of readily available datasets that would be useful to users of the HDSF. These data sets would include both spatial and temporal data relevant to hydrological modelling.

Framework Development

The framework development component of the HDSF project started with the identification of user requirements. Six modelling frameworks (BASINS, SPATSIM, WR IMS, OMS, TIME and OpenMI) were reviewed in detail to determine their suitability for use in the project. These modelling frameworks reviewed were evaluated against a list of requirements identified by a user survey and project team for the HDSF. The requirements for the main components of the HDSF were outlined at follows:

- **Data Model:** The HDSF requires a well designed extensible data model to enable storage of both spatial and temporal data. The data model should facilitate the storage of model input and output data. It should be possible to create a separate database for each modelling project and for databases to be archived. The data model should be designed for the purpose of providing data to and receiving data from simulation models and not as a central repository for national datasets. Tools should be provided in the HDSF to import data from national databases.
- **GIS Tools:** GIS will be an important tool for the setting up of model input data and information and for the analysis of model output data. The HDSF should contain a set of basic GIS tools facilitating the configuration of catchments and flow networks, and for viewing spatial model input and output data. These will be referred to as Internal GIS Tools. The HDSF should facilitate exchange of data with commercial GIS systems and possibly provide HDSF specific tools to be used within these commercial GIS systems. These will be referred to as External GIS Tools. The HDSF should be based on the most widely used GIS data formats.

- **Model Input:** The HDSF should have a user friendly GUI allowing model input data to be entered and reviewed with facility to do range and logic checking of data entered. Ideally the HDSF should provide an extensible generic GUI suitable for all models. However, provision should be made for model specific GUIs to be used as these often contain complex data checks.
- **Data Analysis:** The HDSF should include a comprehensive and easy to use set of tools for analysis and display of model output. These tools should include the graphing and comparison of time-series data and associated statistical analyses.
- **Models:** The HDSF should allow new models to be added in as simple a manner as possible, although it is recognised that the addition of a new model would only be done by the developers of the HDSF, or suitably qualified experts, and not by every user of the HDSF. It should be possible to add legacy models to the HDSF and the addition of these legacy models should require the minimum of code changes. The links between models and other components of the HDSF such as the graphical user interface (GUI), GIS tools and database should be as extensible as possible to allow for changes to models. The HDSF should include facilities for the linking of models in series and parallel.
- **General:** The HDSF should include a system of managing modelling projects and different scenarios within a project, and the archiving of projects. The HDSF is intended to run on the Windows operating platform.
- **Help:** The HDSF should include a system of help files explaining the use of the HDSF and each of the models that may be run from the HDSF.

The SPATSIM modelling framework (Hughes, 2002; Hughes 2005) was selected to form the core of the HDSF as it was locally developed, included a generic extensible database structure for the storage of both spatial and time-series data, included both GIS and data analysis tools and was suited to further development. The SPATSIM modelling framework has been further developed to meet the requirements of the HDSF and this new version is referred to as SPATSIM-HDSF.

The design phase of the project started with the design of a general framework and required the conceptualisation of the framework components and data flows, as shown in Figure 1.

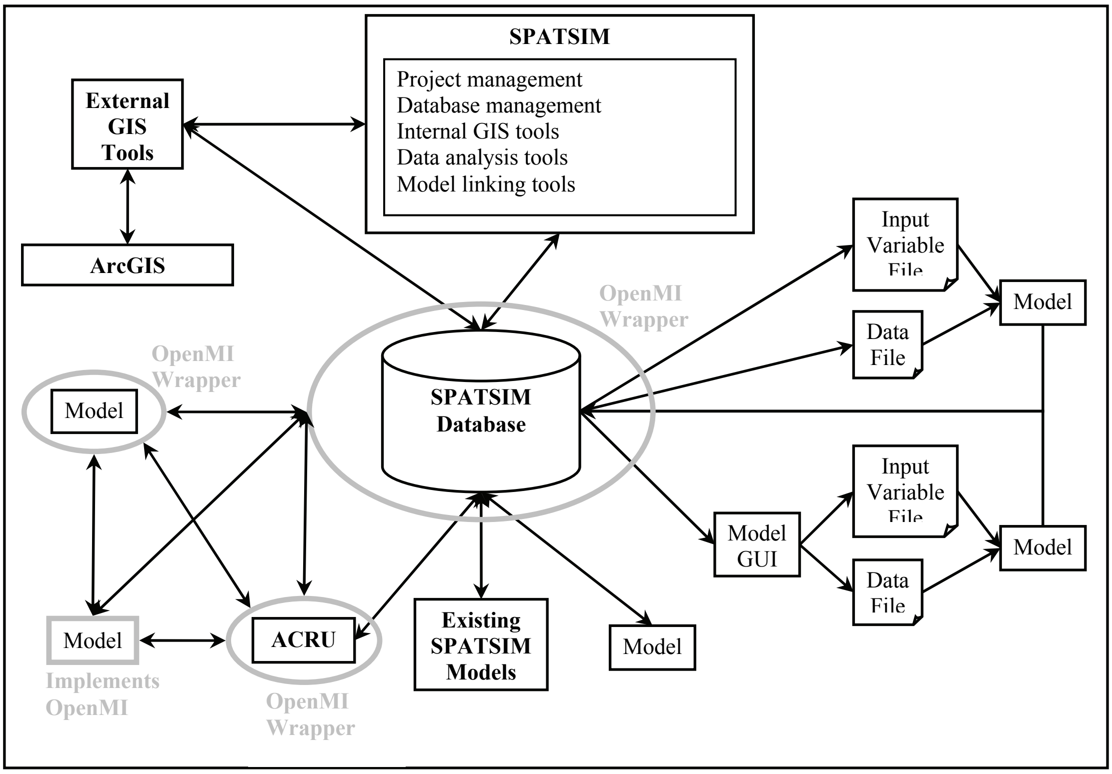


Figure 1: Components and data flows for the proposed HDSF

One of the most important criteria when selecting a framework for use in the HDSF project was the data model used by the framework. The design of the data model used in SPATSIM was based on the approach adopted by the Centre for Ecology and Hydrology (CEH) at Wallingford, United Kingdom (Hughes, 2002). One of the reasons for selecting SPATSIM was the data model used. The SPATSIM database data model was reviewed and extended taking into consideration the requirements for the HDSF. In SPATSIM spatial features are represented by geographic data stored in ESRI shape files and a related set of attributes stored in a SPATSIM database. The link between geographic data and attribute data is achieved using four feature reference tables. Attribute values are stored in a set of attribute value tables based on their attribute data type. A table is used to store information about process (model) runs. In this project the SPATSIM database data model was extended to include:

- provision for non-spatial features
- grouping of attributes into attribute groups
- storage of metadata for features, attributes and attribute values
- a more comprehensive system of units of measure that may be applied to all relevant attribute data types
- storage of relationships between the components of the hydrological system represented by features and objects
- storage of time-series data quality flags
- storage of attributes and attribute values for processes
- storage of information used by the internal and external GIS tools.

An important part of the implementation of the new SPATSIM-HDSF database data model was the development of a set of classes containing standard SPATSIM-HDSF project database access methods to create, populate, edit and query SPATSIM-HDSF databases. These SPATSIM-HDSF project database access methods will simplify database access not only in SPATSIM-HDSF, but also in the models modified to run from within the SPATSIM-HDSF framework. These classes were initially developed in the Delphi for .Net programming language and are packaged as a Dynamic Link Library (DLL) which can be used in code written in any .Net programming language. However, there are many legacy models written in a variety of programming languages that would not be able to utilise these .Net compliant classes, Therefore, a similar set of Component Object Model (COM) accessible classes was developed. Further to this, there may be models such as the *ACRU* model which are written in the Java programming language, and for this reason, a Java version of the data access classes was created. As far as possible these database access classes are based on standard Structured Query Language (SQL) queries to facilitate possible use with a selection of relational database management systems (RDBMSs).

The purpose of the SPATSIM-HDSF user interface software is to provide a graphical user interface for SPATSIM-HDSF users to add, edit, delete, view and analyse data in a SPATSIM-HDSF database. The SPATSIM-HDSF user interface software also enables users to setup and run SPATSIM-HDSF compliant models using data in a SPATSIM-HDSF database. Further development of the SPATSIM user interface during this project included the following:

- conversion of the code from Delphi to the Delphi for .Net programming language
- conversion of the user interface forms and controls from Delphi's Visual Component Library (VCL) components to standard .Net Windows Forms (WinForms) components
- conversion to the use of the ADO.Net database components and OleDB data providers in place of the Borland Database Engine (BDE) data components used previously
- reorganisation of the main form which includes the following components: menu, toolbar, feature selection control, attribute selection control, *Map* tabbed page, *Data* tabbed page and status bar
- redesign of the feature and attribute selection controls as treeview type controls for easier selection of features, object attribute groups and attributes
- development of the *Data* tabbed page to display a data grid showing attribute values for objects in a selected feature and attribute group
- development of forms to enter and edit feature, attribute and attribute value metadata
- development of a new set of feature attribute value editing forms, one form for each of the attribute data types
- development of a set of process attribute value editing forms, one form for each of the attribute data types
- further development of the internal GIS tools to include functionality to setup themes for a spatial feature, use of these themes for rendering based on feature attribute values and display of these themes in a legend display control
- the development of a consolidated set of time-series analysis tools including graphing, statistics, comparative statistics and frequency analysis tools based on the external TSOFT application previously accessed from SPATSIM and the Microsoft Excel based *ACRUView* utility for the *ACRU* model.

An important component of the HDSF development during this project has been the development of the HDSF External GIS tools. The HDSF External GIS tools have been developed as an extension to ArcGIS and provide an important link between SPATSIM and ArcGIS enabling SPATSIM user access to the functionality built into a commercial GIS package for complex editing, analysis and manipulation of geographical data. The HDSF External GIS extension includes:

- a toolbar with menu and button options
- forms to catch and display error messages which may arise during the execution of the code
- database connection classes to access and query information in a SPATSIM-HDSF database
- selection and loading of existing SPATSIM-HDSF projects, where the geographical data pertaining to the selected SPATSIM-HDSF project will be loaded into the ArcGIS environment and appropriate symbology and labelling will be applied to the GIS coverages
- analysis of Gridded GIS information based on a specified SPATSIM-HDSF coverage so that feature attributes in a SPATSIM-HDSF database can be populated automatically
- area-weighting of information in one coverage relative to a second coverage.

The OpenMI model linking framework (HarmonIT, 2005a, HarmonIT, 2005b) was investigated as a means of linking models in parallel (i.e. enable models to interact on a time step by time step basis) and as a means of linking models to a data source, such as SPATSIM-HDSF database. The purpose of this investigation was to develop a simple proof-of-concept example of the use of OpenMI. The proof-of-concept example aimed to: link two simple models written in C#, link two simple models running at two different time-steps and link a simple model to a simple data source using an OpenMI wrapper for the data source. Further to this a recommendation was sought regarding the feasibility of making a Java model OpenMI compliant and being able to link this model to an OpenMI compliant model written in a .Net programming language. For the proof-of-concept example it was assumed that the simple models already existed and therefore the aim was to demonstrate wrapping of an existing model, as opposed to the development of a new model that implemented the OpenMI Standard interface directly. The two models to be linked were in the form of a simple rainfall-runoff model and a simple flow routing model. The proof-of-concept example was completed in the form of three C# code sets, one containing the simple rainfall-runoff model, one containing the simple flow routing model and the third containing a simple data source. Each of the code sets contained an OpenMI compliant wrapper enabling the relevant model or data source to be used with other OpenMI Linkable Components in a model run. The proof-of-concept was successfully developed and there is no reason why more complex models written in a .Net programming language should not be made OpenMI compliant. The complexity of the process of wrapping a model to make it OpenMI compliant will depend on the architecture of the model. A model written in Java could be wrapped in a similar manner to make it OpenMI compliant, however, interoperability between the .Net platform and the Java platform proved to problematic and requires a more detailed investigation.

Model Development

Although model development was a secondary objective of the project, significant model development work was completed. Four of the most commonly used models within SPATSIM, namely the Pitman model, Pitman model for parameter exploration, Desktop Reserve model and Stressor-Response model, were converted to work with the new SPATSIM-HDSF database data model. In addition, the *ACRU* model was modified to run from within the SPATSIM-HDSF modelling framework. Modifications to *ACRU* to run from SPATSIM-HDSF included the development of a XML based model input data structure and model configuration files. These model input data and model configuration files have been designed such that the same file design could be used for other hydrological models. A prototype model configuration editor was developed for the *ACRU* model, but was also designed such that it could potentially be used for other models if a suitable configuration schema file is created for each model.

Several hydrological processes were incorporated into the *ACRU* model as part of this component of the project. Research versions of three modules were consolidated and incorporated into *ACRU*, namely the dam and river operating rule module (Butler, 2001), the *ACRUSalinity* module (Teweldebrhan, 2003) and the *ACRUCane* module (Moult, 2005). The dam and river operating rule module aims to model the supply of water to general water users taking into account the basic human needs reserve and ecological reserve requirements of the National Water Act (NWA, 1998) of South Africa (Act 36 of 1998). This framework has some limitations with regard to its use for operational modelling, therefore an investigation was conducted to determine the feasibility of: (i) modelling water supply to a user from more than one water source, (ii) accounting for lags and attenuations associated with flow releases made from dams, (iii) updating the dam operating rules related to meet the ecological reserve, and (iv) using the framework as an operational tool in the management of water resources in South Africa. The *ACRUSalinity* module was further developed in collaboration with WRC project K5/1301. More detailed soil surface processes were added to *ACRUSalinity* to improve modelling of salinity at the soil surface and to implement rules enabling the abstraction of water from dam and river water sources to be modelled in an improved operational manner based on water availability and salinity levels.

Support to users was provided by the project team for the *ACRU* model as well as for other data and software related queries. This support was provided to several research institutions, consultants, DWAF and other WRC projects.

Databases

One of the objectives of this project has been to compile a collection of readily available datasets in order to support the use of the SPATSIM-HDSF modelling framework and the models that run within it. The objective was not to develop a national database of hydrology in South Africa, as this was beyond the scope of this project, but to rather package with the SPATSIM-HDSF modelling framework relevant datasets which are available from the School of BEEH including products generated by other WRC projects carried out by the School of BEEH. The data provided by this project consist of two main components: a database of climate related time-series data per quaternary catchment, and the *ACRU* Quaternary Catchments Database (AQCD).

The climate time-series database consists of daily time-series of rainfall, maximum temperature, minimum temperature, solar radiation, maximum relative humidity, minimum relative humidity and Penman-Monteith reference crop evaporation for each of the 1946 Quaternary Catchments in South Africa. Each time-series contains 50 years of daily data for the period from 1950 to 1999. Obtaining or generating quality controlled time-series of climate related data is often a difficult and time consuming component of any modelling project. It is expected that this database of time-series data and information will be widely adopted by users of the SPATSIM-HDSF modelling framework. These data has been packaged in two forms: in a SPATSIM-HDSF database for ease of use by SPATSIM-HDSF users and as *ACRU* Composite (Y2K) Input Data files.

The objective of the *ACRU* Quaternary Catchments Database (AQCD) is to provide a pre-populated database containing a baseline set of inputs for the *ACRU* model at a Quaternary Catchment scale. As stated in Schulze et al. (2007), the AQCD enables the use of the *ACRU* model to perform simulations of, for example,

- stormflow
- baseflow, or
- total runoff, as well as
- impacts of land use change on hydrological responses, or of
- climate change on hydrological responses, of
- crop yields
- sediment yield
- irrigation water demand, or
- hydrological risk analyses

at a spatial scale of either

- individual Quaternary Catchments or, in the case of streamflows, for
- hydrologically linked, cascading catchments, thereby facilitating the simulation of cumulated flows from a series of individual Quaternary Catchments that make up a larger area.

It is this baseline set of *ACRU* model inputs for each Quaternary Catchment that has been stored in a SPATSIM-HDSF database thus providing a populated baseline AQCD for users of the SPATSIM-HDSF modelling framework. Although the AQCD is intended primarily for use with the *ACRU* model, many of the variables, which are fully described in Schulze (1995) and Smithers and Schulze (2004), are variables that may be used in other hydrological models. The AQCD has been packaged in two forms: in a SPATSIM-HDSF database for ease of use by SPATSIM-HDSF users and as an *ACRU* 300 series version text based input data file.

Case Study

A case study was conducted in which the SPATSIM-HDSF modelling framework was applied to a selected catchment as a demonstration of the usefulness of the SPATSIM-HDSF modelling framework. The project concentrated on just one case study to demonstrate the use of the modelling framework in order to test the improvements made by the HDSF project to configure and model the hydrology in a catchment rather than to verify hydrological models on several different catchments. The Mgeni System was selected to leverage off work done concurrently by Umgeni Water (Summerton and Gillham, 2007, and Summerton, 2008) who were also part of the project team. The configuration of the *ACRU* model for the Mgeni System was updated in the study by Summerton (2008), and it was a subset of this configuration of the Mgeni System that was used for the case study. The case study included an assessment of the input utilities, ability to run models and time-series analysis utilities in SPATSIM-HDSF. Feedback regarding minor problems encountered during the case study resulted in several improvements to SPATSIM-HDSF One of the outcomes of the case study was a set of recommendations for further development.

In addition SPATSIM-HDSF was demonstrated at a workshop to inform existing SPATSIM users and other interested people of the capabilities of the SPATSIM-HDSF modelling framework.

Discussion, Conclusions and Recommendations

Integrated water resources management is a key requirement of the National Water Act (NWA, 1998) of South Africa (Act 36 of 1998) and this has created a requirement for integrated water resources modelling where a set of domain specific models need to be integrated to represent all facets of the hydrological system. The purpose of hydrological modelling frameworks is to provide a modelling environment facilitating the integration of models, providing a common data model and common data editing and analysis tools.

A user requirements survey was conducted prior to the inception workshop to identify general user needs. The framework related responses received indicated the need for a GIS interface for model configuration, a well designed and extensible database, user friendly graphical interfaces, tools for post-processing and display of model output, the ability to incorporate legacy models in the framework and to link and run models in parallel, i.e. on a time step by time step basis. A number of these requirements were either fully or partially addressed in this project.

Several national and international modelling frameworks were reviewed and the SPATSIM modelling framework was selected for further development. A comprehensive framework design was undertaken which included the design of the data model, GIS tools, framework interfaces, data analysis tools, project management tools and help facilities.

This software development undertaken in this project has resulted in a significantly enhanced modelling framework with regard to the database data model, data editing tools, internal GIS tools and time-series analysis tools. The SPATSIM database data model was expanded to include: non-spatialfeatures, attribute groups, storage of metadata; a more comprehensive system of units of measure, storage of relationships between feature objects, storage of time-series data quality flags, storage of attributes and attribute values for processes and the storage of information used by the internal and external GIS tools. The data editing tools include a set of newly designed attribute value editing forms and the new tabbed page, which enables text, ,integer and real type attribute values for all the objects in a selected feature and attribute group to be viewed as a data grid. Enhancements to the internal GIS tools include: functionality to setup themes for a spatial feature, use of these themes for rendering based on feature attribute values and displaying these themes in a legend display control. A consolidated set of time-series analysis tools based on the external TSOFT application previously accessed from SPATSIM and the Microsoft Excel based *ACRUView* utility for the *ACRU* model includes graphing, statistics, comparative statistics and frequency analysis tools.

In addition, the conversion of the code to .Net programming platform has ensured that future development of SPATSIM-HDSF can take advantage of recent developments in software development technology. The use of the ADO.Net database access framework should enable SPATSIM-HDSF databases to be implemented in RDBMSs other than Microsoft Access, which was used in this project. An important component of the HDSF is the external GIS tools developed in this project as, in addition to the analysis tools, it includes an interface between ArcGIS and SPATSIM-HDSF which will be extremely useful for model configuration and analysing model output.

The ability to link models in parallel will enable two or more expert models to be integrated without needing to integrate the code for these models. The OpenMI model linking framework (Gijsbers, 2003; Blind and Gregersen, 2004; HarmonIT, 2005a) was developed in the European Commission funded HarmonIT project for this purpose. An initial investigation of OpenMI conducted as part of the HDSF project concluded that the adoption of OpenMI would have many potential benefits. Models running on the same operating platform (e.g. .Net, or Java) may be linked using OpenMI. However, this investigation showed that the interoperability between the .Net platform and the Java platform is difficult, even though technologies are available that may make this possible. This has significant implications for the *ACRU* model which is written in Java. Performance of models linked via OpenMI is another issue that needs to be further investigated.

The *ACRU* model was successfully implemented as a model that may be run within the SPATSIM-HDSF modelling framework. The ability to implement a complex physical conceptual model such as *ACRU* within the SPATSIM-HDSF demonstrates the robustness of the design of the SPATSIM-HDSF modelling framework. As part of this *ACRU* implementation, XML based model input data and model configuration files, and a prototype Configuration Editor for population of model input data files, were developed. These were primarily developed for the *ACRU* model, but have been designed in such a way that they could easily be applied to other hydrological models. During this process new technologies and concepts were learned which will be useful in the further development of the SPATSIM-HDSF modelling framework.

Although process modelling was a secondary objective of the project, significant consolidation and further development of hydrological processes in the *ACRU* model was achieved. These processes include aspects of water management, operational modelling, water quality, irrigation efficiency and crop water use, all of which are important components in the implementation of the National Water Act (NWA, 1998).

The dam and river operating rule module developed for the *ACRU* model by Butler (2001) was an important step towards providing the modelling capability in the *ACRU* model for modelling the Ecological Reserve on a daily basis. Therefore the consolidation of the Operating Rule module within the most recent version of the *ACRU* model was important. However, several limitations were recognised and an investigation was conducted into the feasibility of extending the Operating Rule module to facilitate multiple water sources per water user, account for flow lags and attenuations when supplying water, improve modelling of the Ecological Reserve and be suitable for use as an operation tool for management of water resources. This investigation included a review of the techniques used in the WRYM, Mike Basin and a model developed by Mallory and Van Vuuren (2007) and concluded that a rule-based system should be used and that it would be feasible to implement such a system in the *ACRU* model.

Implementation of the National Water Act (NWA, 1998) requires that water quality also be taken into account in the management of water resources. The *ACRU* model includes the *ACRU_NP* nitrogen and phosphorous module and as part of this project the *ACRUSalinity* module developed for *ACRU* by Teweldebrhan (2003) was consolidated within the most recent version of the *ACRU* model. The consolidated *ACRUSalinity* module was then used in WRC Project K5/1301 and, at the request of this project, several enhancements were made to the *ACRUSalinity* module. These include improved modelling of salts at the soil surface and the implementation of rules enabling the abstraction of water from dam and river water sources to be modelled in an improved operational manner based on water availability and salinity levels.

The *ACRUCane* module developed by Moult (2005) was also consolidated within the most recent version of the *ACRU* model. Irrigation is one of the dominant users of water resources in South Africa and the *ACRUCane* module is an important step towards being able to model in more detail the link between hydrology, irrigation scheduling, irrigation efficiency and crop yield. The *ACRUCane* module is based on the FAO56 method and although focussed on sugarcane, could be easily extended to model other irrigated crops.

At the conclusion of this project both the SPATSIM-HDSF and the *ACRU* software are at a stage where significant development has taken place and new designs and concepts have been proven. However, both need to be thoroughly beta tested before being released for general use. It is anticipated that as the new version of SPATSIM-HDSF is used there will be requests and suggestions from users for further development of additional functionality, such as time-series analysis and GIS related tools.

Some recommendations for possible further development of the SPATSIM-HDSF modelling framework include:

- further development of the external GIS tools to enable catchment delineation and the setting up of linkages/relationships between features
- further development of the internal GIS tools to enable the setting up of linkages/relationships between features and rendering based on array attribute elements
- improvement of the speed of display of time-series data in the attribute data editing forms
- development of additional time-series analysis tools including: extreme value analysis, run length analysis and time-series transformation tools
- modification of time-series analysis tools so that graph legends are displayed on a separate control that the user can choose to hide or display
- providing an option to access time-series files in dBase (.dbf) format from the time-series analysis tools
- development of tools to import data from one SPATSIM-HDSF database to another SPATSIM-HDSF database
- development of tools to import and export other data formats
- creation of an OpenMI wrapper around the SPATSIM-HDSF database.

Some recommendations for possible further development of the *ACRU* model and related tools for the SPATSIM-HDSF modelling framework include:

- storage of outputs from the *ACRU* model to the SPATSIM-HDSF database
- linking data to the GIS when importing *ACRU* input "menus" using the *ACRU* Menu Converter tool in SPATSIM-HDSF
- conversion of other *ACRU* input data formats including *ACRU*2000 and CSV files using the *ACRU* Menu Converter tool
- development of a tool to enable *ACRU* users to select variables to be included in the model output, where these variables are to be stored and in which format
- development of tools to setup a base configuration using readily available data by accessing GIS type tools from the SPATSIM-HDSF directly
- development of a full working version of the *ACRU* Configuration Editor
- implementation of OpenMI in *ACRU*.

An important outcome of this project has been the collaboration of two key hydrological modelling groups in South Africa, namely the School of Bioresources Engineering and Environmental Hydrology at the University of KwaZulu-Natal and the Institute for Water

Research at Rhodes University. The collaboration between these modelling groups will impact positively on the sustainability of detailed hydrological modelling in South Africa.

References

Argent, RM and Rizzoli, AE. 2004. Development of Multi-Framework Model Components. Transactions of the 2nd Biennial Meeting of the International Environmental Modelling and Software Society, iEMSs 2004: Osnabrück, Germany.

Blind, M and Gregersen, JB. 2004. Towards an Open Modelling Interface (OpenMI): The HarmonIT Project. Transactions of the 2nd Biennial Meeting of the International Environmental Modelling and Software Society, iEMSs 2004. Osnabrück, Germany.

Butler, AJE. 2001. The development and evaluation of an operating rule framework to satisfy the requirements of the National Water Act. Unpublished MSc Engineering dissertation. Department of Agricultural Engineering, University of Natal, Pietermaritzburg, RSA. 146 pp.

Clark, DJ and Smithers, JC. 2006. Hydrological Decision Support Framework (HDSF) - Initial Design. Water SA, Vol. 32, No.4, October 2006, 465-472. Water Research Commission, Pretoria, South Africa.

Gijsbers, PJA. 2003. OpenMI – Harmonizing linkages between water related models. [Internet] Presented at Integrated Modelling User Group (IMUG) 2003 - International Conference on Application of Integrated Modelling, Tillburg Netherlands. Available from: http://www.wrcplc.co.uk/imug/html/imug_2003.htm [Accessed: 12 March, 2005].

HarmonIT. 2002. State of the Art Review. Document version: 9. Hutchings C. (ed.) [Internet]. Document produced for the HarmonIT project and available to members on registration on the HarmonIT website: www.harmonit.org [Accessed: 26 January 2004].

HarmonIT. 2005a. The OpenMI Architecture - Report A: Scope for the OpenMI. Document version: 1.0. Editor: Tindall I. [Internet]. Document produced for the HarmonIT project and available to members on registration on the HarmonIT website: www.harmonit.org [Accessed: 20 September 2005]

HarmonIT. 2005b. The OpenMI Document Series Part C – The org.OpenMI.Standard interface specification. Document version: 1.0. Editor: Gijsbers P J A. [Internet]. Document produced for the HarmonIT project and available to registered members on the HarmonIT website: www.harmonit.org [Accessed: 15 July 2005]

Havnø, K, Sørensen, HR and Gregersen, JB. 2002. Integrated water resources modelling and object oriented code architecture. Proceedings of the 1st Asia-Pacific DHI Software Conference, Bangkok, Thailand.

Hughes, DA. 2002. The development of an information modelling system for regional water resource assessments. Proceedings of the 4th International FRIEND Conference. IAHS Publication 274.2002, 43-49.

Hughes, DA (ed.). 2005. SPATSIM, an Integrating Framework for Ecological Reserve Determination and Implementation: Incorporating Water Quality and Quantity Components for Rivers. WRC Report TT 245/04, Water Research Commission, Pretoria, South Africa.

Mallory, SJL and Van Vuuren, SJ. 2007. Water Resource Modelling of Complex Catchments: The Development of a Rule-based Modelling Technique for South African Conditions. Unpublished paper. Department of Civil Engineering, University of Pretoria, RSA.

Marston, F, Argent, R, Vertessy, R, Cuddy, S and Rahman, J. 2002. The Status of Catchment Modelling in Australia. Technical Report, Report 02/4. Cooperative Research Centre for Catchment Hydrology (CRCCH), Australia. Available from: http://www.toolkit.net.au [Accessed: 2 February 2005].

Moult, NG. 2005. The development of a catchment scale irrigation systems model for sugarcane. MSc Eng. Dissertation. School of Bioresources Engineering and Environmental Hydrology, School of Bioresources Engineering and Environmental Hydrology, University of KwaZulu-Natal, Pietermaritzburg, South Africa.

NWA. 1998. National Water Act, Act No. 36 of 1998. Government Printer, Pretoria, RSA.

Schulze, RE. 1995. *Hydrology and Agrohydrology: A Text to Accompany the ACRU 3.00 Agrohydrological Modelling System*. Water Research Commission, Pretoria, RSA, Report TT 69/9/95. pp 552.

Schulze, RE (ed.). 2002. Modelling as a Tool in Integrated Water Resources Management: Conceptual Issues and Case Study Applications. WRC Report No 749/1/04. Water Research Commission, Pretoria, RSA.

Schulze, RE, Hallowes, LA, Horan, MJC, Lumsden, TG, Pike, A, Thornton-Dibb, S and Warburton, ML. 2007. South African Quaternary Catchments Database. *In:* Schulze, R.E. (Ed). 2007. *South African Atlas of Climatology and Agrohydrology*. Water Research Commission, Pretoria, RSA, WRC Report 1489/1/06, Section 2.3.

Smithers, JC and Schulze, RE. 2004. *ACRU Agrohydrological Modelling System*: *User Manual Version 4.00*. School of Bioresources Engineering and Environmental Hydrology, University of KwaZulu-Natal, Pietermaritzburg, RSA.

Summerton, MJ. 2008. A Preliminary Assessment of the Impact of Climate Change on the Water Resources of the Mgeni Catchment, Umgeni Water, Planning Services, Report Number 160.8/R001/2008 (DRAFT).

Summerton, MJ and Gillham, SW. 2007. A Water Utility's approach to addressing the potential impacts of Climate Change. In: Proceedings of the Thirteenth South African National Hydrological Symposium. Cape Town, South Africa.

Teweldebrhan, AT. 2003. The Hydrosalinity Module of *ACRU* Agrohydrological Modelling System (*ACRUSalinity*): Module Development and Evaluation. Unpublished MSc Dissertation, School of Bioresources Engineering and Environmental Hydrology, University of Natal, Pietermaritzburg, RSA.

ACKNOWLEDGEMENTS

The research in this report is the result of a project funded by the Water Research Commission (WRC) and the WRC is acknowledged gratefully for this funding.

The authors acknowledge:

The members of the Reference Group for this project for their contribution during the project:

Dr R Dube	Water Research Commission (Chairman)
Prof B Kelbe	Dept of Hydrology, University of Zululand
Mr E Lillie	Consultant
Ms F Makhubu	Department of Water Affairs and Forestry
Prof PJT Roberts	formerly of Forestry South Africa
Dr E Theron	School of Civil Engineering, Central University of Technology
Mr C Tylcoat	Department of Water Affairs and Forestry
Mr JA van Rooyen	Department of Water Affairs and Forestry
Ms T Zokufa	Department of Water Affairs and Forestry

Mrs W Gush for taking minutes at the Reference Group meetings.

The following for their contribution to the report: G Page-Wood, SA Lorentz, SLC Thornton-Dibb, A Lutchminarain, NL Lecler and CPH Water.

Mr AJE Butler and Mr G Page-Wood for their contribution to the project as subcontractors.

The delegates at the inception workshop and framework selection workshop for their comments and guidance.

The staff in the School of Bioresources Engineering and Environmental Hydrology (BEEH) for their work on the project and general assistance.

The University of KwaZulu-Natal (UKZN) for provision of office space, administrative support and computing equipment to project team members working in the School of BEEH.

The members of the Project Team for the work that they contributed to the project:

Prof JC Smithers (Project Leader)	School of BEEH, UKZN
Prof RE Schulze	School of BEEH, UKZN
Prof GPW Jewitt	School of BEEH, UKZN
Mr DJ Clark	School of BEEH, UKZN
Mr SL Thornton-Dibb	School of BEEH, UKZN
Mr A Lutchminarain	School of BEEH, UKZN
Prof DA Hughes	Institute for Water Research, Rhodes University
Mr DA Forsyth	Institute for Water Research, Rhodes University
Mr KB Meier	Umgeni Water
Mr MJ Summerton	Umgeni Water
Dr NL Lecler	SA Sugarcane Research Institute, SASA
Mr AJ Pott	CPH Water

Mr JS Hallowes	CPH Water
Prof AHM Görgens	Ninham Shand
Mr M Watson	Department of Water Affairs and Forestry
Mr B Haasbroek	Department of Water Affairs and Forestry

Some portions of this report have been based on a paper titled "Hydrological Decision Support Framework (HDSF) - Initial Design" published in Water SA (Vol. 32, No.4, October 2006). These sections have been enclosed in quotes where large portions of this paper have been reproduced verbatim in this report, and are referenced to Clark and Smithers (2006). This acknowledgement is intended to cover smaller sections of the paper that have been reproduced or paraphrased but not referenced, so as not to detract from the readability of this report.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

AAHMS	*ACRU* Agrohydrological Modelling System
AAMG	Amalgamation of Agro-Hydrological Modelling Groups
AQCD	*ACRU* Quaternary Catchments Database
BDE	Borland Database Engine
BEEH	School of Bioresources Engineering and Environmental Hydrology
BLOB	Binary Large Object
CEH	Centre for Ecology and Hydrology
CMA	Catchment Management Agencies
COM	Component Object Model
CRCCH	Cooperative Research Centre for Catchment Hydrology
CSV	Comma Separated Value
DLL	Dynamic Link Library
DWAF	Department of Water Affairs and Forestry
ESRI	Environmental Software Research Institute
FWACS	Fractional Water Allocations and Capacity Sharing
GIS	Geographic Information System
GUI	Graphical User Interface
HDSF	Hydrological Decision Support Framework
HTML	Hypertext Markup Language
IWMI	International Water Management Institute
IWR	Institute for Water Research
IWRM	Integrated Water Resources Management
JDBC	Java DataBase Connectivity
JNI	Java Native Interface
MAP	Mean Annual Precipitation
NWA	National Water Act of South Africa, Act 36 of 1998
RDBMS	Relational Database Management System
SQL	Structured Query Language
TMDL	Total Maximum Daily Load
UKZN	University of KwaZulu-Natal
UML	Universal Modelling Language
USA	United States of America
USDA	United States Department of Agriculture
USEPA	United States Environmental Protection Agency
USGS	United States Geological Survey
VCL	Visual Component Library
WRC	Water Research Commission
WRYM	Water Resources Yield Model
XML	Extensible Markup Language

1. INTRODUCTION

The National Water Act (NWA, 1998) of South Africa (Act 36 of 1998) specifies that Government, as the public trustee of the nation's water resources, must ensure that water is protected, used, developed, conserved, managed and controlled in an equitable and sustainable manner for the benefit of all people. The NWA has introduced new paradigms and significant challenges in the management of water resources in South Africa. These include the provision of reserves to meet human and ecological requirements as primary priorities, the devolution of the responsibility for the management of water from a national to a regional level and the requirement that stakeholders participate in the management of water resources.

The NWA calls for the equitable, efficient and sustainable allocation of water resources. This will require decision support at temporal and spatial scales appropriate to the demands of the NWA and which utilises state-of-the-art hydrology, in addition to social, economic and environmental needs which have to be considered. A process-based hydrological model, which can have meaningful links with socio-economic models, is a logical framework on which to build the Decision Support System (DSS).

The NWA requires that the Department of Water Affairs and Forestry (DWAF), as custodian of South Africa's water resources, should provide a National Water Resource Strategy (NWRS) as a framework for the management of water resources in South Africa. The draft version of the NWRS provides the implementation framework for the NWA and identifies 19 Water Management Areas (WMAs) to be managed by Catchment Management Agencies (CMAs).

It was stated in Clark and Smithers (2006) that as the demand on South Africa's water resources increases it is necessary for water to be managed and utilised with greater efficiency. The water management paradigm has changed from a focus on meeting demands from surface water supplies to a more holistic approach which also considers other aspects such as water quality, the environment as a water user, surface water-groundwater interactions, and social and economic impacts. In recent years there has been an international trend towards this new water management paradigm known as Integrated Water Resources Management (IWRM) (Havnø et al., 2002; Schulze, 2002). In South Africa this new paradigm has been captured in Section 3 of the National Water Act (NWA, 1998) of South Africa (Act 36 of 1998) as follows:

> "*3. (1) As the public trustee of the nation's water resources the National Government, acting through the Minister, must ensure that water is protected, used, developed, conserved, managed and controlled in a sustainable and equitable manner, for the benefit of all persons and in accordance with its constitutional mandate.*
> *(2) Without limiting subsection (1), the Minister is ultimately responsible to ensure that water is allocated equitably and used beneficially in the public interest, while promoting environmental values.*
> *(3) The National Government, acting through the Minister, has the power to regulate the use, flow and control of all water in the Republic.*" (NWA, 1998).

As explained in Clark and Smithers (2006), IWRM requires a greater understanding of complex hydrological systems. Since the 1980s advances in computer science combined with improved understanding of hydrological processes has resulted in the development of computer simulation models that encapsulate understanding of hydrological systems. However, many of the models were developed for a particular domain within the hydrological system, such as surface water, groundwater or ecology, by experts in that domain. IWRM requires that the models for each domain be integrated and that scientific experts collaborate in the holistic assessment and management of water resources.

As discussed in Clark and Smithers (2006), the need to integrate water resources models has led to the recognition of the need for and development of modelling frameworks to facilitate integrated water resources modelling. Traditionally models have each been run within their own modelling systems, each consisting of similar tools to prepare model input data, write model input files and analyse model output. Modelling frameworks seek to reduce duplication of effort and facilitate model use and development by providing common data preparation and post-processing tools, including a Geographic Information System (GIS) and database management, for use with several models. From the literature reviewed it is apparent that there is a growing realisation of the need for hydrological modelling frameworks and that there are a number of different frameworks under development internationally (HarmonIT, 2002; Marston et al., 2002; Argent and Rizzoli, 2004).

The aims of the project, as stated in the contract, have been to:

- Provide hydrological decision support for the implementation of the NWA (1998)
- by identifying a suite of suitable and appropriate modules for use by the CMAs
- the linking/integration of selected modules within a Hydrological Decision Support Framework (HDSF), which could include:
    - a generic user interface
    - integrated GIS
    - generic, extensible spatial and temporal databases with seamless input/output to/from models
- The HDSF should be structured to enable inclusion of additional modules
- Build capacity through the implementation of the project.

The aims of the project were translated into the following objectives:

- The primary objective was the development of a Hydrological Decision Support Framework (HDSF) and integration of relevant modules into framework
- The second objective was to extend the capabilities of the some of the modules so as to enable the assessment of water resources and the allocation of water use licences at the level of CMA and to consolidate and encapsulate existing relevant research findings into the selected simulation models
- The third objective was to provide user support and documentation.

The primary objective of this project was the development of a Hydrological Decision Support Framework (HDSF) which can incorporate relevant and appropriate simulation models linked by a common flexible and extensible database and integrated with a GIS for use at a planning and operational level by Catchment Management Agencies (CMAs) at spatial scales ranging from point of use to the entire Water management Areas (WMA) and at temporal time scales of one day. The first step towards this objective was to identify the requirements for the HDSF. Several existing hydrological modelling frameworks and framework development tools were reviewed and evaluated relative to the identified requirements. Based on this evaluation, it was decided not to develop a new framework and one of the existing hydrological modelling frameworks was selected for adoption and further development within the project to form the foundation of the HDSF. In addition, a framework for linking hydrological models was selected for evaluation within the project. The project then entered a design phase where enhancements to the selected hydrological modelling framework and additional modelling tools were conceptualised. The implementation phase of the HDSF then took place and resulted in the HDSF software presented in this report.

A secondary objective of this project was to provide for the modelling requirements of Catchment Management Agencies (CMAs) for the assessment and management of available water resources and their dynamics within a catchment by extending the capabilities of the some of the modules well as consolidating and encapsulating existing relevant research findings into selected simulation modules in order to refine the simulation of hydrological processes. It was recognised that no single model would be suitable to meet all the modelling requirements necessary for IWRM and that also the HDSF should not be restricted to a fixed set of models, but rather provide a framework within which any compatible model could be run. It was decided that a physical conceptual model operating at a daily time-step would be necessary to provide for the modelling requirements of CMAs and should be included in the HDSF as part of this project.

An additional objective of this project was to compile a collection of readily available datasets that would be useful to users of the HDSF. These data sets would include both spatial and temporal data relevant to hydrological modelling.

2. FRAMEWORK DEVELOPMENT

The need for integrated water resources modelling to facilitate IWRM has led to the recognition of the need for and development of hydrological modelling frameworks. At the simplest level, individual models each with their own software support tools could be integrated by simply translating data or information that is common between the models from the format used by one model to the format used by another model. At the next level models are modified to operate within a modelling framework based on a common data format and including a common set of data preparation and post-processing tools, including Geographic Information System (GIS) and database management related tools. "Some modelling frameworks have progressed further than this by breaking models down into sets of modules which may be linked together to create models tailored to meet specific IWRM scenarios (Kralisch et al., 2004; Rahman et al., 2003; Argent and Rizzoli, 2004). However, there are many legacy models in existence, and as it is not financially practical to restructure all these models into a modern programming language, some other means of integration has to be developed. Such legacy models have been linked in series by running one model for a simulation period, converting the model output files to the input format of the second model, and then running the second model. This method of linking models is suitable if there are no feedback relationships between components of the two hydrological systems simulated by the models. Recent developments in computer science provide solutions to this problem and allow integration of legacy models on a time-step-by-time-step basis, which is referred to as linking models in parallel. Integrated models allow modelling at different but appropriate spatial and temporal scales. Examples are modelling surface water runoff at a daily scale but baseflow at a monthly scale, or modelling areas under irrigation at field scale but the upstream catchment areas at a broader catchment scale. However, the ability to link models does not make integrating models a trivial matter and it may be incorrect to link two models of different spatial or temporal scales if the concepts on which the models are based are incompatible. Hence the collaboration of experts in the different water resources modelling domains is crucial for integrated water resources modelling." (Clark and Smithers, 2006).

2.1 User Requirements

The user requirements for a modelling framework were compiled based on experience from members of the project team, from the Inception Workshop (Appendix B.1) for the project and from a user requirement survey reported in Appendix B.2, though the survey focussed on modelling requirements. These user requirements were summarised for use in the Framework Review report (Appendix B.3) and are shown in Table 2.1. The requirements that were addressed as part of this project are indicated in the right-most column of Table 2.1.

Table 2.1: Requirements for the proposed framework

Category	Requirement	Addressed in Project
General	Management of modelling projects and scenarios.	No
	Can be run on a range of computer operating systems.	No
GIS	GIS interface for catchment configuration and display of spatial model input and output data/information. GIS will be a key feature of the HDSF and a comprehensive set of GIS tools will be required. Ideally need to conform to some sort of standard or at least adopt the most widely used software and data format. Required functionality includes: • Catchment delineation. • Network analysis. • Building and display of schematic flow network diagrams. • Extraction and processing of spatial data in vector and grid format (including Theme-on-Theme analysis). • Display of spatially linked input and output data. • Facilities to import spatial data stored in other GIS formats.	Partial No No No Yes Yes No
Database / Data model	Well designed extensible geo-referenced database structure including storage of spatial and temporal data/information. • Need an internationally recognised geodatabase structure. • Should be able to implement the geodatabase in a number of different Relational Database Management Systems (RDBMSs). • Database needs to store model input and output data.	Yes No Yes Yes
Input	User friendly graphical user interface allowing model input data to be entered and reviewed with facility to do range and logic checking of data entered.	Partial
	Easy configuration of catchments and flow networks.	No
	Extensible to allow for easy addition of new models and model variables.	Yes
Output	Comprehensive and easy to use set of tools for post-processing and display of model output. • Spatial display of data. • Graphical display of time-series data. • Comparison of simulated versus observed data. • Comparison of data from two scenarios. • Statistical analyses. • Display of water supply versus demand.	Yes Yes Yes Yes Yes Yes No
Models / Modules	Extensibility of links between databases, user interfaces, GIS and models or modules to facilitate changes and additions to modelling requirements. • Allow legacy models (written in various programming languages) or new modules to be easily integrated, this would probably require facilities for wrapping of model code and some form of interface standard for development of new modules. • Consider using existing hydrological modelling frameworks.	Partial Partial Yes
	Linking of models/modules in series and parallel.	Partial
	To use the framework for near real-time simulations the models or modules used need to be able to "hot-start". This may also be required when running models in parallel.	No
Help	Comprehensive and extensible help system describing the models and modules and the use of the framework.	Partial

A point was made at the Framework Selection Workshop (Appendix B.4) that a modelling framework should not only make modelling easier but should allow smarter modelling, such as linking models in parallel. Hence, the HDSF should not only seek to make modelling easier but also to provide an environment that facilitates integrated modelling.

2.2 Framework Review

A brief initial investigation into and assessment of existing modelling frameworks was carried out and reported on at the Inception Workshop (Appendix B.1) for the project. This assessment was generally based on information gathered from websites for the frameworks and knowledge within the project group. The purpose of the initial assessment was to give direction to a more detailed investigation of modelling frameworks. During this initial investigation it became clear that a definition was required regarding what exactly was meant by the term “modelling framework”. The following definition was proposed for use in this project:

> *A modelling framework is an open modelling environment in which existing models (or modules), databases and user interfaces can be linked or organised in a consistent manner.*

Based on this initial assessment six of these frameworks were then selected to be reviewed in more detail. Selected Geographic Information System (GIS), database systems and data models were also reviewed in more detail. These detailed reviews are documented in the Framework Review report (Appendix B.3). A summary of the six frameworks reviewed appears in Clark and Smithers (2006) and has been reproduced in Section 2.2.1. The review of GIS tools in the Framework Review report (Appendix B.3) has been summarised in Section 2.2.2. The review of database systems and data models in the Framework Review report (Appendix B.3) has been summarised in Section 2.2.3.

2.2.1 Review of Frameworks

“After an initial review of modelling frameworks, six were selected to be reviewed in more detail. These six frameworks were selected as they had either been developed in South Africa, used in South Africa or were based on new computer programming technology. On completion of the reviews it was found that the six modelling frameworks reviewed could be divided into three categories:

- Modelling Framework Application: BASINS, SPATSIM, WR IMS
- Modelling Framework Development Tool: OMS, TIME
- Model Linking Framework: OpenMI

The modelling framework applications are complete functional applications and have the potential to be customised to suit the requirements of the HDSF. If none of the modelling framework applications were suitable for the HDSF project then the modelling framework development tools could be used to build a whole new modelling framework application for the HDSF. In addition to development tools for graphical user interfaces (GUIs), OMS and TIME both include protocols and modelling infrastructure supporting the development of hydrological modelling modules which may be used to built customised hydrological models. The OpenMI model linking framework could be used to link models in whichever modelling framework option was selected. A brief overview of each of the six frameworks reviewed is presented in this section.” (Clark and Smithers, 2006).

BASINS

“The Better Assessment Science Integrating point and Nonpoint Sources (BASINS) system is described as a multipurpose environmental analysis system designed for use by regional, state and local agencies in the United States of America (USA) to perform catchment and water quality based studies (USEPA, 2001). It is designed to support the estimation of total maximum daily loads (TMDLs) using a catchment-based approach including both point and non-point sources for a variety of pollutants, at a variety of scales (USEPA, 2001). The objectives of BASINS are to facilitate examination of environmental information, support analysis of environmental systems and provide a framework for examining management alternatives. BASINS was developed by the United States Environmental Protection Agency's (USEPA's) Office of Water and was first released in 1996. " (Clark and Smithers, 2006).

"BASINS is a system integrating GIS, data analysis and modelling. The BASINS system includes a set of custom databases compiled from a wide range of federal sources in the USA including national databases at the USEPA and United States Geological Survey (USGS). Data are extracted from these databases and stored in Water Data Management (WDM) and DBF files which may be accessed by BASINS models and tools. BASINS uses ArcView to provide a customised user interface. The user interface contains all the standard ArcView menu, button, and toolbar items which access query, spatial analysis, and map generation tools. The tools and menus specific to BASINS are accessed through the BASINS Extension Manager and include assessment tools, data management utilities, catchment characterisation reports, water quality models and catchment hydrology models. BASINS uses the GenScn post-processing tool to facilitate the display and interpretation of observed water quality and other time-series data, and the analysis of model output data. Models to be run from BASINS would need to be able to read from and write to the relevant BASINS files in WDM and DBF format. Currently the HSPF, SWAT, QUAL2E and PLOAD models are included in BASINS." (Clark and Smithers, 2006).

SPATSIM

"The Spatial and Time-series Information Modelling (SPATSIM) software is described as an integrated hydrology and water resource information management and modelling system (IWR, 2005). SPATSIM integrates spatial and time-series information in a flexible framework that includes a variety of data storage, retrieval, analysis and display options suitable for the application of a range of different models (Hughes, 2002). SPATSIM was developed by the Institute for Water Research (IWR), Rhodes University, Grahamstown, and was first released in 1999." (Clark and Smithers, 2006).

"The design of SPATSIM is based on the approach adopted by the Centre for Ecology and Hydrology (CEH) at Wallingford, United Kingdom (Hughes, 2002). SPATSIM contains an extensible database structure enabling storage of spatial and temporal data. Spatial data are stored in Environmental Software Research Institute (ESRI) shapefiles and a set of related attributes stored in a Paradox based relational database. The link between geographic data and attribute data is achieved using four data dictionary tables stored in the database. The GIS functionality of SPATSIM includes vector based GIS tools for visualisation and management of spatial input and output data. SPATSIM includes tools for the analysis and display of time-series data. SPATSIM has been designed in an extensible manner such that new models may be easily added without changing SPATSIM. For a model to be run from SPATSIM, the model or the model user interface needs to be modified to read from and write to a SPATSIM database. The data structure used in SPATSIM facilitates linking of models in series." (Clark and Smithers, 2006). SPATSIM is also described in Hughes (2005).

WR IMS

"The Water Resources Information Management System (WR IMS) is a framework for the management of water resources modelling studies and their data requirements. WR IMS also facilitates the storage and retrieval of metadata related to modelling studies. WR IMS was developed by the Water Resource Planning Systems – Systems Analysis sub-directorate at the Department of Water Affairs and Forestry (DWAF), South Africa. WR IMS was originally developed as a database management system for hydrological and system data to run the Water Resources Yield Model (WRYM) for a project in the Vaal River Basin in 1999 (DWAF, 2004b). It has been developed further into a more general purpose information management system." (Clark and Smithers, 2006).

"WR IMS uses a relational database to store data and metadata related to studies, models, sub-areas and scenarios. WR IMS currently uses a Microsoft Access database but other Relational Database Management Systems (RDBMSs) could be used. Each model included in WR IMS would require data tables specific to the model to be included in the WR IMS database or to be accessed from a different database. WR IMS includes simple GIS functionality to display spatial data stored in ESRI shapefiles. Each model included in WR IMS requires a model manager which may be accessed by means of a COM interface. Each model manager includes a graphical user interface specific to its model, and should be able to display and set up input data for the model, run the model, save model output, and display and analyse output from the model." (Clark and Smithers, 2006).

OMS

"The Object Modeling System (OMS) is described as a Java-based modelling framework consisting of a library of science, control and database components, which facilitates the assembly of selected modelling components into a modelling package suited to the problem, data constraints and scale of application (OMS, 2005; Ahuja et al., 2004). OMS is a modelling framework that facilitates model development, evaluation and deployment (David et al., 2004). The concept behind OMS is to create all system and model tools as independent reusable components that may be coupled using standardised software interfaces to create an application specific modelling package (Kralisch et al., 2004). Development of OMS began in 1996 in the Institute for Geography at Friedrich Schiller University, Jena, Germany (Ahuja et al., 2004). Since October 2000 development of OMS has continued as an interagency project supported by the United States Department of Agriculture - Agricultural Research Service (USDA-ARS), United States Geological Survey (USGS) and United States Department of Agriculture - Natural Resources Conservation Service (USDA-NRCS) (Ahuja et al., 2004)." (Clark and Smithers, 2006).

"The two main types of components in OMS are model components and system components. Model components are the building blocks from which models are created within OMS. System components are those used to assemble user selected model components to create an application specific model, populate the model with suitable data and then execute the model. OMS is supported by graphical user interface (GUI) components and utility components which include a data dictionary, data retrieval and storage, GIS, graphical visualisation and statistical analysis (OMS, 2005; Ahuja et al., 2004). OMS uses custom metadata tags to support component documentation, testing, proper component integration into a model, automatic user interface generation and model execution (David et al., 2004). Tools are available in OMS to assist in migrating legacy models either by direct implementation in OMS or by means of wrappers. It appears that individual models built from OMS model components would be able to be linked." (Clark and Smithers, 2006).

TIME and the Catchment Modelling Toolkit

"The Invisible Modelling Environment (TIME) is described as a modelling and programming system for developing, applying and deploying environmental models (Murray et al., 2004). TIME has been developed on the Microsoft .Net platform and is a collection of .Net classes, libraries and visualisation components for the development of models and model applications. The Catchment Modelling Toolkit is a system of environmental modelling software which integrates a new generation of catchment models and modelling support tools (Marston et al., 2002). The aim of the Catchment Modelling Toolkit is to provide land and water managers, researchers and educators with an integrated collection of software tools and components to simulate catchment response to management and climate variability, at a range of scales and using a variety of approaches (Marston et al., 2002). A modelling framework was required to achieve this aim, allowing models to be developed and integrated quickly and consistently (Rahman et al., 2003). TIME is an environmental modelling framework developed to meet these requirements and is the foundation on which the models, model applications and other modelling tools included in the Catchment Modelling Toolkit are built (Catchment Modelling Toolkit, 2005). TIME and the Catchment Modelling Toolkit were developed by the Cooperative Research Centre for Catchment Hydrology (CRCCH) in Australia." (Clark and Smithers, 2006).

"The architecture of TIME is divided conceptually into five layers: Kernel, Data, Models, Tools, and Visualisation and User Interface. Each layer consists of a family of classes, with the classes in the upper layers using services provided by classes in lower layers. Developers create models in the Model layer using classes in the Kernel and Data layers. The Tools and Visualisation and User Interface layers contain classes that interact with models and provide most of the framework functionality, such as user interface generation and model linking. The use of TIME custom metadata tags allows models to remain independent of these tools, resulting in better model stability (Rahman et al., 2002). TIME makes use of .Net's introspection capabilities for dynamic discovery of system properties at runtime (Rahman et al., 2003). These system properties include class structure, class fields and methods, and custom metadata tags which allow TIME to automate several tasks which facilitate model integration and automatic user interface generation (Rahman et al., 2003). TIME seems to support the approach of restructuring models into a set of linked modules and does not appear to make provision for wrapping or linking to legacy models." (Clark and Smithers, 2006).

OpenMI

"The Open Modelling Interface (OpenMI) is described as a generic interface allowing models simulating different water related processes to be linked on a temporal and spatial basis, allowing simulation of process interactions (Gijsbers, 2003; HarmonIT, 2005a). The objective of OpenMI is to simplify the linking of models running in parallel and which operate at different spatial and temporal scales by means of direct transfer of data values between models without writing to or reading from intermediate text files. OpenMI focuses on resolving or improving several complicated model linking issues, including differences in spatial and temporal scales, feedback loops, differences in spatial and temporal concepts (distributed vs. lumped, steady state vs. dynamic), different units and naming of variables, and distributed computing (Blind and Gregersen, 2004). OpenMI simplifies the linking of models from a computer science point of view, allowing modellers to concentrate on the complexities of linking models from a hydrological point of view. OpenMI has been developed under the European Commission funded HarmonIT project towards meeting the goals of the European Union's Water Framework Directive. The project partners include three commercial partners (DHI - Water & Environment, WL | Delft Hydraulics, HR Wallingford Group) and several other partners from research institutes and universities in Europe. The fact that competing software vendors have joined forces in creating OpenMI, is a key advantage to achieving the objective of setting a standard (Blind and Gregersen, 2004). ” (Clark and Smithers, 2006).

“The architecture for OpenMI in shown in Figure 2.1. The most important part of OpenMI is the Standard (org.OpenMI.Standard), which consists of a set of interfaces. The Standard is not a piece of software, so may be implemented in any object-oriented programming language and related computing platform. Any model implementing the relevant interfaces contained in the OpenMI Standard is described as being OpenMI compliant and may be linked to any other OpenMI compliant model. While OpenMI focuses on data exchange between models at runtime, it may also be used to link models to databases and user interfaces (Blind and Gregersen, 2004). The HarmonIT project has concentrated on implementing the Standard in the C# programming language on the Microsoft .Net computing platform but will also provide a Java implementation (HarmonIT, 2005b). The other namespaces in the OpenMI architecture form the Open Modelling Environment, and provide a set of classes whose purpose is to simplify the migration process and to facilitate the linking and running of the OpenMI compliant models (HarmonIT, 2005a). A default implementation of each interface in the Standard has been created to form a group of classes known as the Backbone (org.OpenMI.Backbone). The org.OpenMI.DevelopmentSupport namespace contains a generic set of low-level classes that can be used in the development of an OpenMI modelling environment. The org.OpenMI.Utilities namespace contains a set of classes that have been created to reduce the amount of re-engineering required to migrate existing model engines and software systems to become OpenMI compliant. A primary design objective for OpenMI was that the cost, skill and time required to migrate an existing model to the standard should not be a deterrent to its use (HarmonIT, 2005a). The HarmonIT project recognises that there are many legacy models written in programming languages such as FORTRAN. To facilitate the linking of legacy models, a group of wrapper classes have been created which allow linking to these legacy models, without rewriting these models in an object-oriented programming language to meet the OpenMI Standard, and with a minimum of changes to the legacy models themselves. These wrapper classes take care of all the bookkeeping associated with handling links, events, exceptions, buffering and basic spatial and temporal interpolation. For a legacy model engine not written in a .Net language or Java to be suitable for wrapping it must be compiled to a dynamic link library (DLL) and must be able to separately initialise, perform single time-steps, finalise and to be disposed of (HarmonIT, 2005b). The org.OpenMI.Configuration namespace contains a set of classes created to help developers to administer, configure and deploy coupled OpenMI compliant modelling systems. The org.OpenMI.Tools namespace contains user interface components and other classes to facilitate user interaction with OpenMI compliant models at configuration and run-time.” (Clark and Smithers, 2006).

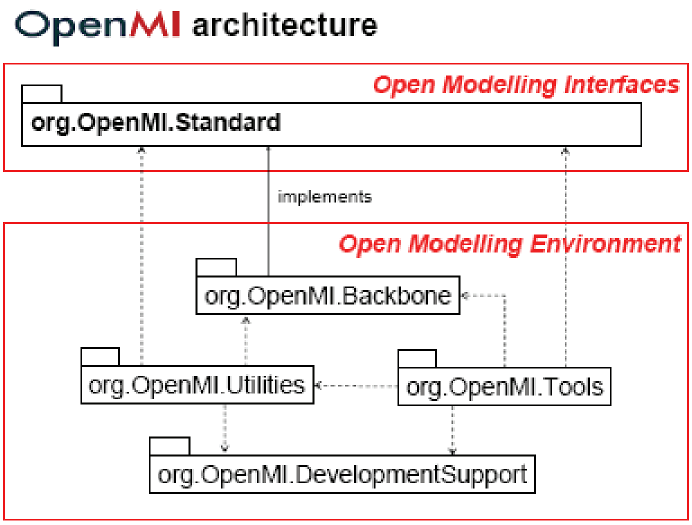


Figure 2.1: OpenMI architecture namespaces (HarmonIT, 2005c)

2.2.2 Review of GIS Tools

GIS is a useful tool for visualisation, analysis and generation of water resources data and information. GIS tools are an important component of the HDSF, however, it was not intended to compete with commercial GIS software packages which include complex GIS analysis tools. It is important that the HDSF be compatible with one or more commercial GIS software packages to facilitate the exchange of spatial datasets, but not be completely dependent on these for all GIS tools. With regard to GIS data file formats, there is currently no single standard, as each commercial GIS software vendor has their own formats. The review made the assumption that ESRI's GIS software and data formats were the most widely used in South Africa and would thus be the focus of the review. The review acknowledged the fact that at that time ESRI's old generation ArcInfo and ArcView 3.x products were probably the most widely used. However, the new generation ArcGIS range of products offer many advantages for the HDSF and thus the review focussed on the ArcGIS range of products.

Some of the modelling frameworks reviewed included GIS tools or provided software components that could be used to develop GIS tools. The GIS related functionality of these modelling frameworks was summarised as part of the review of GIS tools (Appendix B.3) and is summarised further in Table 2.2.

Table 2.2: GIS functionality in reviewed modelling frameworks

Framework	GIS Functionality
Basins	BASINS 3.0 is a customised ESRI ArcView 3.x GIS application and thus has access to the full ArcView 3.x functionality. ArcView Spatial Analyst is required for some of the advanced features. At the time of the review BASINS was being restructured to create the BASINS 4.0 Systems Application software package which was expected to include some non-proprietary mapping tools and would enable the selection of either ArcView 3.x or ArcGIS 8.x as external GIS tools.
OMS	OMS includes a collection of graphical user interface components that may be used by developers to create custom visualisation tools such as maps and graphs to display modelling results.
SPATSIM	ESRI MapObjects controls are used to provide the GIS component of SPATSIM. In SPATSIM spatial features are represented by geographic data stored in ESRI shape files and a set of related attributes stored in a Paradox database. The GIS functionality provided in SPATSIM includes simple viewing and rendering tools required for setting up and storage of the attribute data required as input to models, and for displaying model output data.
TIME and the Catchment Modelling Toolkit	TIME contains a collection of visualisation and user interface classes including controls for displaying vector and raster maps. The E2 modelling application in the Catchment Modelling Toolkit contains GIS functionality to create flow networks of subcatchments, nodes and links, and also for display and analysis of model output data. CatchmentSIM is a standalone 3D-GIS application in the Catchment Modelling Toolkit. CatchmentSIM contains a collection of topographic and hydrologic analysis algorithms that may be used to delineate catchments, subdivide catchments into subcatchments, calculate their areas and spatial topographic attributes, and analyse their hydrologic characteristics.
Water Resources IMS (WR IMS)	The WR IMS includes simple GIS functionality created with MapObjects LT to display study sub-areas and other spatial feature layers. The spatial feature layers displayed are set up in a configuration file that references a set of ESRI shapefiles. The GIS Viewer software module contains a set of ActiveX GIS tools written using MapObjects LT that may be used to include GIS functionality in applications.

ArcGIS developed by ESRI is an integrated collection of software products for building a complete GIS suited to the requirements of a particular user (ESRI, 2005). ArcGIS is based on a modular, scaleable, cross-platform architecture consisting of libraries of software components called ArcObjects. The ArcView, ArcEditor and ArcInfo versions of ArcGIS Desktop may be customised and extended using ArcObjects. The ArcGIS Desktop applications could be extended to include links to the HDSF. ArcGIS Engine is a set of cross platform ArcObjects that may be used to develop custom embedded GIS and mapping functionality in standalone software applications. ArcEngine could be used to develop simple GIS functionality within the HDSF. However, the advantages of using ArcGIS would need to be evaluated relative to the licensing costs.

2.2.3 Review of Database Systems and Data Models

One of the key requirements for the HDSF was a database structure suitable for the storage of spatial and time-series data, to enable a common data source to be used for all models included in the modelling framework. As part of the framework review some hydrology related database management systems and data models were investigated. ESRI appear to be leading the development of the type of database structure described above and have used the term "geodatabase" to describe such a data model. The SPATSIM and BASINS modelling frameworks may be considered to already include a geodatabase and the data models for these were evaluated as part of the review of these frameworks. In addition ESRI's ArcGIS geodatabase and Arc Hydro data models, the Geography Markup Language (GML) and the

Hydstra database system were investigated and details regarding these can be found in the Framework Review report (Appendix B.3).

The ArcGIS geodatabase is described in detail in Zeiler (1999). It is described as an object-relational database as it uses a relational database to store information about spatial and non-spatial features using an object-oriented data model. The ArcGIS geodatabase data model does not just store the shape, location and attributes of spatial features but also the behaviour of features and the relationships between them. The ArcGIS geodatabase introduces the concept of feature classes, where the features within a particular feature class have similar behaviour and relationships. The ArcGIS geodatabase data model also allows for non-spatial features which may be simply referred to as objects. The object-oriented nature of the ArcGIS geodatabase data model enables it to represent more closely the actual water resources system, and also facilitates the use of the geodatabase by object-oriented water resources simulation models. Features classes and object classes are represented in the geodatabase as a set of relational tables. A feature or object is represented as a record in a table. Feature or object attributes are represented as fields in a table. The ArcGIS geodatabase data model has been designed to store vector, raster and TIN geographic data.

ESRI together with experts in various GIS application domains has developed several data models to extend the ArcGIS geodatabase. One of these data models, Arc Hydro, has been developed specifically to represent water resources systems and is described in Maidment, (2002). Arc Hydro operates within ArcGIS and is a geospatial and temporal data model for water resources. Arc Hydro has an associated set of tools which populate the attributes of features in the data framework, connect features in different data layers, and support hydrologic analysis. The purpose of Arc Hydro is to facilitate the building of hydrologic information systems which combine geospatial and temporal water resources data to support hydrologic analysis and modelling (Maidment, 2002). Arc Hydro is a data model that supports information management for water resources and linking of hydrologic models (Maidment, 2002). The ArcHydro data model is intended as a guide and can be adapted to suit specific modelling requirements. It should be noted that the Arc Hydro data model was designed to describe natural water systems and it does not describe water pipe systems (Maidment, 2002). Also the Arc Hydro data model focuses on the description of surface water hydrology and hydrography and does not contain data structures to describe aquifers and other groundwater features (Maidment, 2002).

The Geography Markup Language (GML) is an Extensible Markup Language (XML) grammar written in XML Schema for the modelling, transport, and storage of geographic information, including both the geometry and properties of geographic features (ISO, 2004). GML defines a set of entities and their properties that may be used for describing geography including features, coordinate reference systems, geometry, topology, time and units of measure. Many complex descriptions of geographic data are possible with GML, and GML may be extended or restricted to meet the requirements of a particular application. The importance of GML is indicated by its acceptance as an ISO standard and implementation by many GIS and RDBMS software vendors. Although GML has been used to store large datasets it is difficult to see the advantage of this compared to storage in a well designed database in a RDBMS.

The Hydstra database system is described an integrated suite of software and services for the management and analysis of large amounts of time-series data by the hydro-power, water resources and wastewater industries (Kisters, 2005). The Hydstra database system is designed as a complete solution including acquisition, management and analysis of engineering and environmental data. Hydstra includes components for the management of time-series data, a mapping interface and modelling tools. Hydstra includes specialised modules for the management of water quality and groundwater bores. Hydstra may be used for scheduling and acquisition of data from remote measuring stations using telemetry. Hydstra includes tools to display, edit and report on time-series data. The mapping interface is not intended to supply GIS functionality but rather to provide a means for users to select locations for which they may access time-series data, images, documents and previously compiled reports. Hydstra includes a Windows-based program for synchronous modelling using time-series data. The models are written in an algebraic-like scripting language. This modelling tool may be used for flow forecasting, yield analysis, data comparison, regression and catchment modelling. Despite the fact that Hydstra seems to be widely used, there is very little information describing it and therefore not much basis on which to recommend Hydstra for use in HDSF.

The HYMOS information system for water and environment was developed by WL | Delft Hydraulics in the Netherlands and is described as an information system for the storage, processing, analysis and presentation of data related to water resources management, surface and groundwater hydrology, meteorology, water quality and environmental assessment (WL | Delft Hydraulics, 2005a, 2005b). HYMOS combines an efficient database structure with a set of tools for data entry, validation, completion, analysis and retrieval (WL | Delft Hydraulics, 2005b). HYMOS is time-series data oriented but also provides for spatial data analysis. The open data structure enables interaction and data transfer with other databases and modelling systems. HYMOS includes the Sacramento rainfall-runoff model and other hydrological, hydrodynamic or environmental models can be linked to the system. Neither the HYMOS software nor detailed information about its data structure were available at the time of the review.

The review concluded that there was no single modelling framework option that would provide for all the requirements of the HDSF. Hence it would be necessary to adopt a combination of frameworks, GIS tools, geodatabase and data model that best suited the requirements of the HDSF project. It was further concluded that as far as possible the HDSF project should seek to adopt standards as opposed to adopting software to minimise the risks of being reliant on particular software vendors. The use of expensive proprietary software should be the user's choice and should not limit the use of the HDSF. It was recommended that software development in Java or one of the .Net programming languages should be considered to make best use of recent advances in the field of computer programming. The linking of models was considered to be important especially the linking of legacy models, but it was acknowledged that there was no easy way of linking models.

2.3 Framework Selection

The six selected frameworks reviewed were then evaluated relative to the framework requirements identified for the HDSF. This evaluation was presented at the Framework Selection Workshop and which resulted in the selection of a modelling framework for the HDSF project. Details of the framework evaluation and the Framework Selection Workshop are reported in the Framework Selection report (Appendix B.4). The framework selection options are shown in Figure 2.2. A summary of the rationale and outcome of the framework selection process appears in Clark and Smithers (2006) and has been reproduced below.

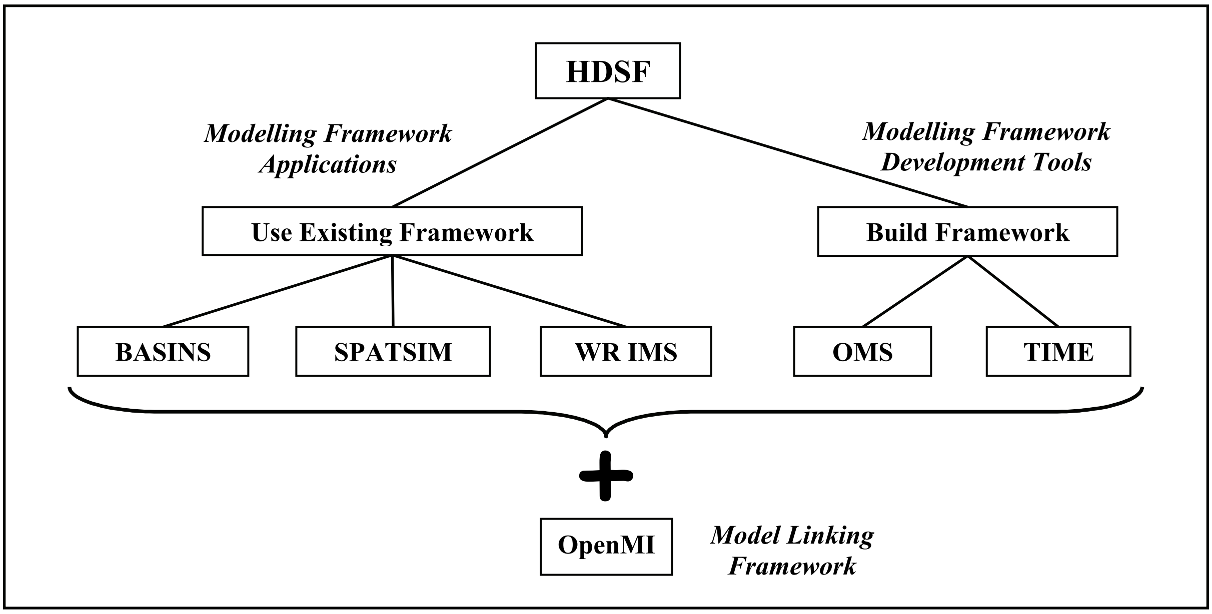


Figure 2.2: Modelling framework selection options

"The first step in selecting a modelling framework for use in the HDSF project was deciding on whether to use one of the existing modelling framework applications or to build a new framework using one of the modelling framework development tools. One of the considerations in the decision was the use of a whole model approach to incorporating models in a framework adopted by the existing frameworks, versus the modular modelling approach adopted by the framework application development tools. It was decided that the whole model approach was the safest in terms of preserving the scientific integrity of legacy models, requires the least changes to legacy models, and would gain wider acceptance in South Africa amongst users of well established hydrological modelling tools. However, the adoption of an existing modelling framework application using the whole-model approach, would not preclude the implementation of modular modelling. One of the main advantages of using an existing modelling framework application is that it would allow a working version of the HDSF to be developed in a shorter time with fewer resources. The advantages of using one of the modelling framework development tools are that a framework could be developed to meet the specific requirements of the HDSF, and that the framework developed would be based on newer programming languages and technology. Though the development of a new framework specific to the requirements of the HDSF project, based on modular modelling and using the latest programming technology was attractive, it was decided at a framework selection workshop that the best approach, given the constraints of the HDSF project, would be to use one of the existing modelling framework applications." (Clark and Smithers, 2006).

"Many factors were taken into account at the framework selection workshop when considering which existing modelling framework application to use in the HDSF project. One of the primary considerations was that the selected framework should meet most of the requirements listed in Table 2.1. Other important considerations were related to the accessibility to the code, suitability of the framework for further development, and availability of software support to the project team and future users of the framework. In particular it was acknowledged that the selection of a framework developed in South Africa would be beneficial to the project and encourage collaboration between hydrological modelling groups in South Africa. Given the above considerations, SPATSIM was selected for use in the HDSF project. SPATSIM satisfies many of the requirements for the HDSF and was developed in South Africa. The developer of SPATSIM is willing to collaborate and supports further development. A particular strength of SPATSIM is its generic extensible data model that includes storage of spatial and time-series data. SPATSIM also includes generic GIS and data analysis tools. SPATSIM is currently being used in South Africa by water resource consultants, DWAF and in other WRC projects such as the Water Resources of South Africa, 2005 Study (WR2005). SPATSIM already includes several models including Reserve determination models and the widely used Pitman monthly model. The adoption of SPATSIM will facilitate both the further development of SPATSIM software and collaboration between hydrological modelling groups in South Africa." (Clark and Smithers, 2006).

"To satisfy the requirement for the HDSF of being able to link models in parallel, OpenMI was adopted. OpenMI seems to be conceptually well designed and appears to be the best attempt so far at creating a standard for linking models, which is a result of the relatively large and diverse group of partners involved with the HarmonIT project. Some of these partners will be using OpenMI in commercial hydrological modelling software products. OpenMI has been well documented during the HarmonIT project. OpenMI is intended to be used for linking whole models, and not for the plug-and-play building of models from a collection of model components, which makes it suitable for the requirements of the HDSF project. Implementing OpenMI will give users access to a variety of other OpenMI compliant modelling tools. The purpose of OpenMI is to provide a standard means of linking models, particularly legacy models, and will also allow linking of models to databases and user interfaces. Any attempt to link legacy models requires changes to the model code, although OpenMI tries to minimise these changes as far as possible, some changes will still be necessary." (Clark and Smithers, 2006).

At the Framework Selection Workshop (Appendix B.4) there was also some discussion regarding GIS and there was general agreement among delegates that:

- The GIS tools included in the HDSF needed to be based on a standard, and that ESRI's ArcGIS appeared to be the current standard
- Although ArcGIS was not widely used at the time of the workshop for hydrological modelling in South Africa, ArcGIS should be adopted as by the end of the project ArcGIS would probably be more widely used
- ArcView 3.x was still widely used, so where possible the development of GIS tools should facilitate the use of ArcView 3.x
- The high cost of ArcGIS would not be prohibitive to the potential users of the HDSF.

Based on the selection of the SPATSIM modelling framework and OpenMI the project then moved into a design and development phase in which the SPATSIM modelling framework was further developed to meet the modelling framework requirements identified for the HDSF.

2.4 Framework Design

The SPATSIM modelling framework which consists of the SPATSIM user interface, the SPATSIM database data model and the existing set of SPATSIM compliant models was selected to form the core of the HDSF. The SPATSIM modelling framework required further development to meet the requirements for the HDSF. The new version of the SPATSIM modelling framework developed within this project is referred to as SPATSIM-HDSF. The following gives a broad outline of the requirements of the main components of the HDSF and the purpose of each of these components:

- **Data Model:** The HDSF requires a well designed extensible data model to enable storage of both spatial and temporal data. The data model should facilitate the storage of model input and output data. It should be possible to create a separate database for each modelling project and for databases to be archived. The data model should be designed for the purpose of providing data to and receiving data from simulation models and not as a central repository for national datasets. Tools should be provided in the HDSF to import data from national databases.
- **GIS Tools:** GIS will be an important tool for the setting up of model input data and information and for the analysis of model output data. The HDSF should contain a set of basic GIS tools facilitating the configuration of catchments and flow networks, and for viewing spatial model input and output data. These will be referred to as Internal GIS Tools. The HDSF should facilitate exchange of data with commercial GIS systems and possibly provide HDSF specific tools to be used within these commercial GIS systems. These will be referred to as External GIS Tools. The HDSF should be based on the most widely used GIS data formats.
- **Model Input:** The HDSF should have a user friendly GUI allowing model input data to be entered and reviewed with facility to do range and logic checking of data entered. Ideally the HDSF should provide an extensible generic GUI suitable for all models. However, provision should be made for model specific GUIs to be used as these often contain complex data checks.
- **Data Analysis:** The HDSF should include a comprehensive and easy to use set of tools for analysis and display of model output. These tools should include the graphing and comparison of time-series data and associated statistical analyses.
- **Models:** The HDSF should allow new models to be added in as simple a manner as possible, although it is recognised that the addition of a new model would only be done by the developers of the HDSF, or suitably qualified experts, and not by every user of the HDSF. It should be possible to add legacy models to the HDSF and the addition of these legacy models should require the minimum of code changes. The links between models and other components of the HDSF such as the graphical user interface (GUI), GIS tools and database should be as extensible as possible to allow for changes to models. The HDSF should include facilities for the linking of models in series and parallel.
- **General:** The HDSF should include a system of managing modelling projects and different scenarios within a project, and the archiving of projects. The HDSF is intended to run on the Windows operating platform.
- **Help:** The HDSF should include a system of help files explaining the use of the HDSF and each of the models that may be run from the HDSF.

The design phase of the project started with the design of a general framework and required the conceptualisation of the framework components and data flows as shown in Figure 2.3. The details of the original framework design can be found in the Framework Design report (Appendix B.5). This section of the report will describe the final design of the various components of the HDSF.

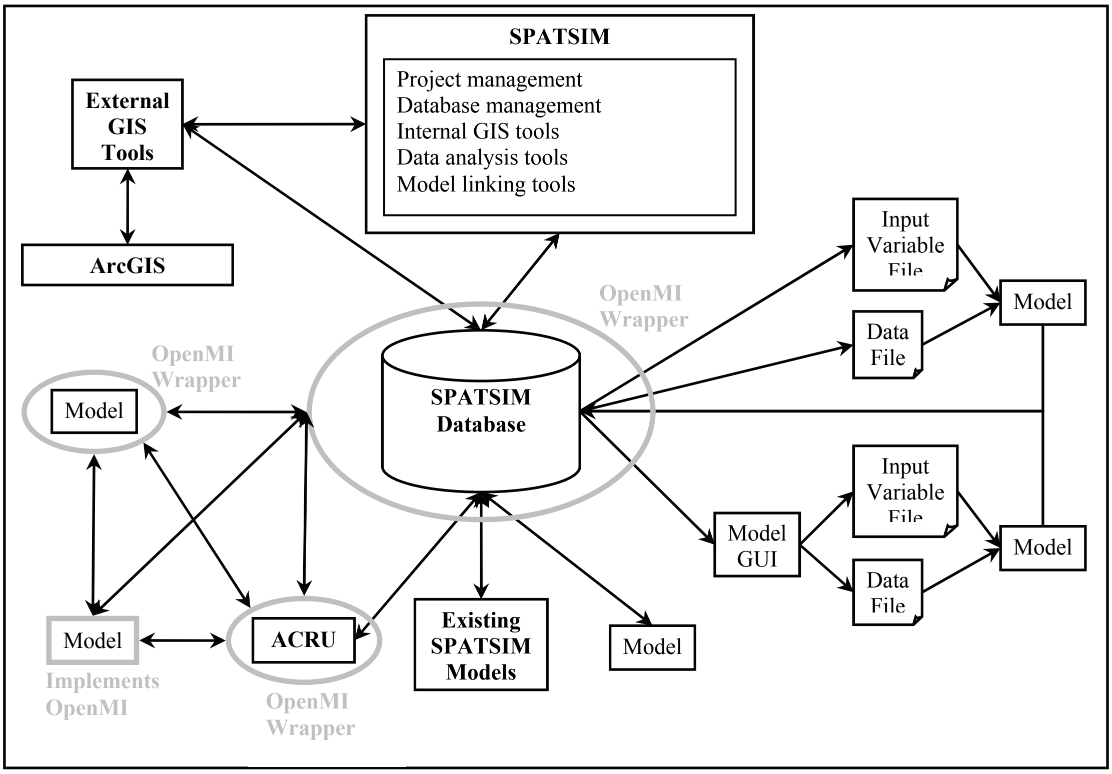


Figure 2.3: Components and data flows for the proposed HDSF

The purpose of the SPATSIM-HDSF user interface is to provide a set of general modelling utilities including:

- *Project management*
  Project management tools including setting up new project databases, archiving of projects and the management of scenarios within a project
- *Database management*
  Database management tools to add, edit and delete data in a SPATSIM-HDSF database, tools to link model input and output variables to spatial feature attributes in a SPATSIM-HDSF database and tools to import data into and export data from a SPATSIM-HDSF database
- *Internal GIS tools*
  A set of built-in GIS tools suited to setting up simple hydrological modelling inputs and providing a spatial view of model input and output parameters. These internal GIS tools are intended to provide a basic set of GIS tools enabling users to setup and run hydrological models without requiring expensive commercial GIS software. The internal GIS tools in SPATSIM have been developed using Environmental Software Research Institute (ESRI) MapObjects GIS software development components and use ESRI shape files as the storage format for geographical data.
- *Data analysis tools*
  A set of built-in data analysis tools including graphing and statistical analysis tools to enable a user to view and interpret model input and output parameters
- *Model linking tools*
  In SPATSIM the use of a common database for each modelling project enables models to be easily linked in series. If feasible, tools could be developed to facilitate the linking of OpenMI compliant models in parallel and linking of these models to SPATSIM databases via an OpenMI wrapper.

One of the reasons for selecting SPATSIM was the database data model it used. The SPATSIM database data model enables the storage of spatial and temporal data and is extensible in that new hydrological modelling attributes may be stored in the database without any change to the data model. The data model has been extended to enable storage of metadata, units of measure and relationships between components of the hydrological system being modelled. SPATSIM databases were previously implemented in Paradox and accessed using the Delphi Borland Database Engine (BDE) data components, however, databases are now implemented in Microsoft Access and accessed using the .Net framework's standard ADO.Net database components and OleDb data providers. To aid the developers of the SPATSIM-HDSF software and developers of SPATSIM-HDSF compatible models a set of data access classes has been developed to create, populate, edit and query SPATSIM-HDSF databases. Although this project has focussed on the implementation of SPATSIM-HDSF databases in Microsoft Access the shift to the use of the ADO.Net database components should enable databases to be implemented in other RDBMSs in the future.

The existing set of SPATSIM compliant models will be retained and modified where necessary due to changes made to SPATSIM. As part of the HDSF project the ACRU agrohydrological model will be modified and developed to enable it to access a SPATSIM-HDSF database and run within SPATSIM-HDSF.

A set of external GIS tools developed as extensions to the ArcGIS desktop software will provide a link between ArcGIS and SPATSIM-HDSF, and will provide SPATSIM-HDSF users with access to advanced spatial analysis tools in ArcGIS beyond the basic GIS tools provided internally within SPATSIM-HDSF. The external GIS tools will enable a user to view spatial data from a SPATSIM-HDSF database in ArcGIS, extract information from GRIDS to assist in populating a SPATSIM-HDSF database and area weighting information from one coverage, relative to a second coverage.

The HDSF project will assess the feasibility to implement the OpenMI by providing tools to enable the use of the OpenMI to link OpenMI compliant models in SPATSIM-HDSF. In addition an OpenMI wrapper could potentially be created for the SPATSIM-HDSF database as a means of linking OpenMI compliant models to a SPATSIM-HDSF database.

2.4.1 SPATSIM-HDSF Database Data Model

One of the most important criteria when selecting a framework for use in the HDSF project was the data model used by the framework. The design of the data model used in SPATSIM was based on the approach adopted by the Centre for Ecology and Hydrology (CEH) at Wallingford, United Kingdom (Hughes, 2002). One of the reasons for selecting SPATSIM was the data model used. The SPATSIM database data model was reviewed and extended taking into consideration the requirements of the HDSF project, including:

- Being extensible – adding new models and model variables must not require changes to the data model
- Being generic – it must be suitable for application of a wide range of hydrology related models
- Being object-oriented – it must enable storage of data in a manner suited to application with object-oriented models
- Enabling spatial and non-spatial components of the hydrological system to be represented

- Enabling linking of model variables to attributes in multiple features representing different components of the hydrological system
- Storing global model variables not related to components of the hydrological system
- Representing relationships between components of hydrological system
- Enabling easy running of different scenarios, while avoiding unnecessary repetition of data
- Enabling storage of metadata about datasets and values
- Storing units of measure for all relevant data types
- Including the feasibility of linking to the ArcGIS geodatabase data model.

An entity relationship diagram of the new SPATSIM-HDSF database data model is shown in Figure 2.4. Only some of the relationships between tables have been shown in Figure 2.4 in order to simplify the diagram. Descriptions, data types and other properties of the fields in each table in the new SPATSIM-HDSF database data model are documented in Appendix A.1. Details of the original SPATSIM database data model and details of the rationale for the new SPATSIM-HDSF database data model can be found in Appendix B.5. During framework development several amendments were made to the design the new SPATSIM-HDSF database data model, details of these amendments can be found in Appendices B.6, B.7 and B.8.

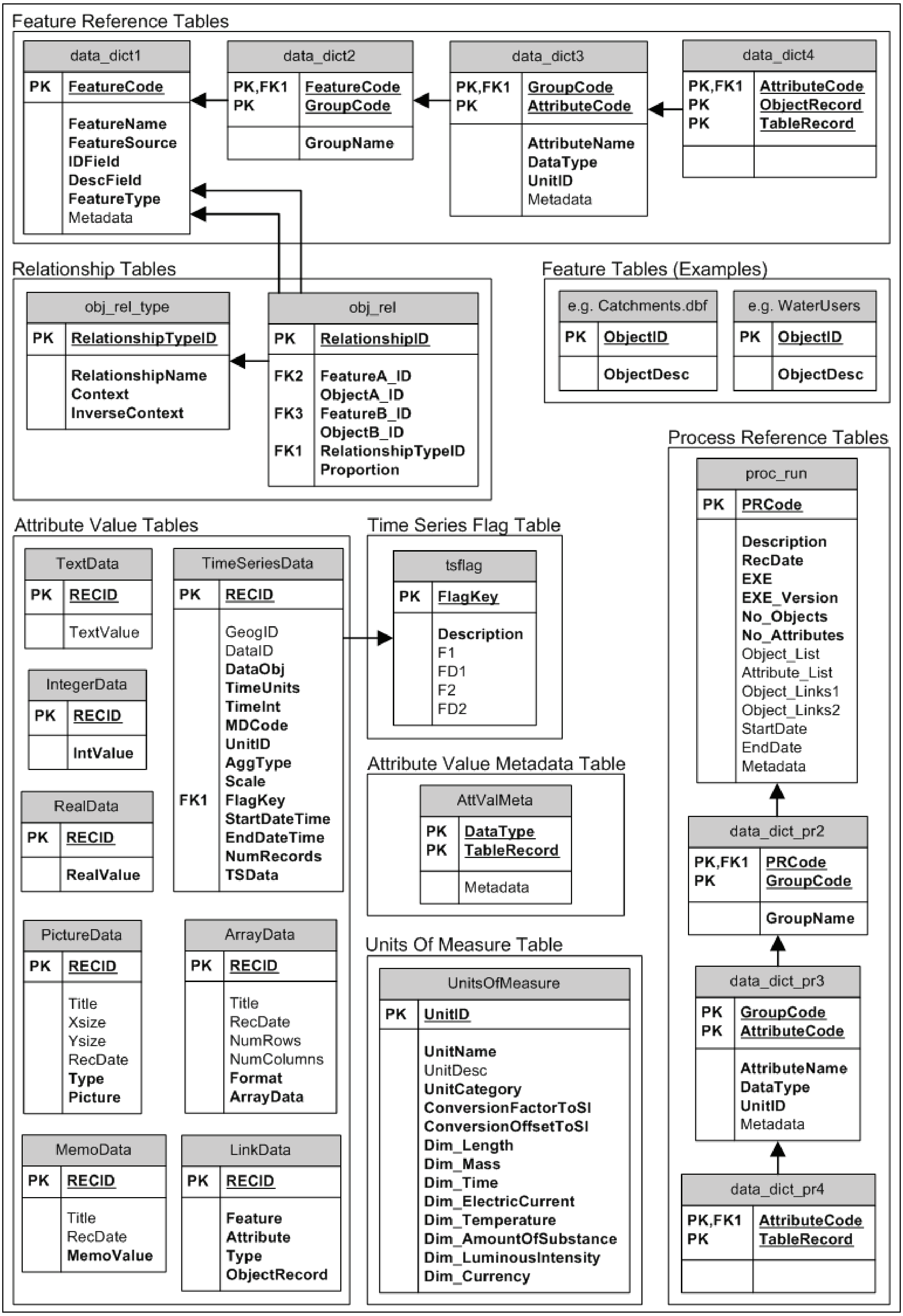


Figure 2.4: Entity relationship diagram of the new SPATSIM-HDSF data model

Spatial features are represented by geographic data stored in ESRI shape files and a set of related attributes stored in a SPATSIM-HDSF database. Each shape file used in SPATSIM-HDSF must have at least two fields in its attribute table: one field containing a unique integer number for each feature object in the shape file and a second field containing a unique text string identifying each feature object. The new data model makes provision for non-spatial features to represent non-spatial components of the hydrological system, for example, water users. These non-spatial features are represented by tables in a SPATSIM-HDSF database, one table for each type of non-spatial feature to be represented.

The link between geographic data and attribute data is achieved using four feature reference tables stored in a SPATSIM-HDSF database. The *data_dict1* table stores a list of the spatial, non-spatial and image features loaded for a project with references to the related ESRI shape files, non-spatial feature tables or image files respectively. The *data_dict2* table stores a list of attribute groups and the features to which they belong. The *data_dict3* table stores a list of attributes and for each attribute: the attribute group to which it belongs, the data type and the units of measure. The *data_dict4* table contains a cross-reference between attributes, records in a shape file or non-spatial feature table, and the corresponding record in the relevant attribute value table. A unit of measure may be associated with each attribute through a reference to a record in the *UnitsOfMeasure* table. The *UnitsOfMeasure* table contains a pre-defined fixed list of units of measure including dimensions, and conversion factor and offset values to enable conversion to SI units.

The database data model is designed such that there is a different attribute value table for each of eight data types viz. Text, Integer, Real, Time-series, Picture, Array, Memo and Link. These attribute value tables are named *TextData*, *IntegerData*, *RealData*, *TimeSeriesData*, *PictureData*, *ArrayData*, *MemoData* and *LinkData* respectively. The *TextData* table stores individual string data up to 256 characters. The *IntegerData* table stores individual signed 32-bit integer values. The *RealData* table stores individual 32-bit floating point values. The *TimeSeriesData* table stores a time-series of 32-bit floating point values in a Binary Large Object (BLOB) in the *TSData* field, that is, each record in the *TimeSeriesData* table stores a whole time-series. The other fields in the *TimeSeriesData* table store information describing the time-series. Time-series may be fixed interval or variable interval (break-point) and may contain data at a range of time intervals ranging from years to minutes. Data quality flags may also be associated with individual time-series data values. Sets of data quality flags are stored in the *tsflag* table and references to these data quality flags are stored together with the time-series data values in the *TSData* field of the *TimeSeriesData* table. The *PictureData* table enables the storage of pictures in a range of different formats as a BLOB. The *ArrayData* table enables the storage of two-dimensional arrays of 32-bit floating point values in a BLOB in the *ArrayData* field, with a reference to a text file containing information about the array. The *MemoData* table stores large string datasets that are too big for the *TextData* table. The *LinkData* table enables the storage of links to other attribute values already stored for other attributes, for the purpose of preventing the repetition of data in a SPATSIM-HDSF database, especially large time-series datasets. The *AttValMeta* table stores metadata for individual attribute values referenced by attribute value table and record.

One of the requirements for the SPATSIM-HDSF data model is that it should enable the storage of relationships between components of hydrological system, that is, relationships between objects in the same or different features. For example, it would be useful to be able to specify a representative rain gauge for each catchment being modelled. Components of the hydrological system do not exist in isolation from each other, but are interrelated to form a complex system. Relationships are important in modelling complex hydrological systems and the *obj_rel* and *obj_rel_type* tables have been included in the data model as a simple means of representing these relationships. The purpose of the *obj_rel_type* table is to store a set of relationship types, for example, a "streamflow" relationship type where the context is "is downstream of" and the inverse context is "is upstream of". The purpose of the *obj_rel* table is to store individual relationships of a specified relationship type between two specified components of the hydrological system being modelled.

The SPATSIM-HDSF software is designed with the goal of enabling a number of SPATSIM-HDSF compatible models to be run from within a common modelling environment. However, other modelling software utilities may also be run from SPATSIM-HDSF, and the SPATSIM-HDSF compatible models and utilities are collectively referred to in SPATSIM-HDSF as processes. Each time a process is run within SPATSIM-HDSF the user is required to specify necessary data related to the run in a record of the *proc_run* table. To facilitate the storage of additional information about individual process runs, for example, model options and descriptions of the scenarios run, a set of process attribute groups and attributes may be setup. These process attribute groups and attributes work in a similar manner to the feature attribute groups and attributes. The *data_dict_pr2* table stores a list of process attribute groups and the process runs to which they belong. The *data_dict_pr3* table stores a list of process attributes and for each process attribute group. The *data_dict_pr4* table contains a cross-reference between process attributes and the corresponding record in the relevant attribute value table. The process attribute values are stored in the same set of attribute value tables as the feature attribute values. Link type attributes are not permitted for process attributes.

The SPATSIM-HDSF database data model has been designed to be extensible such that model parameters for any SPATSIM-HDSF compatible model may be stored. To aid the developers of the SPATSIM-HDSF software and developers of SPATSIM-HDSF compatible models a set of data access classes has been created.

The further development of the internal GIS tools required the addition of the *FeatureTheme* table to the data model to enable the storage of display information for each theme associate with a feature in the *data_dict1* table. The external GIS tools developed within this project required the addition of the *ETBaseData* and *ETAutoCalc* tables to the data model. The *ETBaseData* table stores information about basedata coverages and GRIDs which is then available for use in the auto calculation functions. The *ETAutoCalc* table stores information relating a grid contained in the *ETBaseData* table (ETBaseDataID) and an attribute in the *data_dict3* table (AttributeCode) into which summary information will be placed. An entity relationship diagram of these new GIS related tables in the SPATSIM-HDSF database data model is shown in Figure 2.5. Descriptions, data types and other properties of the fields in each table are documented in Appendix A.1.

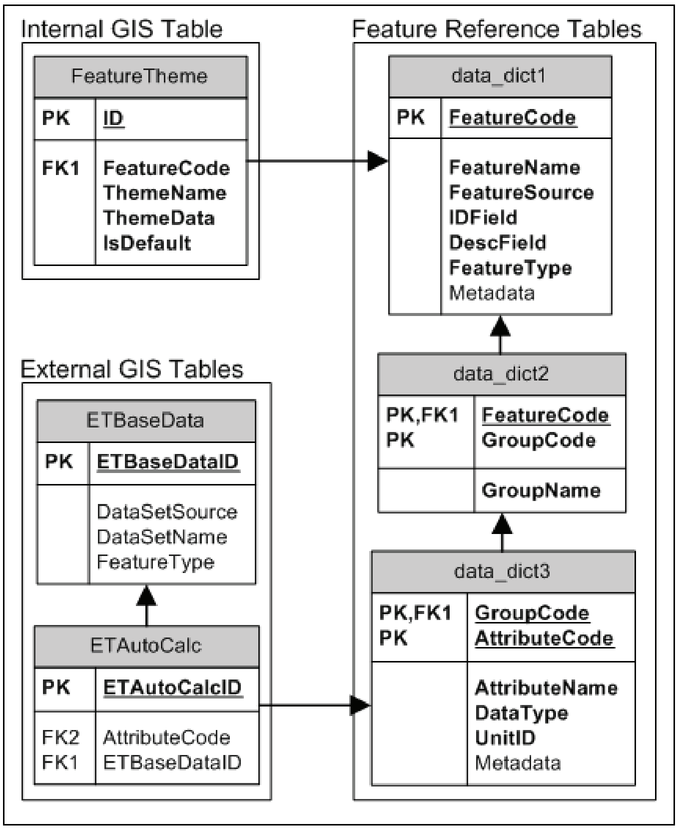


Figure 2.5: Entity relationship diagram of GIS related tables in SPATSIM-HDSF data model

The folder structure for SPATSIM-HDSF is shown in Figure 2.6. When the SPATSIM-HDSF software is installed a folder named *spatsim* is created on the user's hard drive. The initialisation file (.ini) for each SPATSIM-HDSF project is stored in the *spatsim* folder. The sub-folder named *bin* contains executable files for models and utility software to be run from SPATSIM-HDSF. The *Docs* folder contains documentation files for models to be run from SPATSIM-HDSF. The folder named *export* is the folder to which SPATSIM-HDSF data export files are saved. The *text_data* folder contains text files with *.req*, *.rqo*, *.txo* and *.txp* extensions and other files used by SPATSIM-HDSF to define model requirements, model output and array formats. The *thukela* folder is an example of a SPATSIM-HDSF project folder, which typically contains three subfolders named *cover*, *data* and *dbdata*. The *cover* folder contains shapefile files. The *data* folder may contain data input files and SPATSIM-HDSF data output files. The *dbdata* folder contains the SPATSIM-HDSF database files.

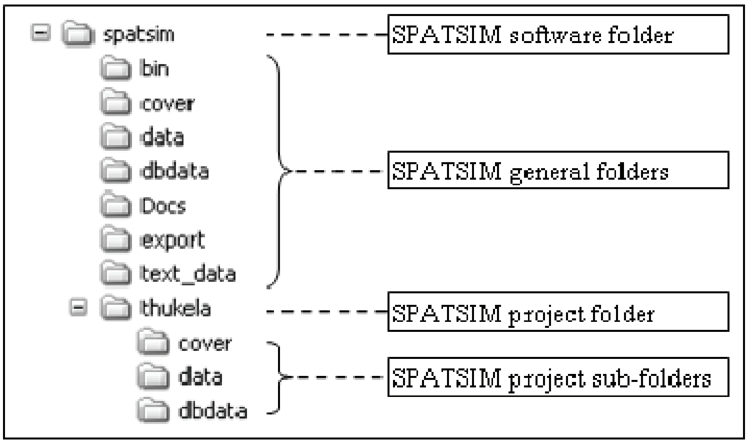


Figure 2.6: Folder structure for SPATSIM-HDSF

2.4.2 Data Access Classes

An important part of the implementation of the new SPATSIM-HDSF database data model is the development of a set of classes containing standard SPATSIM-HDSF project database access methods to create, populate, edit and query SPATSIM-HDSF databases. These SPATSIM-HDSF project database access methods will simplify database access not only in SPATSIM-HDSF, but also in the models modified to run from within the SPATSIM-HDSF framework. These classes were initially developed in the Delphi for .Net programming language and are packaged as a Dynamic Link Library (DLL) which can be used in code written in any .Net programming language. However, there are many legacy models written in a variety of programming languages that would not be able to utilise these .Net compliant classes; therefore, a similar set of Component Object Model (COM) accessible classes was developed. Further to this, there may be models such as the *ACRU* model which are written in the Java programming language. Technically it is possible to access COM objects and .Net classes in a DLL from Java code using the Java Native Interface (JNI), however, in practice this was found to be difficult. Therefore, a Java version of the data access classes was created to simplify use of these classes in Java code and provide better run-time performance. As far as possible these database access classes are based on standard Structured Query Language (SQL) queries to facilitate possible use with a selection of relational database management systems (RDBMSs).

.Net and COM Version

A Universal Modelling Language (UML) diagram of the .Net version of the SPATSIM-HDSF data access classes is shown in Figure 2.7. The namespace *Spatsim* was selected as the main namespace and the namespace *Spatsim.Data* was created under this to contain the data access classes developed. The *Spatsim.Data* namespace contains the namespaces *ADO_Ole*, *COM* and *Exceptions* and three classes *IniInfo*, *TSInfo* and *ArrayInfo*. The data access classes have been developed using ADO.Net database components and OleDb data providers. The two main data access classes *DataAccess* and *Database* have been created in the *Spatsim.Data.ADO_Ole* namespace. The classes in the *Spatsim.Data.ADO_Ole* namespace and all the classes on which they depend have been compiled into a .Net accessible DLL named *SPATSIM_DataAccess_DotNet.dll*. The *Spatsim.Data.COM* namespace contains COM accessible versions of the *DataAccess*, *IniInfo*, *TSInfo* and

ArrayInfo classes. The classes in the Spatsim.Data.COM namespace and all the classes on which they depend have been compiled into a separate COM accessible DLL named *SPATSIM_DataAccess_COM.dll*. The *Spatsim.Data.Exceptions* contains a set of Exception classes used in the data access classes. The classes in the *SPATSIM_DataAccess_DotNet.dll* and *SPATSIM_DataAccess_COM.dll* libraries are supported by documentation in Hypertext Markup Language (HTML) format.

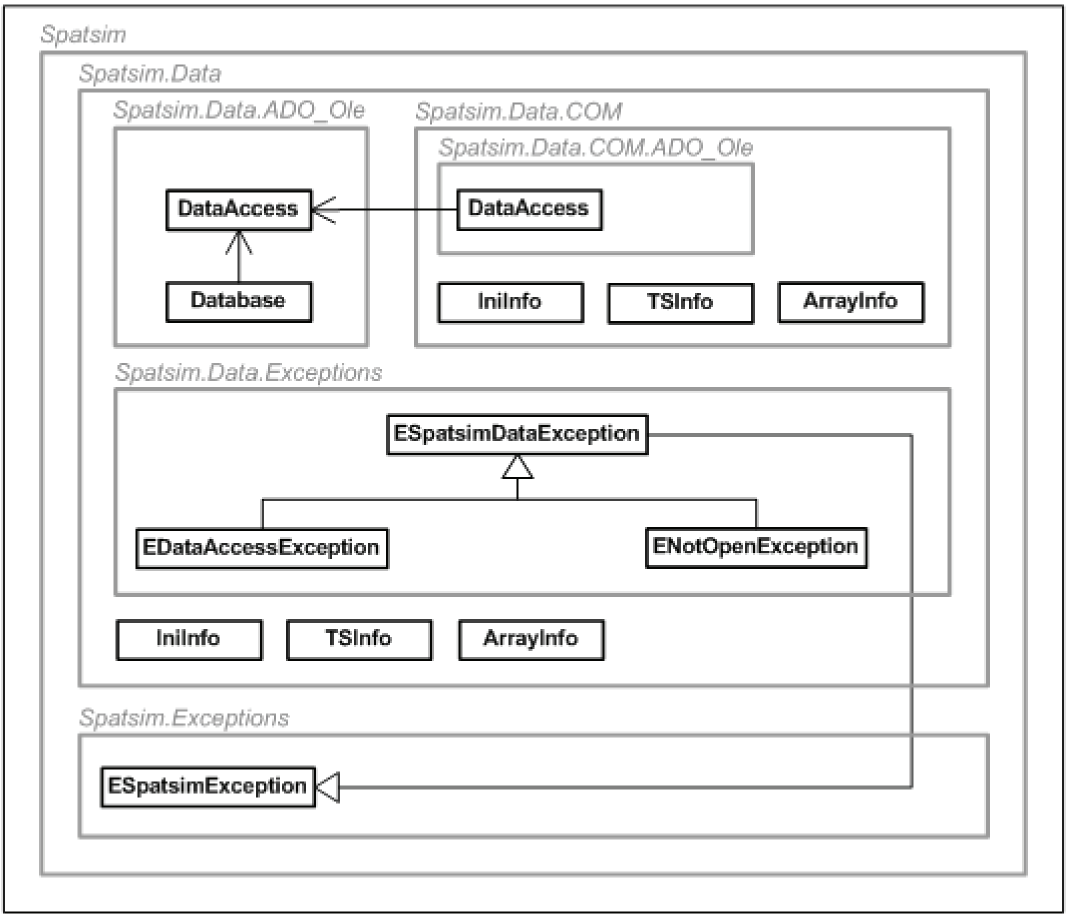


Figure 2.7: UML diagram of the .Net version of the SPATSIM-HDSF data access classes

The *DataAccess* class in the *Spatsim.Data.ADO_Ole* namespace is the core of the set of data access classes developed. This *DataAccess* class contains a comprehensive set of over 250 methods to perform the following operations on a SPATSIM-HDSF database:

- connect to a specified database
- create tables in a new database
- return SQL query strings to query a database
- insert new records into a database
- update existing records in a database
- delete records from a database.

The *Database* class in the *Spatsim.Data.ADO_Ole* namespace is a wrapper around the *DataAccess* class. This ADO.Net specific *Database* class contains an almost identical set of methods to those in the *DataAccess* class with the difference that for each of the methods in the *DataAccess* class which return SQL query strings the corresponding methods in the

Database class return a *System.Data.DataTable* object containing a result set returned by the SQL query.

The *IniInfo* class in the *Spatsim.Data* namespace contains a set of methods to read, query, modify and write the project initialisation information contained in SPATSIM-HDSF project initialisation files. The *TSInfo* class in the *Spatsim.Data* namespace contains a set of methods to read, query, modify and write the time-series data stored in the *TSData* BLOB field in the *TimeSeriesData* attribute value data table. Instances of the *TSInfo* class should be used in conjunction with the *Database* class to work with time-series data stored in SPATSIM-HDSF databases. The *ArrayInfo* class in the *Spatsim.Data* namespace contains a set of methods to read, query, modify and write the array data stored in the *ArrayData* BLOB field in the *ArrayData* attribute value data table. Instances of the *ArrayInfo* class should be used in conjunction with the *Database* class to work with array data stored in SPATSIM-HDSF databases.

The *DataAccess* class in the *Spatsim.Data.COM.ADO_Ole* namespace is a wrapper around the *Spatsim.Data.ADO_Ole.DataAccess* class and contains an almost identical set of methods to those in the *Spatsim.Data.ADO_Ole.DataAccess* class. As COM does not support overloading of method names the method names in the *Spatsim.Data.COM.ADO_Ole.DataAccess* classes have been modified with a suffix (e.g. "_1") to make each method name unique. Similarly the *IniInfo*, *TSInfo* and *ArrayInfo* classes in the *Spatsim.Data.COM* namespace are wrappers around the *IniInfo*, *TSInfo* and *ArrayInfo* classes in the *Spatsim.Data* namespace.

Java Version

A UML diagram of the Java version of the SPATSIM-HDSF data access classes is shown in Figure 2.8. The classes in the Java version of the SPATSIM-HDSF data access classes are very similar to their equivalent classes in the .Net version. The namespace *Spatsim* was selected as the main namespace and a namespace *Spatsim.Data* was created under this to contain the data access objects developed. The *Spatsim.Data* namespace contains the namespaces *JDBC* and *Exceptions* and two classes *IniInfo* and *TSInfo*. The Java version of the data access objects have been developed using the Java DataBase Connectivity (JDBC) database components. The two main data access objects *DataAccess* and *Database* have been created in the *Spatsim.Data.JDBC* namespace. The *Spatsim.Data.Exceptions* namespace contains a set of Exception classes used in the data access objects. All the classes shown in Figure 2.8 have been packaged into a Java archive file named *Spatsim_DataAccess_Java.jar*. The Java data access classes and their methods are supported by documentation in HTML format. When using the *DataAccess* and *Database* classes a driver specific to the RDBMS may be used if available otherwise the standard JDBC-ODBC drivers may be used.

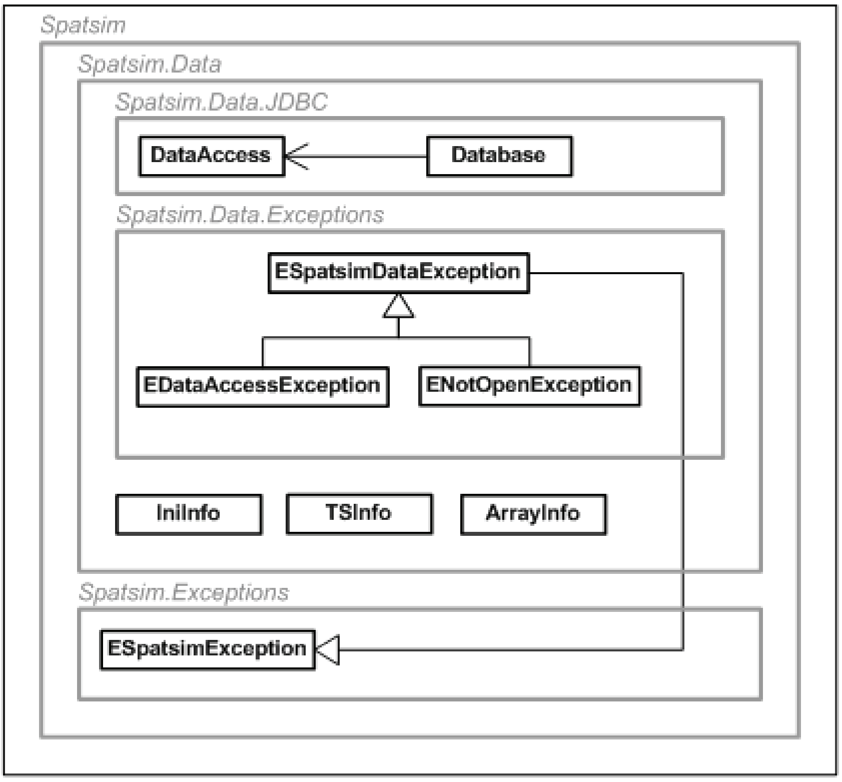


Figure 2.8: UML diagram of the Java version of the SPATSIM-HDSF data access classes

The development of the .Net, COM and Java versions of data access classes took a considerable amount of time and work, however, it is expected that these classes will be invaluable not only to the developers of the SPATSIM-HDSF framework but more importantly to anyone wanting to modify a model or modelling utility to run within the SPATSIM-HDSF modelling framework.

2.4.3 SPATSIM-HDSF User Interface Software

The purpose of the SPATSIM-HDSF user interface software is to provide a graphical user interface for SPATSIM-HDSF users to add, edit, delete, view and analyse data in a SPATSIM-HDSF database. The SPATSIM-HDSF user interface software also enables users to setup and run SPATSIM-HDSF compliant models using data in a SPATSIM-HDSF database.

The SPATSIM-HDSF user interface software was originally written in the Delphi programming language (Borland Delphi – Version 5) and as part of the HDSF project has been converted to the Delphi for .Net programming language. The conversion of the SPATSIM-HDSF user interface software to .Net was a recommendation made at the Framework Selection Workshop (Appendix B.4). The conversion to .Net will enable compatibility with other current international hydrological modelling developments using the .Net framework and ensure that SPATSIM-HDSF remains up-to-date with current software development trends. Some advantages of using Microsoft's .Net framework are:

- application components may be written in a variety of different .Net programming languages

- a consistent object-oriented programming environment
- replacement of previous Microsoft interoperability protocols
- improved data access and GUI components and controls.

As a part of the process of converting the SPATSIM-HDSF user interface software to Delphi for .Net, all the GUI forms and controls were converted from Delphi's Visual Component Library (VCL) component library to Windows Forms (WinForms), the Microsoft .NET standard library of visual components for developing graphical user interfaces (GUIs) in .Net programming languages. The reasons for converting to WinForms included problems in using ESRI MapObjects (Version 2.2) controls in forms belonging to the VCL for .Net component library and a preference for using standard .Net framework GUI components where possible.

Previously the SPATSIM user interface accessed SPATSIM databases using the Delphi Borland Database Engine (BDE) data components, and another part of the conversion to Delphi for .Net was to convert to using the .Net framework's standard ADO.Net database components and OleDb data providers to access SPATSIM-HDSF databases. The BDE provides easy access to Paradox and dBASE databases, but the conversion to the use of ADO.Net database components enables easier access from SPATSIM-HDSF to SPATSIM-HDSF databases implemented in other commonly used RDBMSs, which will be useful as ArcGIS geodatabases become more widely used. This conversion to the use of ADO.Net database components included changing the code to use the SPATSIM-HDSF data access classes developed as part of the HDSF project and described in Section 2.4.2.

2.4.3.1 Main Form

The main form in the SPATSIM-HDSF user interface has been reorganised as shown in Figure 2.9. The main form consists of a menu bar and tool bar at the top of the form, a navigation panel on the left-hand side of the form and a display panel on the right-hand side of the form. In the navigation panel the feature selector is a tree-view including a list of all the objects within a feature. Selecting one of the objects belonging to a spatial feature, highlights the object on the map in the display panel. In the navigation panel the attribute selector is a tree-view displaying attribute groups and the attributes contained within them. The tabbed page labelled *Maps* displays maps for spatial features and the tabbed page labelled *Data* displays a data grid showing attribute values for a selected spatial or non-spatial feature and attribute group. A new attribute management system has been created to add, edit or remove attributes groups and attributes. New forms, one form for each attribute data type, have been developed to enable attribute value data to be viewed and edited. The original SPATSIM data import routines have been modified to be compatible with the new database data model.

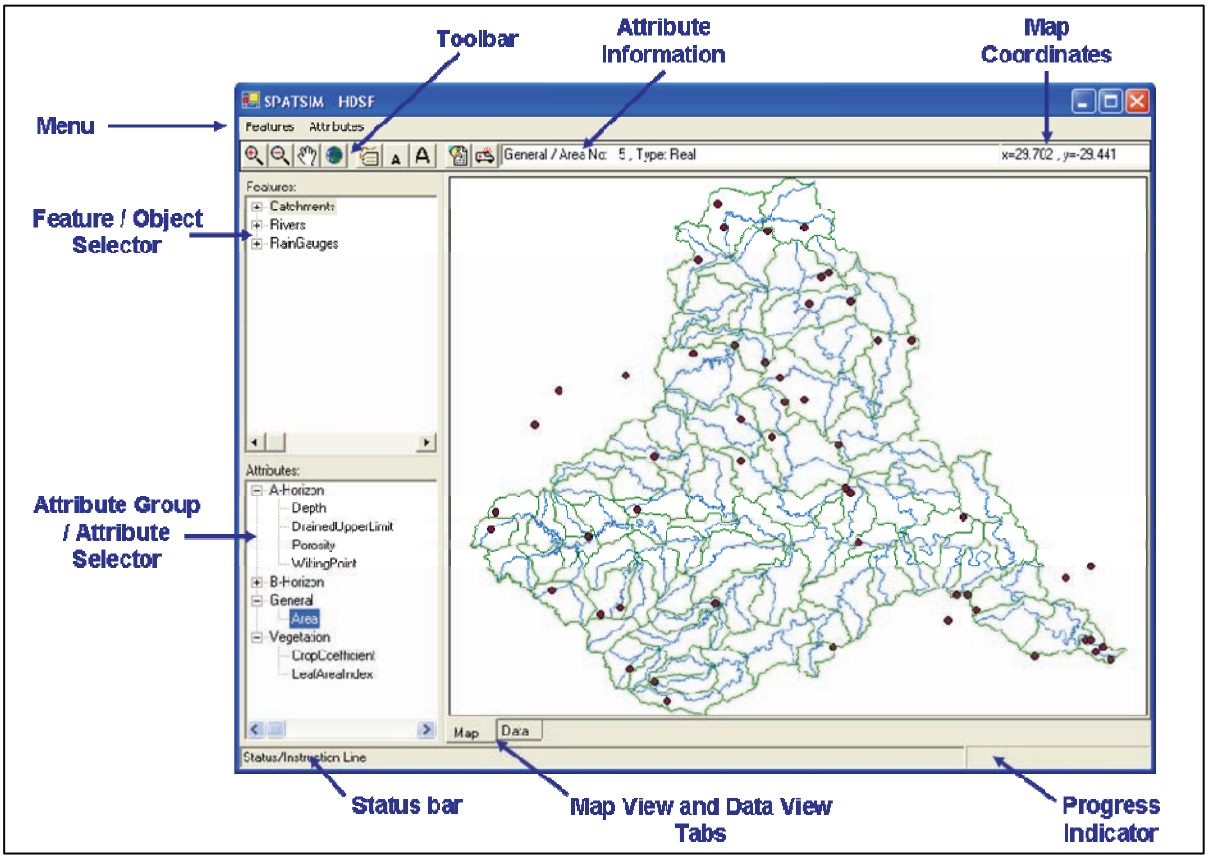


Figure 2.9: The new layout of the main form in the SPATSIM-HDSF user interface

Spatial, Non-Spatial and Image Feature Types

The new SPATSIM-HDSF database data model facilitates three different types of features: spatial, non-spatial and image. The *Features / Add / New* menu option enables the addition of new spatial, non-spatial and image features to a SPATSIM-HDSF project. When adding a spatial or a non-spatial feature the user is prompted to select which data fields associated with the feature are to be used as the identity and description fields using the form shown in Figure 2.10. Once a feature has been added to a SPATSIM-HDSF project the user may choose to hide the feature from view in the feature selection tree view control and the map view. The user may also completely remove a feature from a SPATSIM-HDSF project database. Similar to previous versions of SPATSIM the user may edit which data field associated with a spatial or non-spatial feature is to be used as the description field using the form shown in Figure 2.10. The user may edit the values in the selected description field for a feature using the form shown in Figure 2.11.

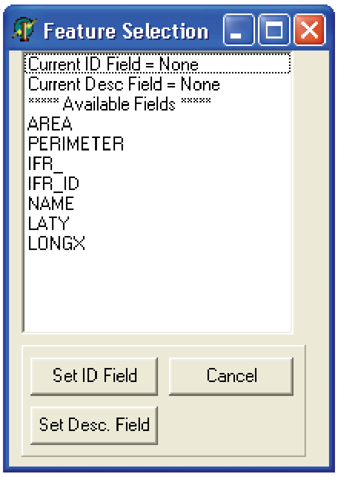


Figure 2.10: Form used to select identity and description fields for a feature

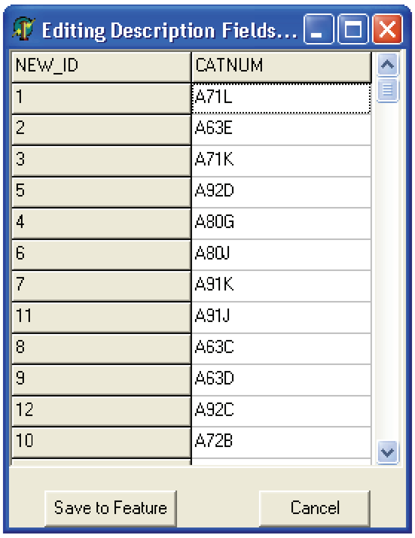


Figure 2.11: Form used to edit values in the description field for a feature

Feature and Object Selection Control

The feature and object selection control is located at the top of the navigation panel on the left-hand side of the main form. Some examples of the new feature and object selection control are shown in Figure 2.12. The advantage of the new feature and object selection control is that in addition to enabling the user to select a feature, it will enable the user to view and select a set of objects within a selected feature. The values in the feature attribute table field specified by the *DescField* field in the *data_dict1* database table for each feature are displayed as the object names in the new feature selection tree view control. In Figure 2.12 the example on the left-hand side looks similar to the feature selection list box control in the previous version of SPATSIM and shows three features with the *Catchments* feature selected. The example on the right-hand side of Figure 2.12 shows the *Catchments* feature expanded to display a list of catchment objects where catchment *V11A* has been selected. If an object in a spatial feature is selected then the corresponding object in the map view is highlighted enabling users to easily locate the selected object in the map view.

Figure 2.12: Examples of the new feature selection tree view control

Attribute Group and Attribute Selection Control

The attribute group and attribute selection control is located at the bottom of the navigation panel on the left-hand side of the main form. The new attribute group and attribute selection control includes the concept of attribute groups introduced in the new *data_dict2* table. Some examples of the new attribute group and attribute selection control are shown in Figure 2.13. The advantage of the new attribute group and attribute selection control is that it displays a list of attribute groups in the first level of the tree structure and a list of the attributes belonging to each attribute group in the second level of the tree structure. The tree structure of attributes organised into attribute groups makes it easier for the user to locate an individual attribute or sets of attributes. By default an attribute group named *General* is created for each new spatial and non-spatial feature. In Figure 2.13 the example on the left-hand side shows four attribute groups with the *General* attribute group selected. The example on the right-hand side of Figure 2.13 shows some of the attribute groups expanded and the *Area* attribute in the *General* attribute group has been selected.

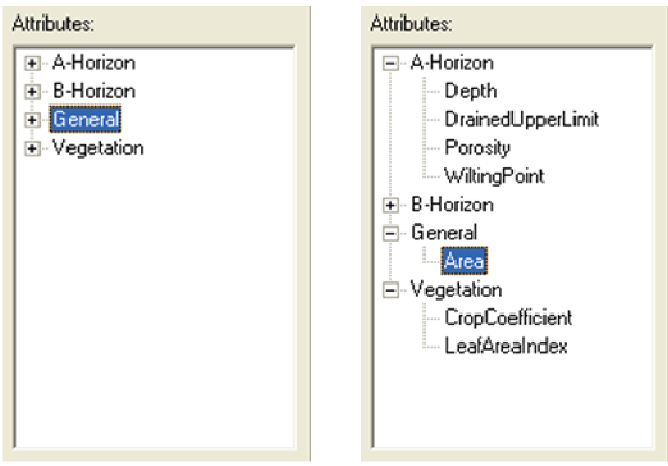


Figure 2.13: Examples of the new attribute selection tree view control

Form for Adding Attributes and Attribute Groups

With the new SPATSIM-HDSF database data model the inclusion of the concept of attribute groups, and the association of units of measure and metadata with an attribute, required the development of a new *Add Attribute* form as shown in Figure 2.14. On the *Add Attribute* form the user first selects one of the management options to add, delete or change an attribute or attribute group.

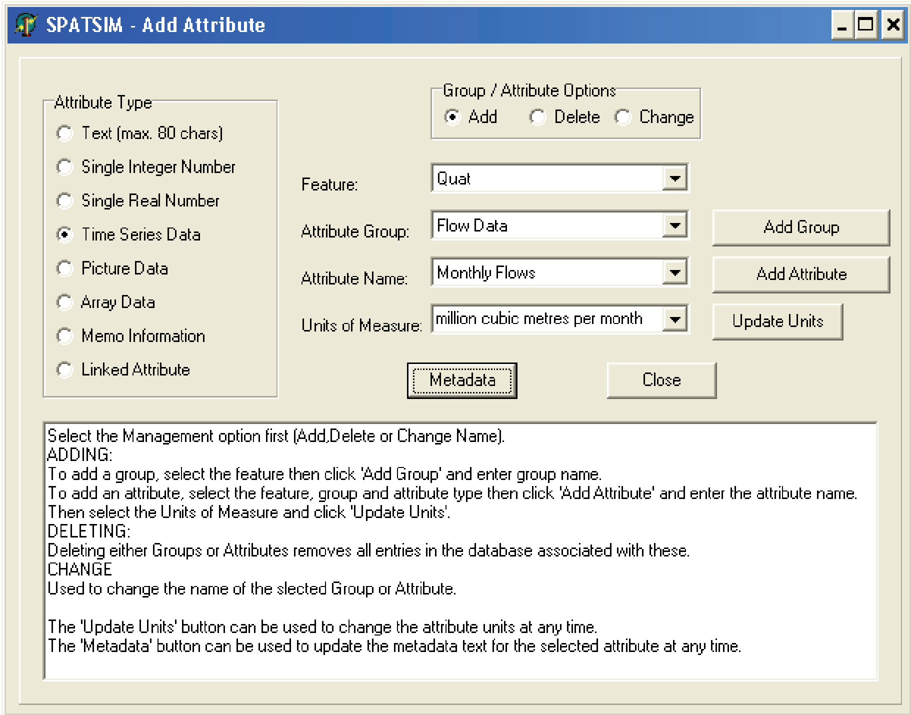


Figure 2.14: Example of the new *Add Attribute* form

To add a new attribute, the attribute type of the new attribute must be specified, and then the feature and attribute group to which the new attribute will belong, before selecting the *Add Attribute* button which enables a name to be entered for the new attribute. If a suitable attribute group does not exist for the new attribute then a new attribute group can be added by selecting the *Add Group* button which enables a name to be entered for the new attribute group. If a *Time-series Data*, *Integer Data* or *Real Data* attribute type is selected then a unit of measure may be specified by selecting a unit of measure from the *Units of Measure* combobox and then selecting the *Update Units* button to save the selected unit of measure. For other attribute types a default dimensionless unit of measure is automatically assigned to the attribute. Metadata for an attribute may be entered or edited by selecting the *Metadata* button to open the *Attribute Metadata* form shown in Figure 2.16. If the *Change* management option is selected then the *Add Group* and *Add Attribute* buttons are replaced by *Change Group* and *Change Attribute* buttons respectively enabling the names of attributes and attribute groups to be changed. If the *Delete* management option is selected then the *Add Group* and *Add Attribute* buttons are replaced by *Delete Group* and *Delete Attribute* buttons respectively. Deleting an attribute group will also delete all the attributes and associated data belonging to that attribute group from the database. Deleting an attribute will delete all the associated data belonging to that attribute from the database.

Data Tabbed Page

During the design phase of the project it was proposed that it would be an advantage to be able to be able to view, for a selected feature and attribute, the attribute values for several objects at the same time in a data grid. As seen in the main SPATSIM-HDSF screen shown in Figure 2.9 the tabbed page labelled *Map* displays maps. A tabbed page labelled *Data* has been developed to display a data grid showing attribute values for a selected feature and attribute group as shown in Figure 2.15. This will be particularly useful for non-spatial features, for which there are no maps to display. If an image type feature is selected then the *Data* tabbed page will be disabled.

The left-most column, named *Object*, in the data grid displays a list of object names based on the values in the table field specified by *DescField* for the feature in *data_dict1*. To the right of the left-most column in the data grid a set of columns is displayed, one for each attribute in the selected attribute group. For the simple attribute types *TextData*, *IntegerData* and *RealData* the data values will be displayed in the cells of the data grid. For the more complex attribute types *TimeSeriesData*, *PictureData*, *ArrayData*, *MemoData* and *LinkData* the attribute data type is displayed in the cells of the data grid. The data grid is by default disabled for editing. If the user needs to edit an attribute value then the *Edit* button should be clicked to make the data grid editable. When the data grid is editable the user may edit attribute values for *TextData*, *IntegerData* and *RealData* type attributes directly in the cells of the data grid. Alternatively all attribute data types may be edited by right-clicking with the mouse on a cell in the data grid and selecting *Edit* from the popup menu to open the appropriate attribute data value editing form. Once the user has edited one or more *TextData*, *IntegerData* or *RealData* type attribute values in the data grid, the *Save* button may be used to save edits or the *Cancel* button to reject unsaved edits and repopulate the data grid with attribute values from the database. The user is prompted to save any unsaved edits when a new attribute group is selected. The user may view some general information about an attribute displayed in the data grid by hovering the mouse over the header of an attribute column in the data grid to display the tooltip text.

Above the data grid on the *Data* tabbed page is a set of record selection controls. The *Select* controls enable the user to display a subset of the objects in the data grid by selecting a column in the data grid and specifying a condition and a constraint. For example, applying the condition "Starts With" and the constraint "V" to the *Object* column for a feature containing quaternary catchments would display all the quaternary catchment objects with a description starting with the letters "V". The default condition is *All*, which will cause all the objects in the feature to be displayed in the data grid.

By default the data grid is sorted first by the *Object* column in alphanumeric order. Using the mouse to right click on the header of an attribute column in the data grid will display an attribute popup menu. The *Sort Ascending* menu item enables the user to sort the rows in the data grid such that the values for the selected *TextData*, *IntegerData* or *RealData* type attribute column are sorted in ascending order. The *Sort Descending* menu item enables the user to sort the rows in the data grid such that the values for the selected *TextData*, *IntegerData* or *RealData* type attribute column are sorted in descending order.

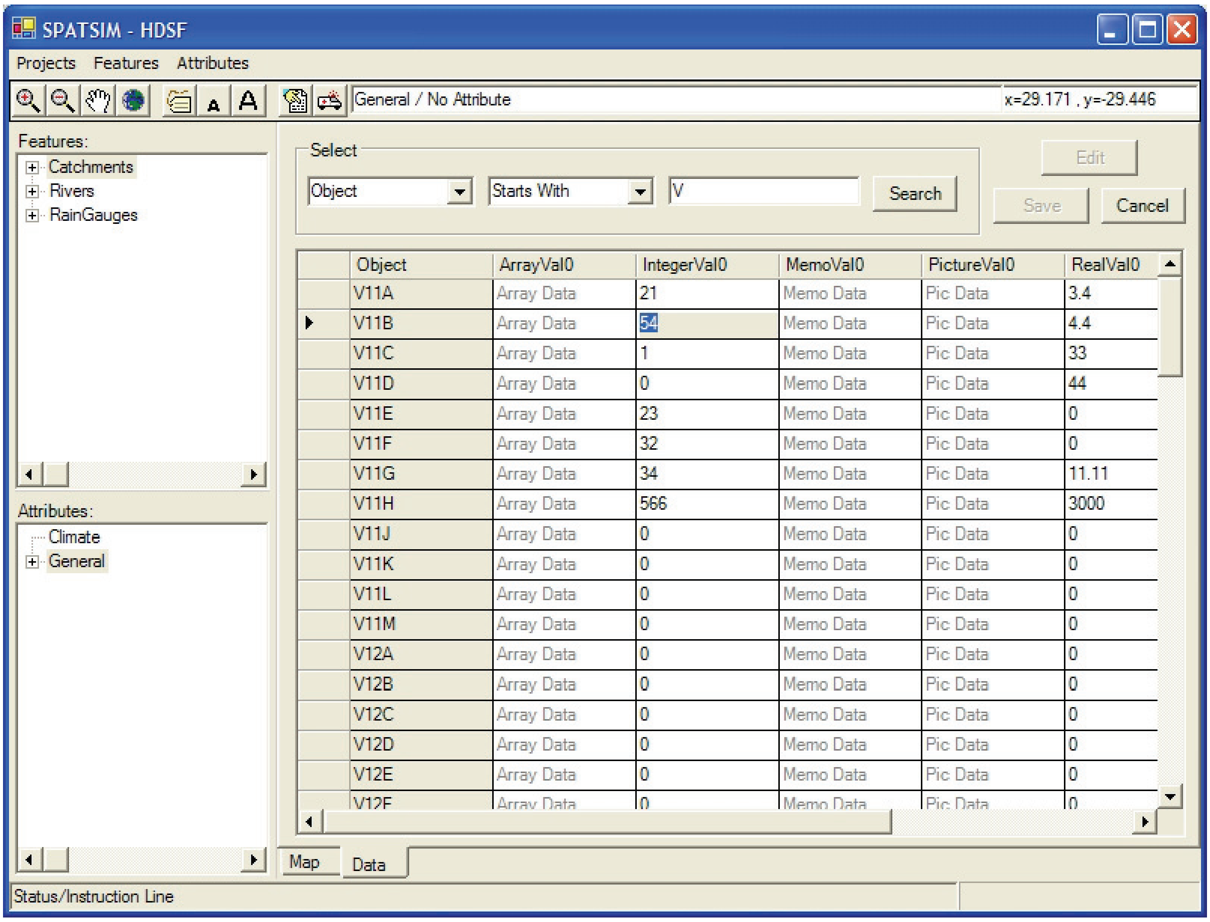


Figure 2.15: An example of the *Data* tabbed page

Feature, Attribute and Attribute Value Metadata Forms

The new *Metadata* field in *data_dict1* contains metadata describing a feature and may be populated and edited using the new modal *Feature Metadata* form shown in Figure 2.16. The *Feature Metadata* form is accessed using the new *Metadata* menu item added to the *Features* menu.

The new *Metadata* field in *data_dict3* contains metadata describing an attribute and may be populated and edited using the new modal *Attribute Metadata* form shown in Figure 2.16. The *Attribute Metadata* form is be accessed using the *Metadata* button on the *Add Attribute* form.

The new *AttValMeta* table enables metadata to be stored for individual attribute values. Attribute value metadata may be entered and edited using the new *Attribute Value Metadata* form shown in Figure 2.16. The *Attribute Value Metadata* form is accessed using the *Metadata* button on each of the attribute value data editing forms.

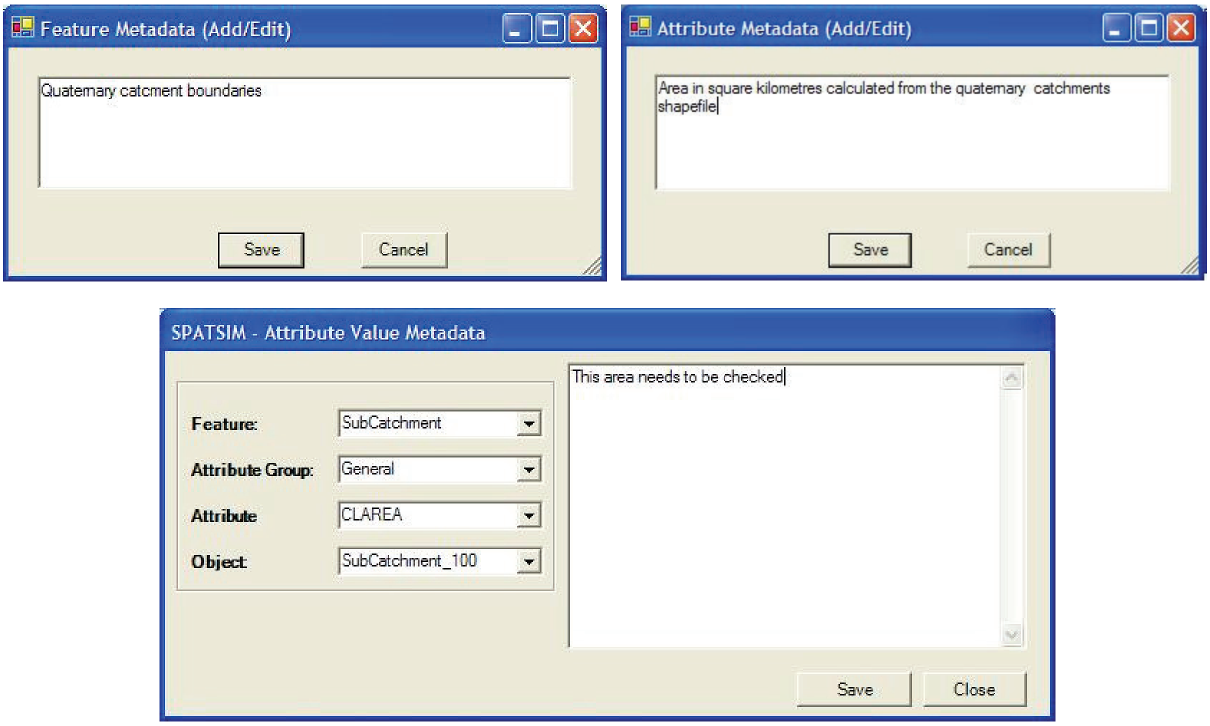


Figure 2.16: Examples of the new *Feature Metadata*, *Attribute Metadata* and *Attribute Value Metadata* forms

Importing Text, Integer and Real Attribute Data from Text Files

Similar to the previous version of SPATSIM the user may import text, integer or real data from a text file into an attribute. The text files from which attribute data is to be imported should consist of two space separated columns: one column containing values identical to the values in the description field for the feature to which the attribute belongs, and a second column containing text, integer or real data values.

2.4.3.2 Attribute Data Editing Forms

The feature attribute data editing tools in the previous version of SPATSIM have been replaced by a set of modal forms, one form for each of the attribute data types, namely: Text, Integer, Real, Time-Series, Picture, Array, Memo and Link. The new SPATSIM-HDSF data model facilitates the storage of additional information about individual process runs, by means of a set of process attribute groups and attributes. Therefore a set of process attribute data editing forms has also been created to enable process attribute data to be edited. The feature and process attribute data editing forms have been developed in the Delphi for .Net programming language and are packaged as a DLL named *SPATSIM_DataForms.dll* and are thus available to be used in any code written in a .Net programming language.

Feature Attribute Data

The new feature attribute data editing forms enable the user to view and edit attribute data for a selected feature, object and attribute. An example of the *Text Data* attribute data editing form is shown in Figure 2.17. Before displaying this form the user must first select a feature, object and Text type attribute on the main SPATSIM-HDSF form. Once the Text Data attribute data editing form is open the user may use the *Feature*, *Object*, *Attribute Group* and

Attribute controls to select other Text type attributes in other features or attribute groups. The user may edit and then save the text data associated with the selected attribute. The user may also view and edit the metadata associated with the selected attribute by clicking on the *Metadata* button.

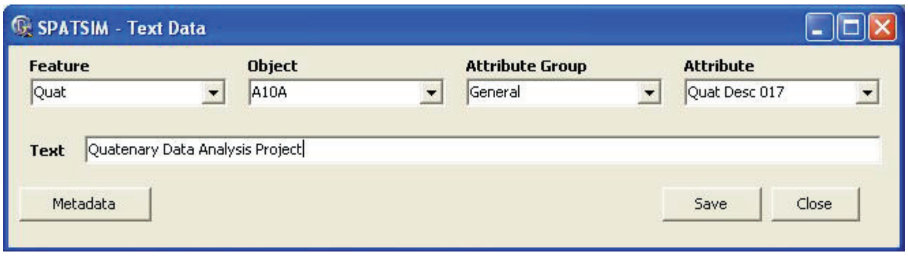


Figure 2.17: An example of the *Text Data* attribute data editing form

An example of the *Integer Data* attribute data editing form is shown in Figure 2.18. This form works in a similar manner to the *Text Data* attribute data editing form. In addition, units of measure associated with the attribute value are displayed on the *Integer Data* attribute data editing form. However, the user may not change the units of measure for the attribute value shown on this form as the units of measure are set by the user for the attribute as a whole on a different form.

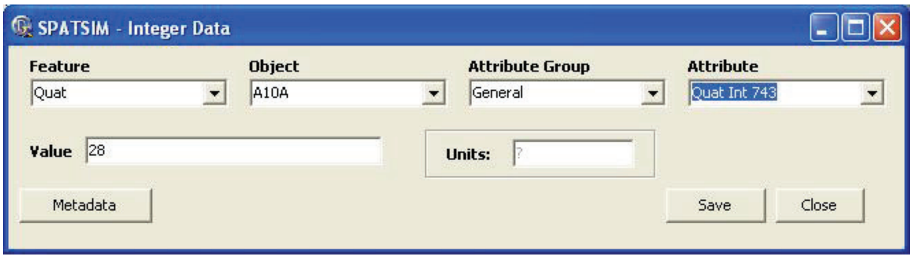


Figure 2.18: An example of the *Integer Data* attribute data editing form

An example of the *Real Data* attribute data editing form is shown in Figure 2.19. This form works in a similar manner to the *Integer Data* attribute data editing form.

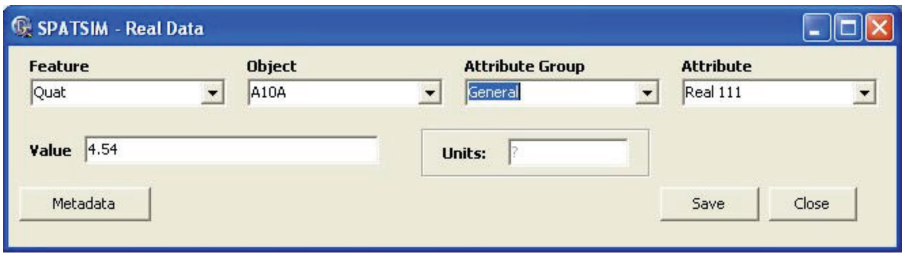


Figure 2.19: An example of the *Real Data* attribute data editing form

An example of the *Time-Series Data* attribute data editing form is shown in Figure 2.20. The feature, object and attribute selection works in a similar manner to the previously mentioned attribute data editing forms. On the left-hand side of this form there is a table displaying the date, time, value and flag information stored in the Binary Large Object (BLOB) associated with the selected Time-Series type attribute. In this table the user may edit the time-series subject to a set of appropriate rules based on the type of time-series. On the right-hand side of this form there are a set of controls that the user may use to edit information related to the selected time-series attribute.

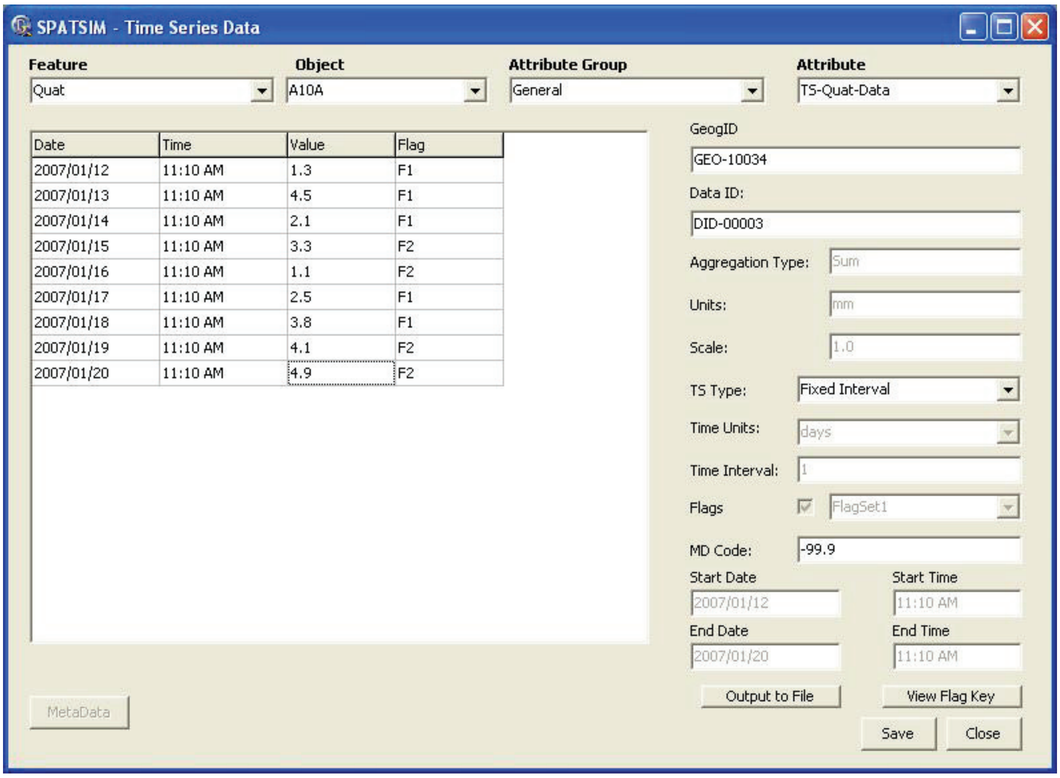


Figure 2.20: An example of the *Time-Series Data* attribute data editing form

An example of the *Picture Data* attribute data editing form is shown in Figure 2.21. This form displays the picture stored for the selected attribute and object. The title, storage date, size and type of the picture is also displayed on the form. The user may load a new picture for the selected attribute and object by clicking on the *Load Graphic* button and then selecting the relevant file using the file selection dialog. The user may also export the picture associated with the selected attribute and object by clicking on the *Export Graphic* button and then specifying a name for the file using the file save dialog.

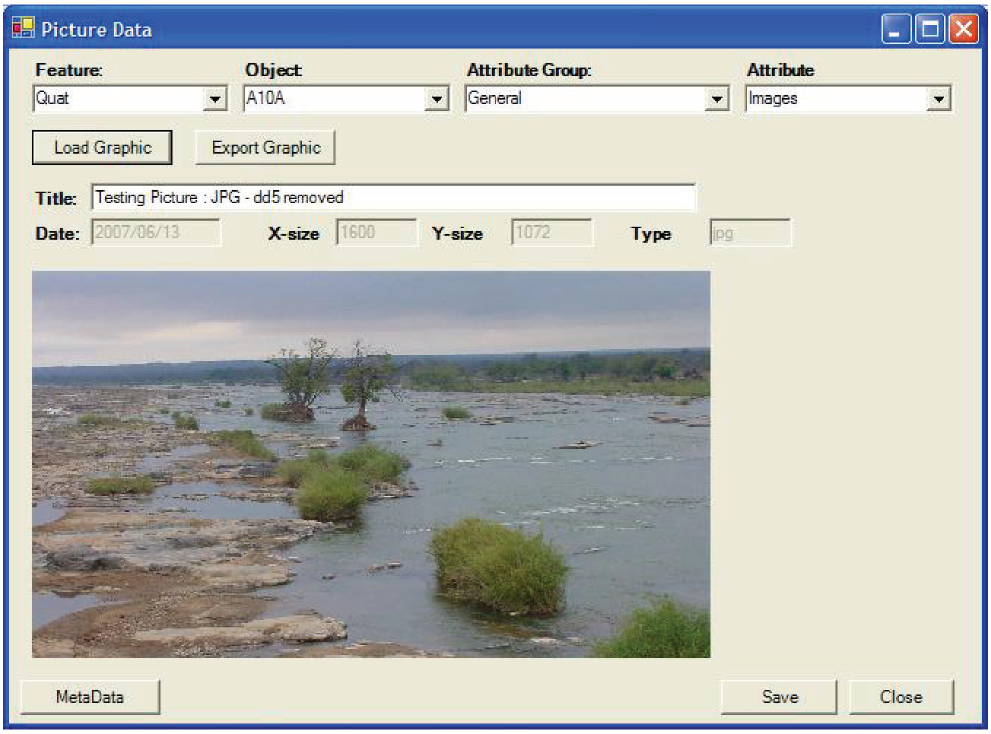


Figure 2.21: An example of the *Picture Data* attribute data editing form

An example of the *Array Data* attribute data editing form is shown in Figure 2.22. This form displays an array of numbers stored for the selected attribute and object. The title and information about the format of the array is also displayed on the form. The user may edit and then save individual data values in the selected array. Tools are provided to enable the user to import data into the selected array attribute, or to export or print array data for a selected attribute and object. Additional tools are provided to enable users to copy or scale existing array data.

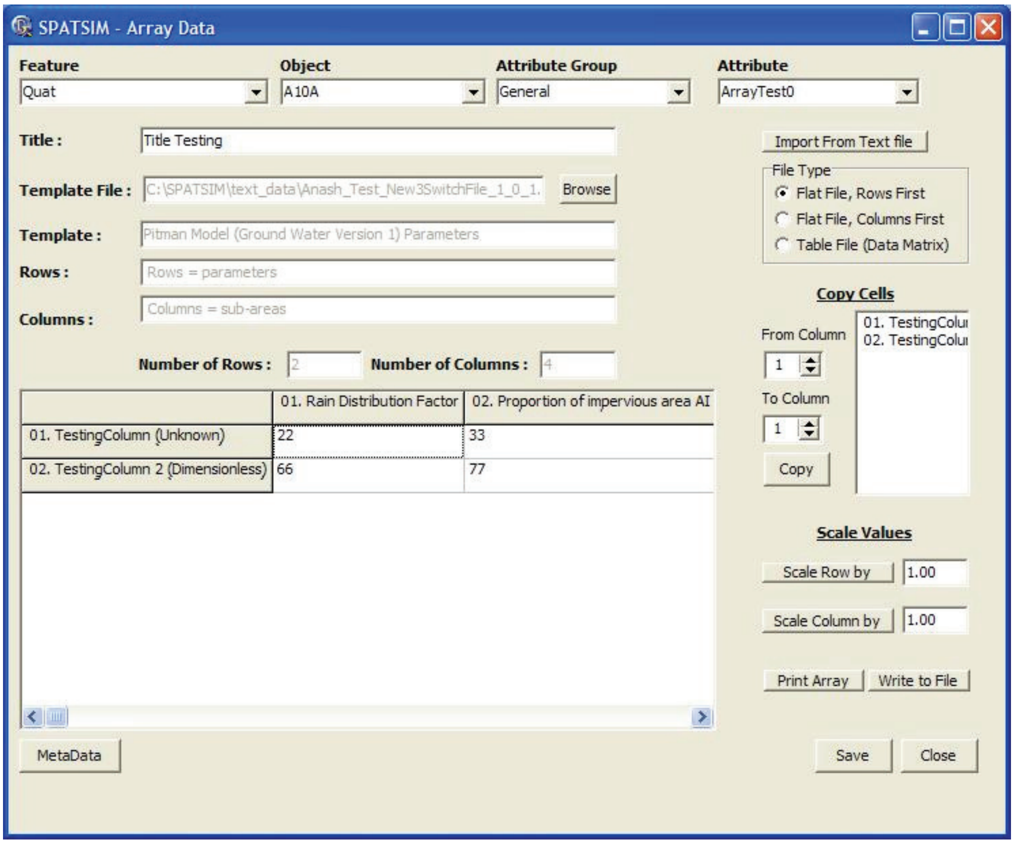


Figure 2.22: An example of the *Array Data* attribute data editing form

The *Array Data* attribute data editing form is shown in Figure 2.22 displays the data for an *ArrayData* type attribute for one selected object in a feature. The *Compare Arrays* form shown in Figure 2.23 has been developed to enable the comparison of rows or columns in an *ArrayData* type attribute for two or more objects. Once a feature, attribute group, *ArrayData* type attribute and set of feature objects has been selected, the user may select one array column or one array row to be compared. While the *Compare Arrays* form is open the user may select a different *ArrayData* type attribute using the selection controls at the top of the form.

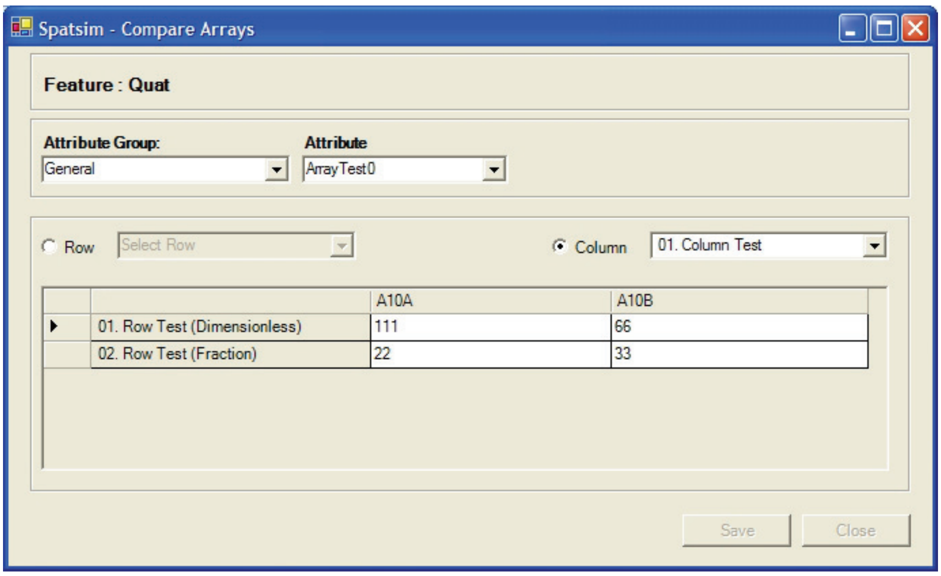


Figure 2.23: An example of the *Compare Arrays* form

An example of the *Memo Data* attribute data editing form is shown in Figure 2.24. This form displays memo text stored for the selected attribute and object. The title and memo creation date are also displayed on the form.

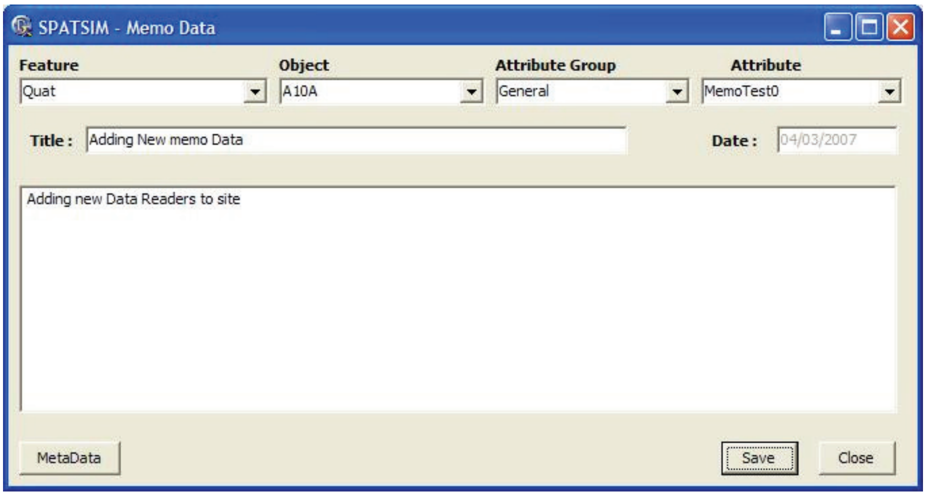


Figure 2.24: An example of the *Memo Data* attribute data editing form

An example of the *Link Data* attribute data editing form is shown in Figure 2.25. This form displays information about a linked attribute for the selected attribute and object. The purpose of linked attributes is to prevent replication of data and enables data stored in the database for one attribute and object to be used by one or more other attributes and/or objects. For the selected attribute and object the user may specify the source attribute and object containing the data to be referenced. Both the selected attribute and the source attribute must be of the same data type.

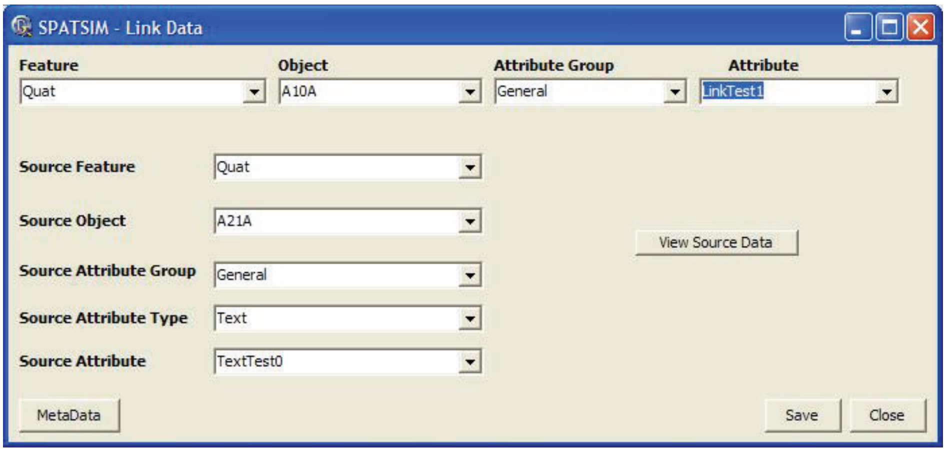


Figure 2.25: An example of the *Link Data* attribute data editing form

Process Attribute Data

The new SPASIM database data model described in Section 2.4.1 includes tables that enable the storage of attribute data for a process in a similar manner to which attribute data is stored for a feature. A set of process attribute data editing forms were developed to enable the user to view and edit attribute data for a selected process run and attribute. The process attribute data editing forms are very similar to the feature attribute data editing forms and therefore, have not been described in this report.

2.4.3.3 Internal GIS Tools

The SPATSIM-HDSF user interface software contains a set of built-in GIS tools providing a spatial view of model input and output parameters. These internal GIS tools are intended to provide a basic set of GIS tools enabling users to setup and run hydrological models without requiring expensive commercial GIS software. The internal GIS tools in SPATSIM-HDSF have been developed using Environmental Software Research Institute (ESRI) MapObjects GIS software development components and use ESRI shape files as the storage format for geographical data.
The internal GIS functionality currently provided in SPATSIM-HDSF includes the following:

- Display spatial features in the *Map* tabbed page
- Set up themes and display these in a legend
- Zoom-in, zoom-out, zoom-to-previous-extent, pan and zoom-to-full-extent tools
- Show, hide and resize feature labels
- Rendering of features as a theme using attribute values
- Selection of feature objects on the map for use in other SPATSIM-HDSF tools and models.

A significant addition to the internal GIS tools as part of this project has been the development of functionality to set up themes for a feature, use of these themes for rendering based on feature attribute values and the display these themes in a legend display control. The main form in the SPATSIM-HDSF user interface with the legend display control turned on is shown in Figure 2.26. The legend display button is used to toggle the legend display control on and off so that the map display area can be maximised when required. One or more themes may be set up for each spatial feature in a SPATSIM-HDSF database. The setup information for each theme is stored in the SPATSIM-HDSF database using the *FeatureTheme* table. To add, edit or remove a theme for a spatial feature the feature should be selected in the feature selection control, then right-clicking the mouse and selecting the *Edit Theme* option to display the *Edit Themes* form. An example of the *Edit Themes* form is shown in Figure 2.27. The *Edit Themes* form enables themes to be added, edited or removed for the selected spatial feature. For each theme a name, type and data source may be specified. The three options for theme type are: *SingleSymbol*, *ValueMap* and *ClassBreak*. The data source for a theme is specified by selecting an attribute group and then an attribute belonging to the selected feature. Data sources include SPATSIM-HDSF database attributes with *TextData, IntegerData*, *RealData* or *ArrayData* data types. Each theme may include one or more symbols, and these symbols may be edited by using the mouse to double-click on an existing symbol to display the *Symbol Properties* form, an example of which is shown in Figure 2.28. The *Symbol Properties* form enables a name, value, style, size, colour and outline colour to be set for the selected theme symbol. The zoom tools were also further developed in this project to include a zoom-to-previous-extent option and being able to zoom using the scroll-wheel on a mouse.

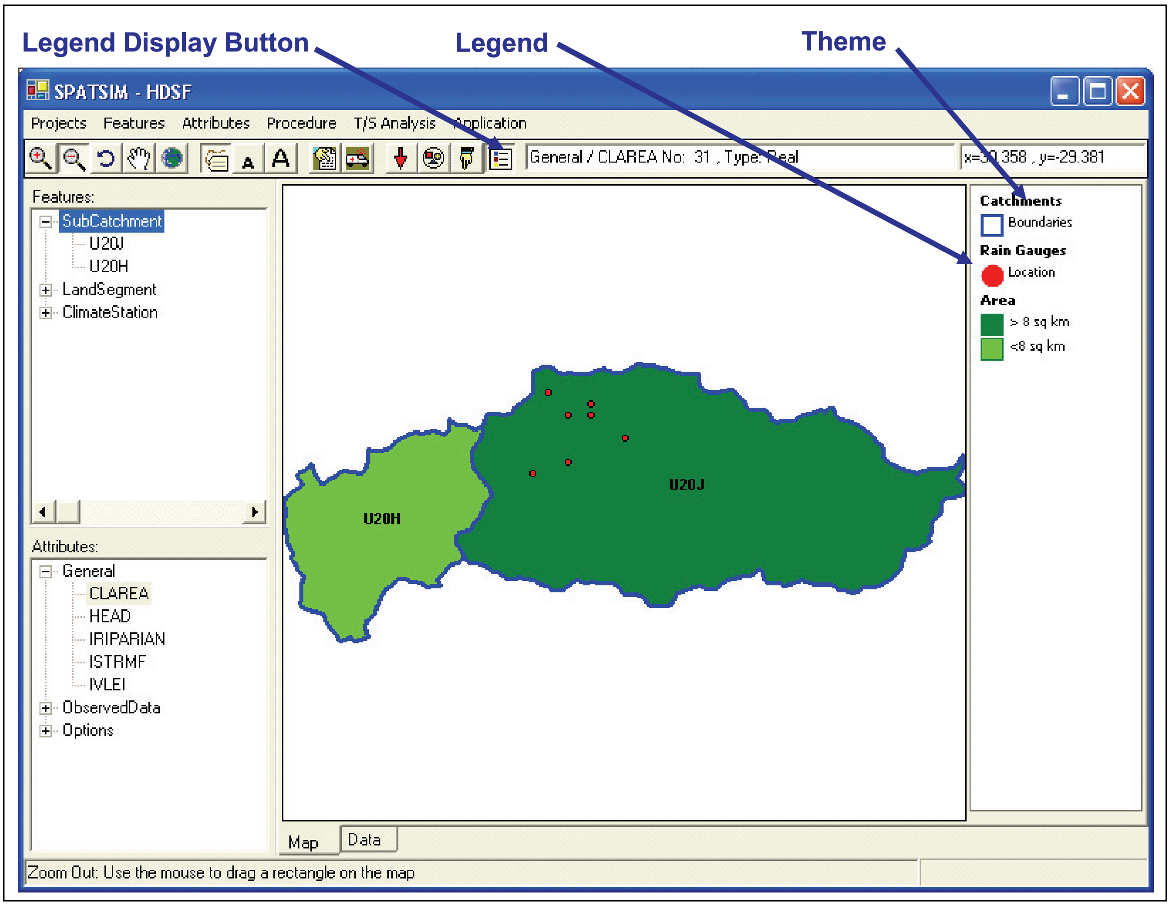


Figure 2.26: The main form in the SPATSIM-HDSF user interface with the legend displayed

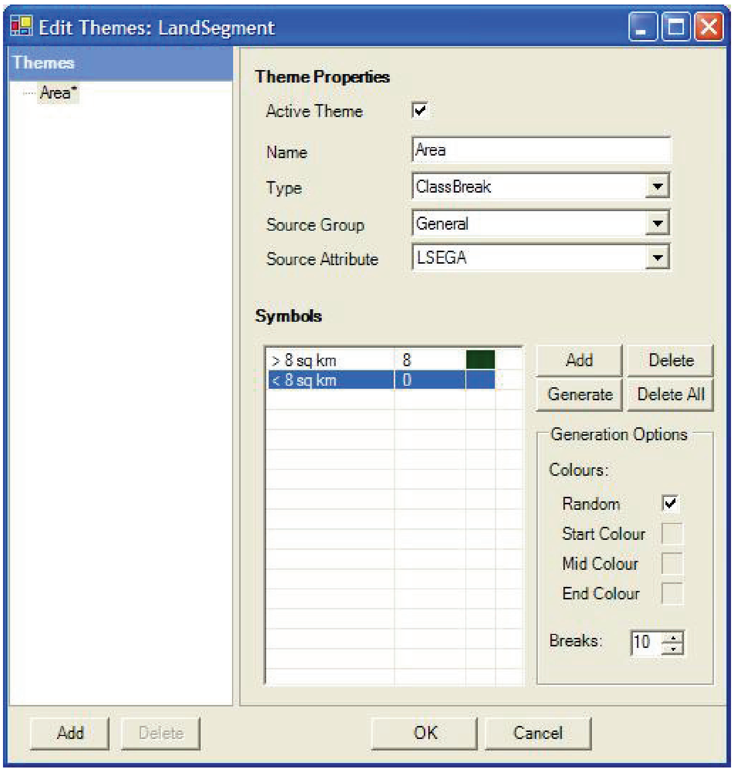


Figure 2.27: The internal GIS *Edit Themes* form

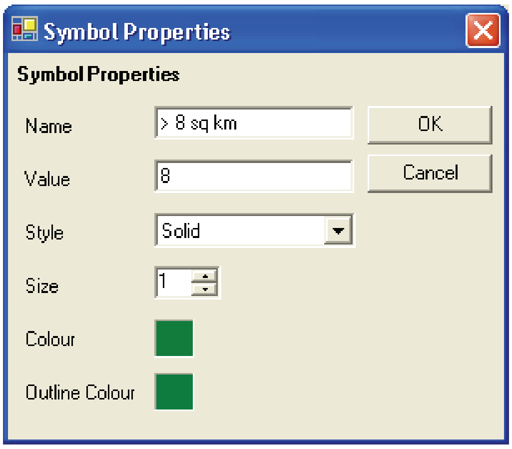


Figure 2.28: The internal GIS *Symbol Properties* form

2.4.3.4 Time-Series Analysis Tools

One of the requirements of the HDSF was that it should include a set of tools for analysis and display of model input and output data. There are two main types of data analysis and display tools required in the HDSF: spatial data tools and time-series data tools. Spatial data analysis tools have been provided in the form of the internal GIS tools (Section 2.4.3.3) and the external GIS tools (Section 2.4.4) as described in this report. This section of the report will deal with the development of a set of tools for the analysis of time-series data including the graphing and comparison of time-series data and associated statistical analyses.

The previous version of SPATSIM included access to the external TSOFT application via the *Application* menu. The TSOFT application contained tools to display graphs of time-series data stored in a SPATSIM database or in SPATSIM binary output files and to perform a range of time-series analyses on the graphed time-series. A Microsoft Excel based data analysis application named *ACRUView* was developed by the School of Bioresources Engineering and Environmental Hydrology (BEEH) to analyse time-series data output from the *ACRU* model. As part of the HDSF project the tools in the TSOFT and *ACRUView* applications have been consolidated into a set of tools for the display and analysis of time-series data. These time-series analysis tools have been developed in the Delphi for .Net programming language and are packaged as a DLL named *TSAnalysis.dll* and are thus available to be used in any code written in a .Net programming language. Four of these consolidated time-series analysis tools have been developed, namely *Graphs*, *Statistics*, *Comparative Statistics* and *Frequency Analysis*, and these are accessed via the *T/S Analysis* option on the main menu.

The previous version of SPATSIM also included several built-in data analysis tools accessed from the *Procedure* menu. To date the two procedures that have been converted to work with the new SPATSIM-HDSF database data model are those that are most commonly used and were developed for specific users of SPATSIM, who also contributed funds for the development. These are the *T/S Summary* and *Drought Indices* procedures which can be accessed via the *Procedure* option on the main menu.

Time-Series Analysis - Graphs

The purpose of the *Time-series Analysis – Graphs* tool is to enable one or more sets of time-series data to be plotted for visual analysis by the user. An example of the new *Time-series Analysis – Graphs* form is shown in Figure 2.29. The form contains two sets of graph axes separated by an adjustable splitter. On each set of graph axes one or more time-series may be plotted as either line or bar graphs and a legend is created. On each set of graph axes a graph title and a y-axis title may be set. For each time-series the colour of the line or bar may be set. Each set of graph axes includes controls to zoom and scroll to selected portions of a time-series.

A new time-series is added to the active set of graph axes using the *Time-series Analysis - Graphs - Add Series* form shown in Figure 2.30. The time-series data to be added may be sourced from the SPATSIM-HDSF project database or from a SPATSIM binary file containing model output generated by one of the models run from SPATSIM-HDSF. If the source is a SPATSIM-HDSF database then the time-series data to be plotted is selected by selecting the relevant feature, object, attribute group and attribute. A short name should be specified for each time-series added for identification and display purposes, the default is the attribute name followed by the object name. A start date and an end date within the date range of the selected time-series may be specified if it is required that a shorter section of the selected time-series be plotted, the default is the start and end date of the time-series. The selected time-series may be plotted as either a line or a bar graph. A scaling factor may be applied to the time-series data to be plotted, the default is a factor of 1.0. The selected time-series may be plotted as a single time-series of values or as multiple time-series, where selected months and years are plotted as separate time-series.

Once the user has set up a set of graphs, this graph setup may be saved to an analysis template file so that the same graph setup may be opened at another time without the user needing to set up the same set of graphs from scratch.

Figure 2.29: Example of the new *TimeSeries Analysis – Graphs* form

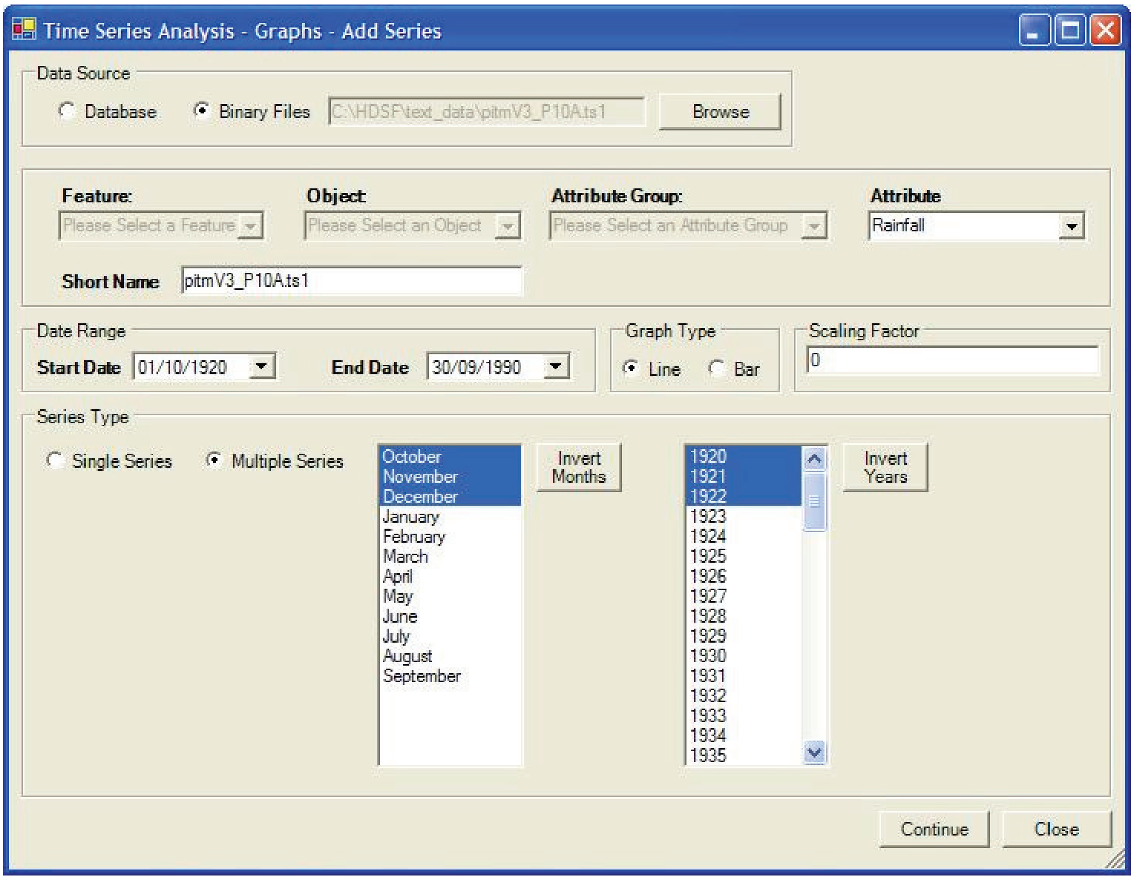


Figure 2.30: Example of the new *Time-series Analysis – Graphs – Add Series* form

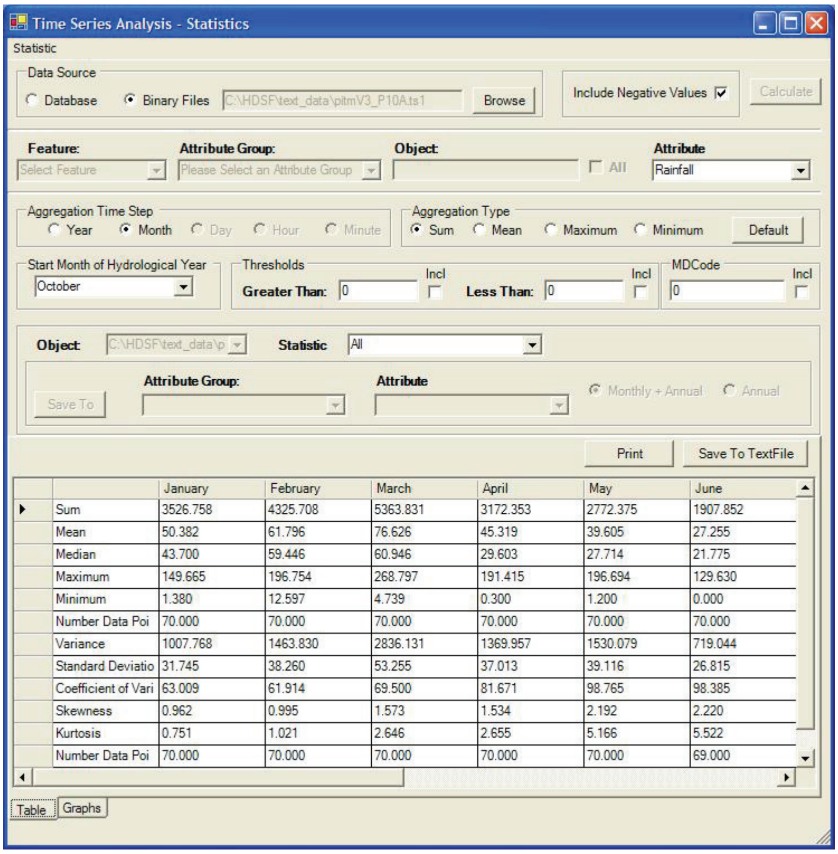
Time-series Analysis - Statistics

The purpose of the *Time-series Analysis – Statistics* tool is to enable a set of useful month-by-month and annual statistics for a selected time-series to be calculated, graphed and saved. The following statistics are calculated:

- sum
- mean
- median
- minimum
- maximum
- number of data points
- standard deviation
- variance
- coefficient of variation
- skewness
- kurtosis
- number of values less than a user specified threshold value
- number of values greater than or equal to a user specified threshold value.

An example of the new *Time-series Analysis – Statistics* form is shown in Figure 2.31. This form consists of two main sections, the time-series selection and options panel at the top, and the statistics *Table* and *Graphs* tabbed pages at the bottom. The time-series data to be analysed may be sourced from the SPATSIM-HDSF project database or from a SPATSIM binary file containing model output generated by one of the models run from SPATSIM-HDSF. If the source is a SPATSIM-HDSF database then the time-series data to be analysed is selected by selecting the relevant feature, object, attribute group and attribute, or alternatively the statistical analysis may be done for all the objects in the selected feature. The time-series data to be analysed may be aggregated to a coarser time step by selecting an aggregation time step and aggregation type relevant to the selected time-series. The start month for the hydrological year may be selected, the default is October. The user may also specify upper and lower threshold values so that the number of values in the time-series that fall outside these thresholds may be calculated. A missing data code may be specified so that these data points are not included in the analysis. Once a statistical analysis has been performed by selecting the *Calculate* button, the user may select the object for which the results of the statistical analysis are to be shown in the *Table* and *Graphs* tabbed pages at the bottom of the form. The statistics for the selected object may be printed, saved to a text file or saved to a suitable *ArrayData* type attribute in the SPATSIM-HDSF database. If the *Table* tab is active then one or all of the calculated statistics for the selected object may selected to be displayed in the data grid at the bottom of the form as shown in Figure 2.31. If the *Graphs* tab is active then one of the calculated statistics for the selected object may selected to be displayed in a graph at the bottom of the form as shown in Figure 2.32.

Once the user has set up a statistical analysis for a selected attribute and object, or objects, the setup for this analysis may be saved to an analysis template file so that the same analysis setup may be opened at another time without the user needing to set up the same analysis from scratch.



	January	February	March	April	May	June
Sum	3526.758	4325.708	5363.831	3172.353	2772.375	1907.852
Mean	50.382	61.796	76.626	45.319	39.605	27.255
Median	43.700	59.446	60.946	29.603	27.714	21.775
Maximum	149.665	196.754	268.797	191.415	196.694	129.630
Minimum	1.380	12.597	4.739	0.300	1.200	0.000
Number Data Poi	70.000	70.000	70.000	70.000	70.000	70.000
Variance	1007.768	1463.830	2836.131	1369.957	1530.079	719.044
Standard Deviatio	31.745	38.260	53.255	37.013	39.116	26.815
Coefficient of Vari	63.009	61.914	69.500	81.671	98.765	98.385
Skewness	0.962	0.995	1.573	1.534	2.192	2.220
Kurtosis	0.751	1.021	2.646	2.655	5.166	5.522
Number Data Poi	70.000	70.000	70.000	70.000	70.000	69.000

Figure 2.31: Example of the new *Time-series Analysis – Statistics* form

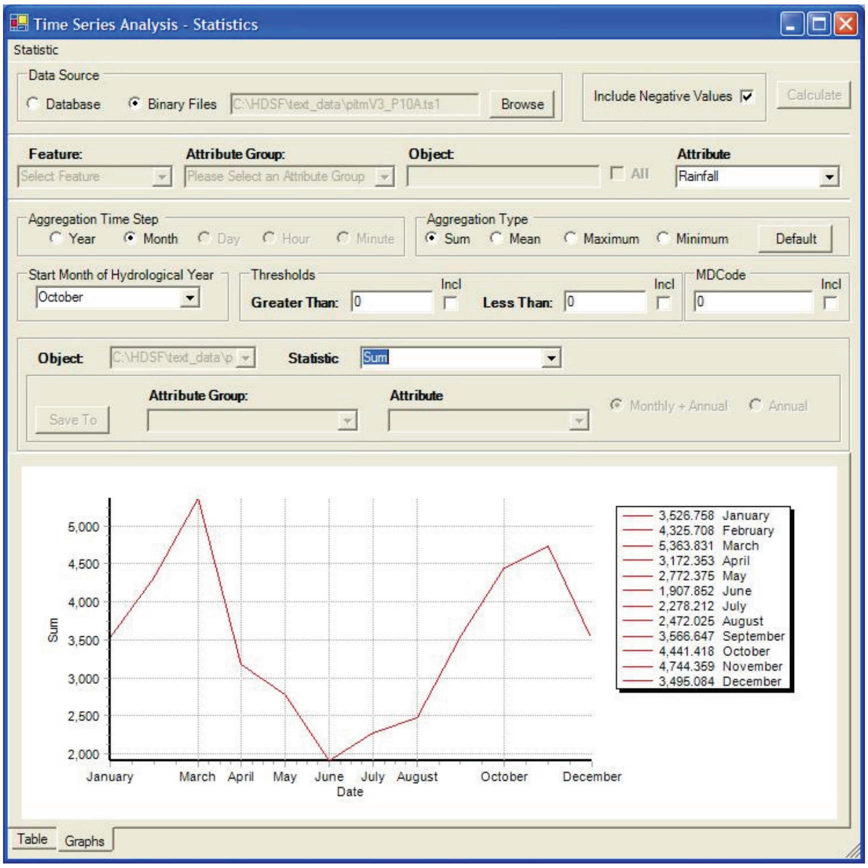


Figure 2.32: Example of the *Graphs* tab on the new *Time-series Analysis – Statistics* form

Time-series Analysis - Comparative Statistics

The purpose of the *Time-series Analysis – Comparative Statistics* tool is to enable the calculation of conservation and regression statistics between two time-series, typically time-series of observed and simulated data, and to create a scatter plot of the data in these two time-series. The following conservation statistics are calculated:

- sum (Attribute A and Attribute B)
- average error
- mean (Attribute A and Attribute B)
- percentage difference between means
- t statistic for comparing means
- variance (Attribute A and Attribute B)
- percentage difference between variances
- standard deviation (Attribute A and Attribute B)
- percentage difference between standard deviations
- standard error (Attribute A and Attribute B)
- percentage difference between standard errors
- total root mean square error (RMSE)
- coefficient of variation (Attribute A and Attribute B)
- percentage difference between coefficients of variation
- skewness coefficient (Attribute A and Attribute B)
- percentage difference between skewness coefficients
- kurtosis coefficient (Attribute A and Attribute B)

- percentage difference between kurtosis coefficients
- t statistic for correlation testing.

The following regression statistics are calculated:
- correlation coefficient (Pearson's r)
- regression coefficient (slope)
- regression intercept
- total sum of squares (SST)
- sum of squares due to regression (SSR)
- residual sum of squares (SSE)
- computer rounding error (SST - (SSR + SSE))
- coefficient of determination (R^2)
- coefficient of efficiency
- coefficient of agreement.

An example of the new *Time-series Analysis – Comparative Statistics* form is shown in Figure 2.33. This form consists of two main sections, the time-series selection and options panel at the top, and the *Conservation Statistics*, *Regression Statistics* and *Graph* tabbed pages at the bottom. The time-series data to be analysed may be sourced from the SPATSIM-HDSF project database or from a SPATSIM binary file containing model output generated by one of the models run from SPATSIM-HDSF. If the source is a SPATSIM-HDSF database then the time-series data to be analysed is selected by selecting the relevant feature, object and attribute group, and then selecting two the attributes containing the time-series to be compared. The time-series data to be analysed may be aggregated to a coarser time step by selecting an aggregation time step and aggregation type relevant to the selected time-series. A missing data code may be specified for each of the two selected attributes so that these data points are not included in the analysis. Once an analysis has been performed by selecting the *Calculate* button, the user may view the results of the analysis in the *Conservation Statistics*, *Regression Statistics* and *Graph* tabbed pages at the bottom of the form. The results of the analysis may be printed or saved to a text file. An example of the *Conservation Statistics* tab is shown in Figure 2.33. An example of the *Regression Statistics* tab is shown in Figure 2.34. An example of the *Graph* tab is shown in Figure 2.35.

Once the user has set up an analysis for a selected object and attributes, the setup for this analysis may be saved to an analysis template file so that the same analysis setup may be opened at another time without the user needing to set up the same analysis from scratch.

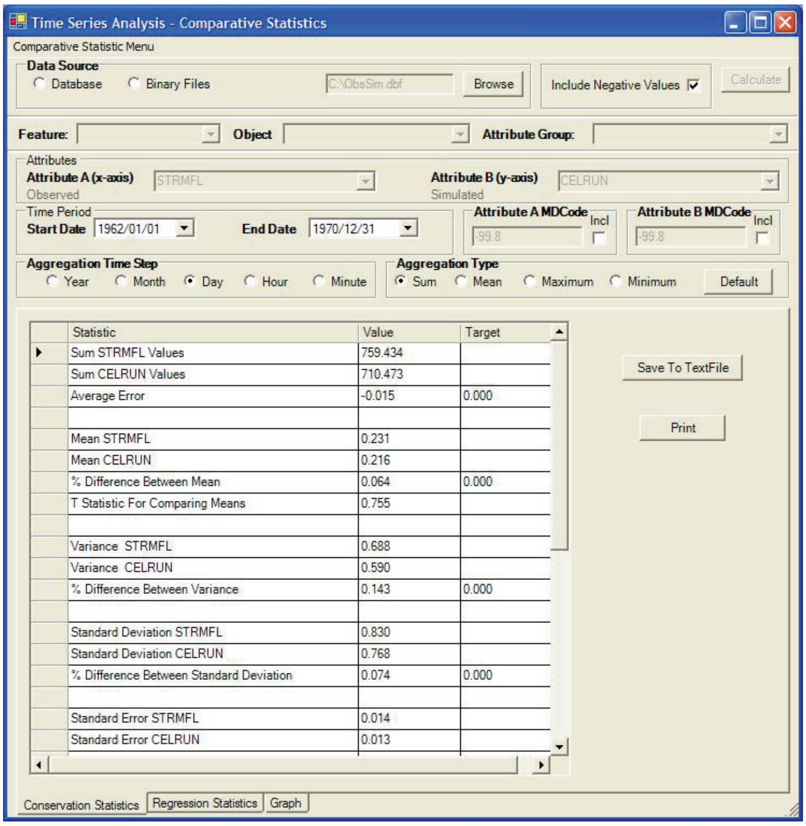


Figure 2.33: Example of the new *Time-series Analysis – Comparative Statistics* form

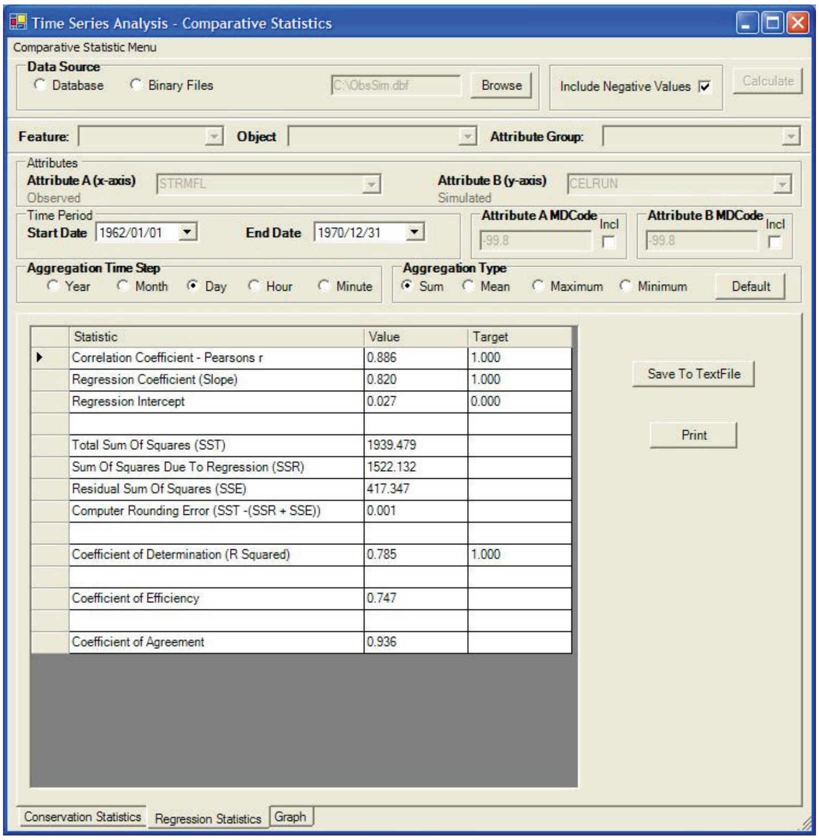


Figure 2.34: Example of the *Regression Statistics* tab on the new *Time-series Analysis – Comparative Statistics* form

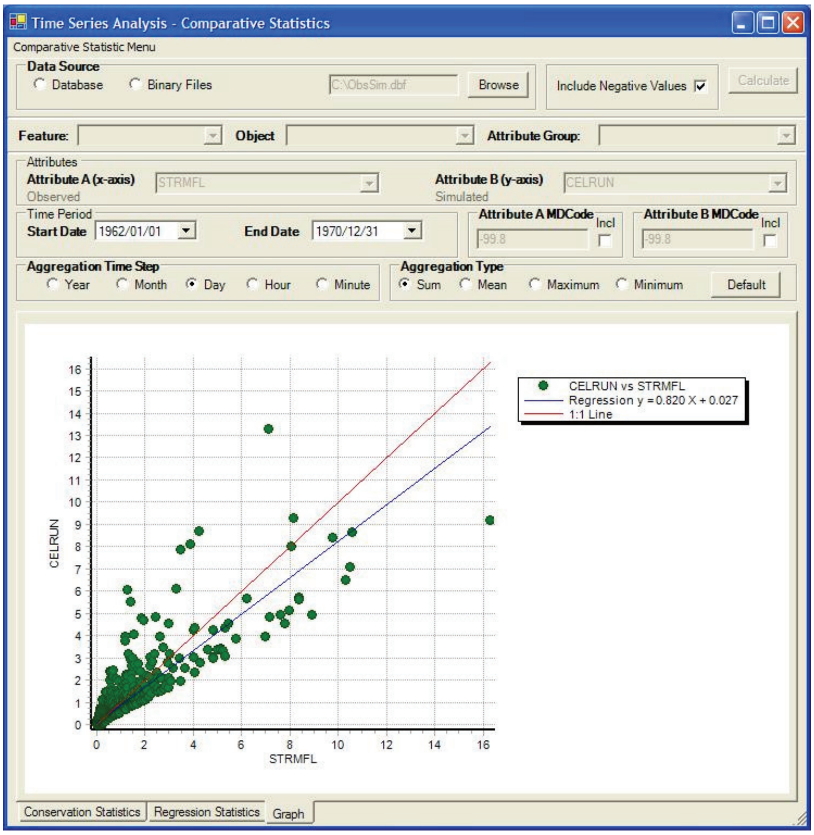


Figure 2.35: Example of the *Graph* tab on the new *Time-series Analysis – Comparative Statistics* form

Time-series Analysis - Frequency Analysis

The purpose of the *Time-series Analysis – Frequency Analysis* tool is to enable monthly, partial and annual frequency analyses to be performed on a selected time-series and the results displayed as a table of values or a set of graphs. An example of the new *Time-series Analysis – Frequency Analysis* form is shown in Figure 2.36. This form consists of two main sections, the time-series selection and options panel at the top, and the frequency analysis results *Table* and *Graphs* tabbed pages at the bottom. The time-series data to be analysed may be sourced from the SPATSIM-HDSF project database or from a SPATSIM binary file containing model output generated by one of the models run from SPATSIM-HDSF. If the source is a SPATSIM-HDSF database then the time-series data to be analysed is selected by selecting the relevant feature, object, attribute group and attribute. The time-series data to be analysed may be aggregated to a coarser time step by selecting an aggregation time step and aggregation type relevant to the selected time-series. The start month for the hydrological year may be selected, the default is October. The user may also specify upper and lower threshold values so that values that fall outside these thresholds may be excluded from the analysis. A missing data code may be specified so that these data points are not included in the analysis. The user may select whether to calculate probabilities of exceedance or non-exceedance. The user may also select which months of the year to be include in a partial analysis. Once a frequency analysis has been performed by selecting the *Calculate* button, the user may view the results of the analysis in the *Table* and *Graphs* tabbed pages at the bottom of the form. The results of the frequency analysis may be printed, saved to a text file or saved to a suitable *ArrayData* type attribute in the SPATSIM-HDSF database. An example of the *Table* tab is shown in Figure 2.36. An example of the *Graphs* tab is shown in Figure 2.37.

Once the user has set up a frequency analysis for a selected attribute and object, or objects, the setup for this analysis may be saved to an analysis template file so that the same analysis setup may be opened at another time without the user needing to set up the same analysis from scratch.

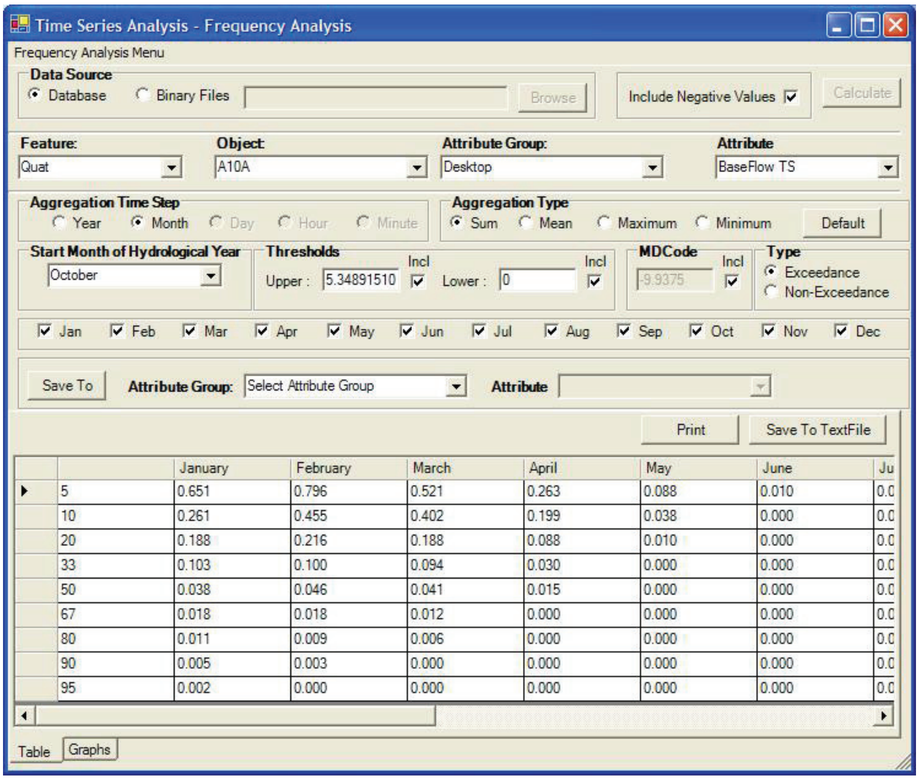


Figure 2.36: Example of the new *Time-series Analysis – Frequency Analysis* form

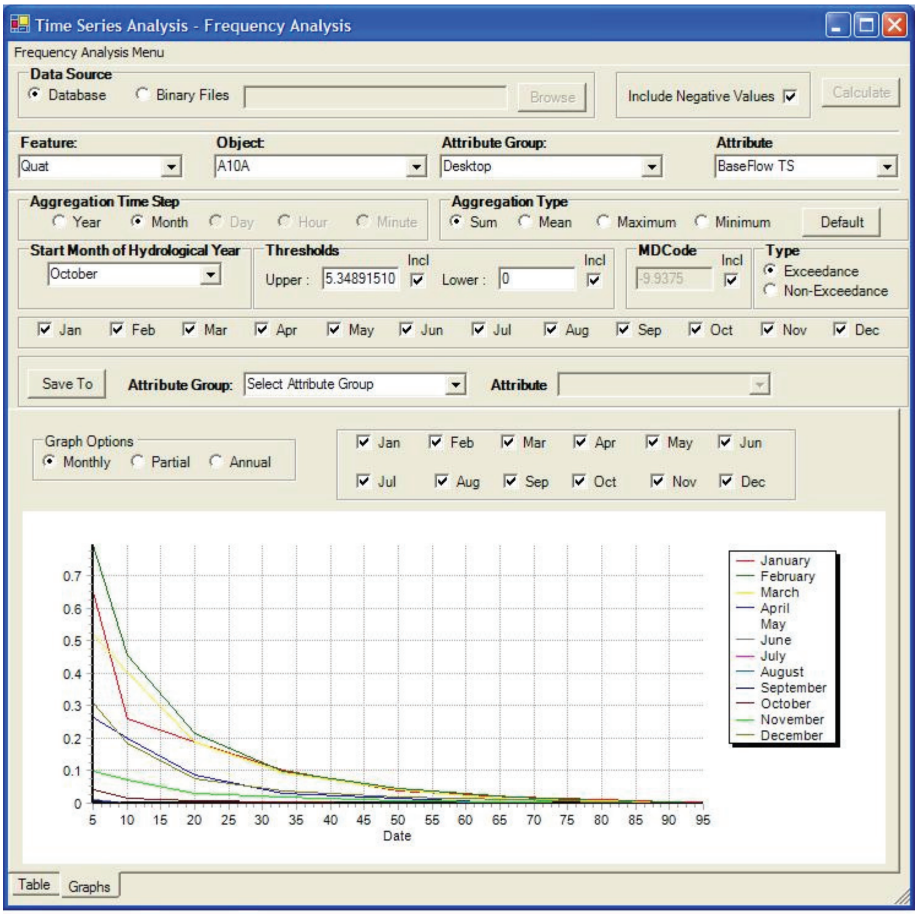


Figure 2.37: Example of the *Graphs* tab on the new *Time-series Analysis – Frequency Analysis* form

Procedure - T/S Summary

The time-series summary process, illustrated in Figure 2.38, is started by selecting any flow data time-series attribute within SPATSIM-HDSF. As illustrated in Figure 2.38, there are four possible output options, based on the input flow data time-series and an optional set of baseflow separation parameters. These are flow duration curve tables, monthly and annual statistics (means, standard deviations, coefficients of variation, % time of zero flow, etc.) and the generation of baseflow time-series using the input separation parameters. The procedure operates on all the time-series data that are available (i.e. it uses all the spatial components for the current feature that have data connected to them and ignores those spatial components for which there are no input flow data).

This is therefore a very quick way of getting a summary of time-series data for a large number of points or polygons associated with a spatial feature. The annual and monthly statistics can then be written out to a text file by selecting the appropriate output attribute in SPATSIM-HDSF and exporting the data.

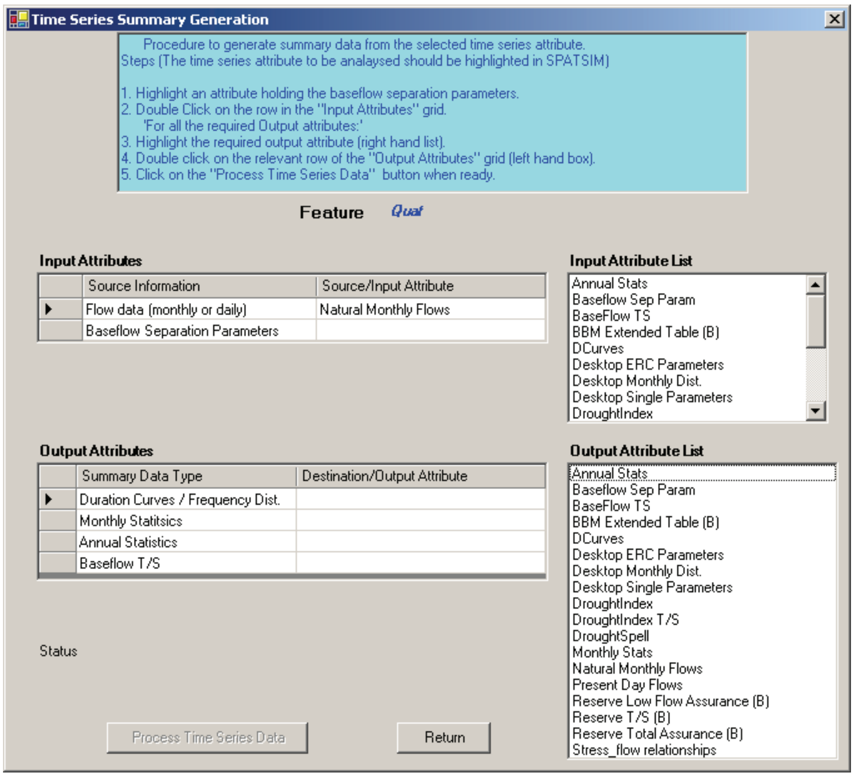


Figure 2.38: The *T/S Summary* main page

Procedure - Drought Indices

There are two components to this procedure. The first is to generate time-series of drought indices from input monthly rainfall data, while the second is to extract some of the indices from specific years and use these to render a map and investigate regional patterns of drought severity.

The drought index generation process, illustrated in Figure 2.39, is more fully described in the SPATSIM Help (available on the IWR web page under the link to hydrological modelling software and SPATSIM [http://www.rhodes.ac.za/institutes/iwr/software/spatsim.html]). These procedures were developed for IWMI in Sri Lanka and have been quite widely used in parts of SW Asia. The approach used in SPATSIM-HDSF is identical to that used in SPATSIM.

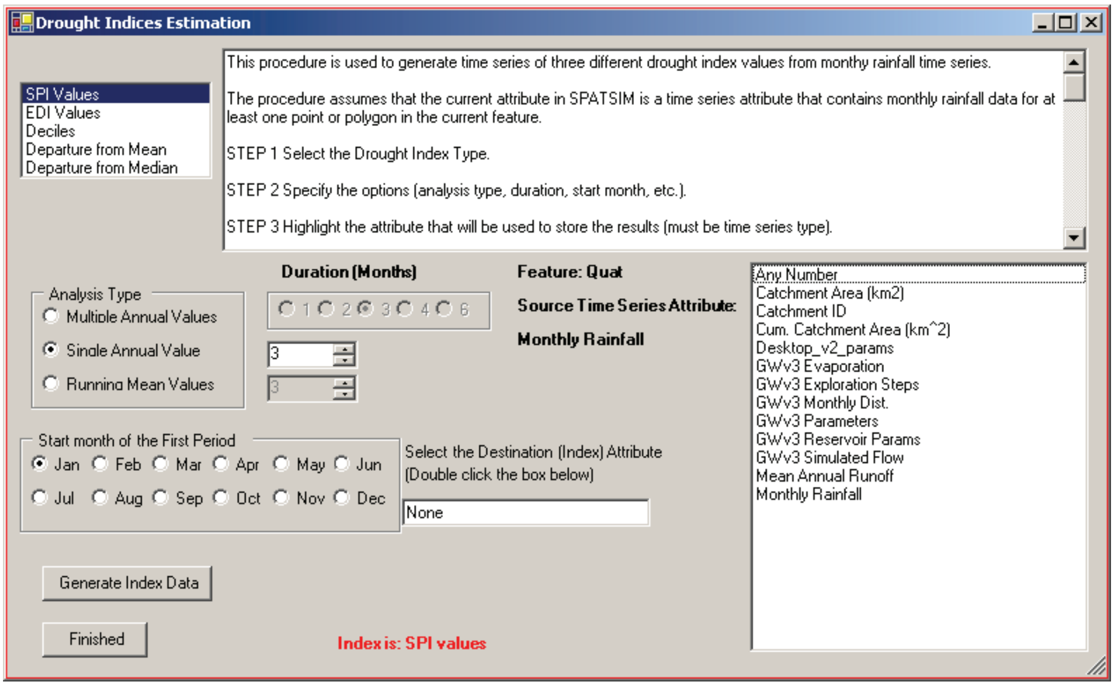


Figure 2.39: The *Drought Indices* main page

2.4.4 External GIS Tools

GISs have become invaluable in hydrological modelling and GIS tools are an important component of the HDSF. The internal GIS tools in SPATSIM-HDSF are suitable for visualisation of spatial data and simple editing of geographical data in a SPATSIM-HDSF project. However, for complex editing or large scale geographical data manipulation the external GIS component of the HDSF enables users to utilise the functionality built into a commercial GIS package to display, edit and manipulate geographical information associated with a SPATSIM-HDSF project. The external GIS tools have been developed as an extension to the ArcGIS desktop software using the Visual Basic .Net (Version 2.0) programming language. The HDSF External GIS extension includes:

- a toolbar with menu and button options
- forms to catch and display error messages which may arise during the execution of the code
- database connection classes to access and query information in a SPATSIM-HDSF database
- selection and loading of existing SPATSIM-HDSF projects, where the geographical data pertaining to the selected SPATSIM-HDSF project will be loaded into the ArcGIS environment and appropriate symbology and labelling will be applied to the GIS coverages
- analysis of Gridded GIS information based on a specified SPATSIM-HDSF coverage so that feature attributes in a SPATSIM-HDSF database can be populated automatically
- area weighting of information in one coverage relative to a second coverage.

The HDSF External GIS tools toolbar for ArcGIS is shown in Figure 2.40. Using this toolbar the user may select a SPATSIM-HDSF database, view spatial data from a SPATSIM-HDSF database and perform Grid and area weighting analyses.

Figure 2.40: An example of the *HDSF* external GIS tools toolbar for ArcGIS

An example of the use of the HDSF External GIS tools to display attribute data for a spatial feature stored in a SPATSIM-HDSF database is shown in Figure 2.41. In Figure 2.41 it is shown how using the HDSF External GIS tools toolbar in ArcGIS, a connection to a SPATSIM-HDSF database has been created, values of mean annual precipitation (MAP) for a feature containing quaternary catchments have been loaded into ArcGIS and displayed. A predefined legend has been used and the quaternary catchments have been automatically labelled using one of the fields in the shape file as specified in the layer (*.lyr*) file associated with the shape file. Once data has been loaded from a SPATSIM-HDSF database all ArcGIS functionality is available including the creation of layouts for printing.

Using the GRID analysis tool, which is part of the HDSF External GIS tools, data can be extracted from GRIDs by selecting a SPATSIM-HDSF spatial feature and attribute and specifying an associated GRID from which to extract the data. The extracted data values are then saved in the SPATSIM-HDSF database for the selected attribute. An example of the use of the GRID analysis setup form is shown in Figure 2.42. In the example shown, all the catchment values for MAP are populated with the average MAP derived from the GRID named 'mean_an_precip'.

Using the coverage area weighting analysis tool, which is part of the HDSF External GIS tools, spatial data can be area weighted by selecting a SPATSIM-HDSF spatial feature and attribute and specifying a coverage containing data to be area weighted and the data field to be area weighted. The area weighted data values are then saved in the SPATSIM-HDSF database for the selected attribute. An example of the use of the coverage area weighting analysis setup form is shown in Figure 2.43. In the example shown, the crop coefficients for January are area weighted per catchment and saved to the specified attribute in the SPATSIM-HDSF database.

A user manual has been written for the external GIS tools and has been included as Appendix B.10 to this report.

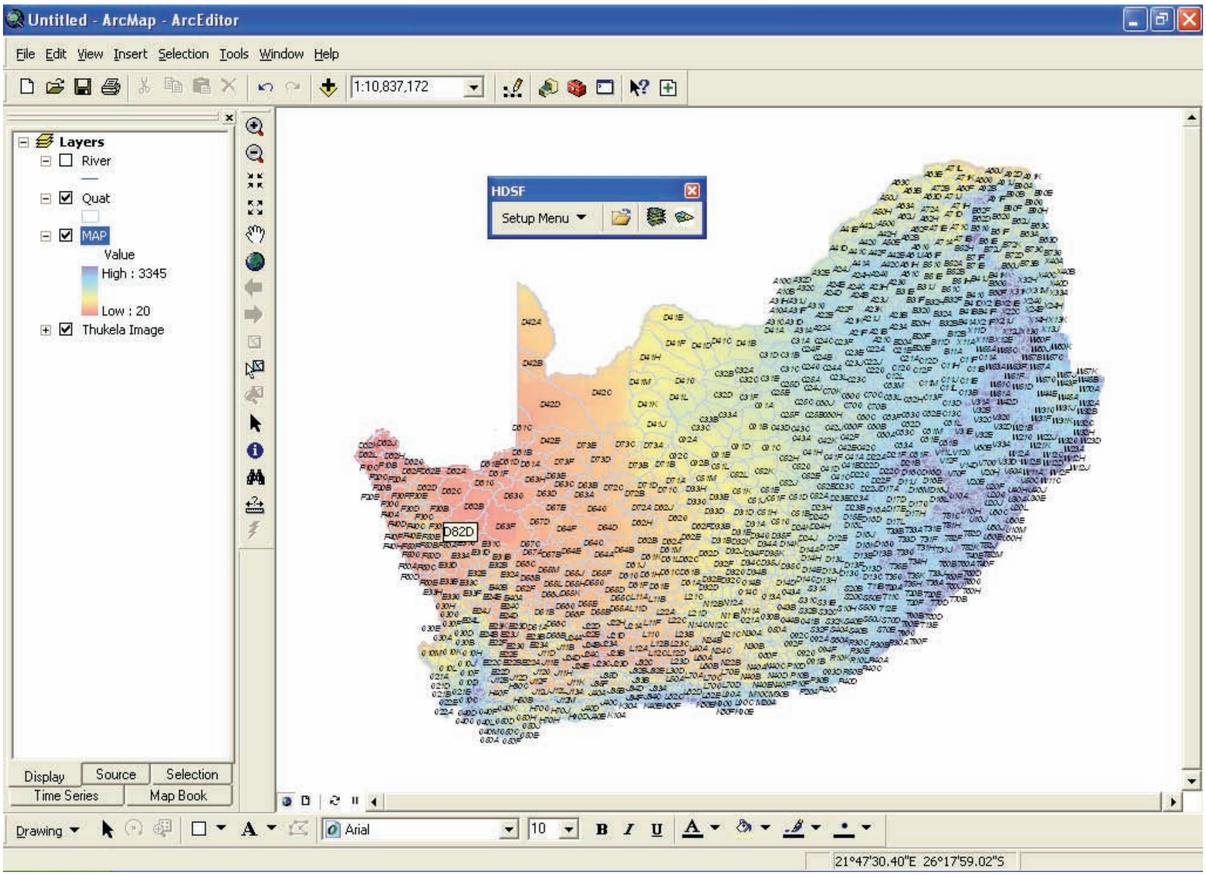


Figure 2.41: Example of attribute data from a SPATSIM-HDSF database displayed in ArcGIS

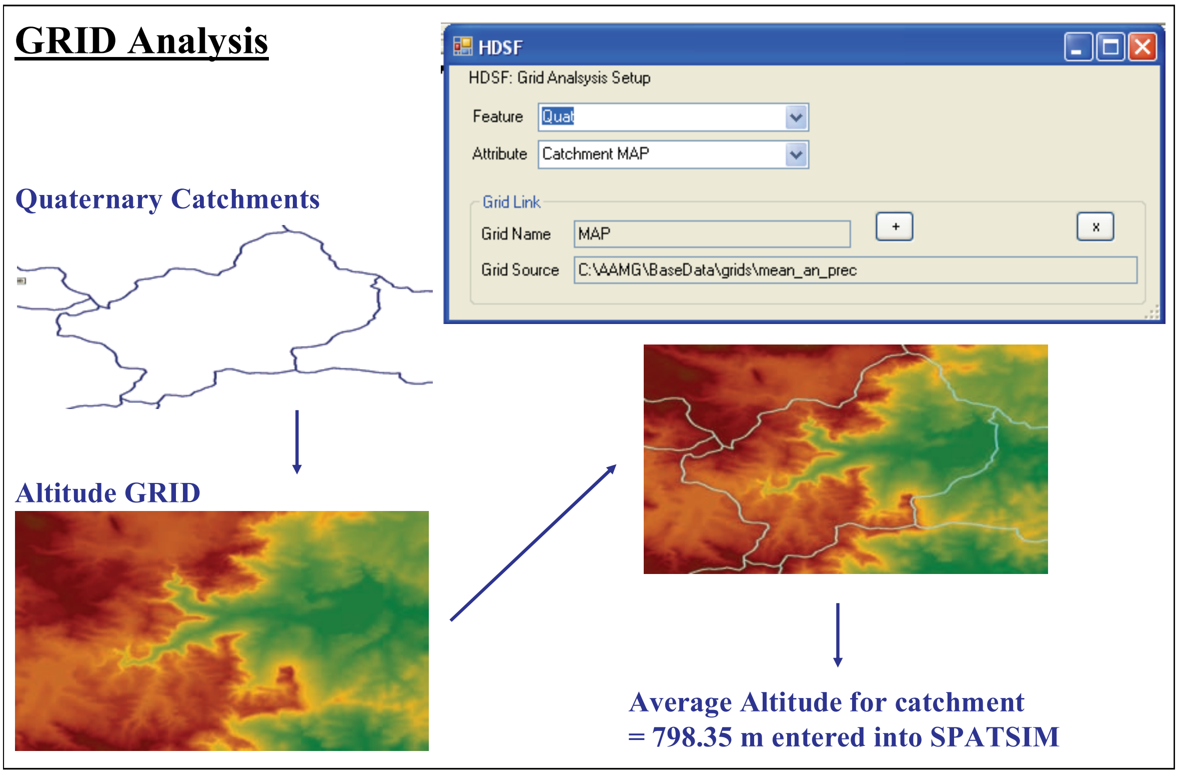


Figure 2.42: An example of the HDSF GRID analysis tool for ArcGIS

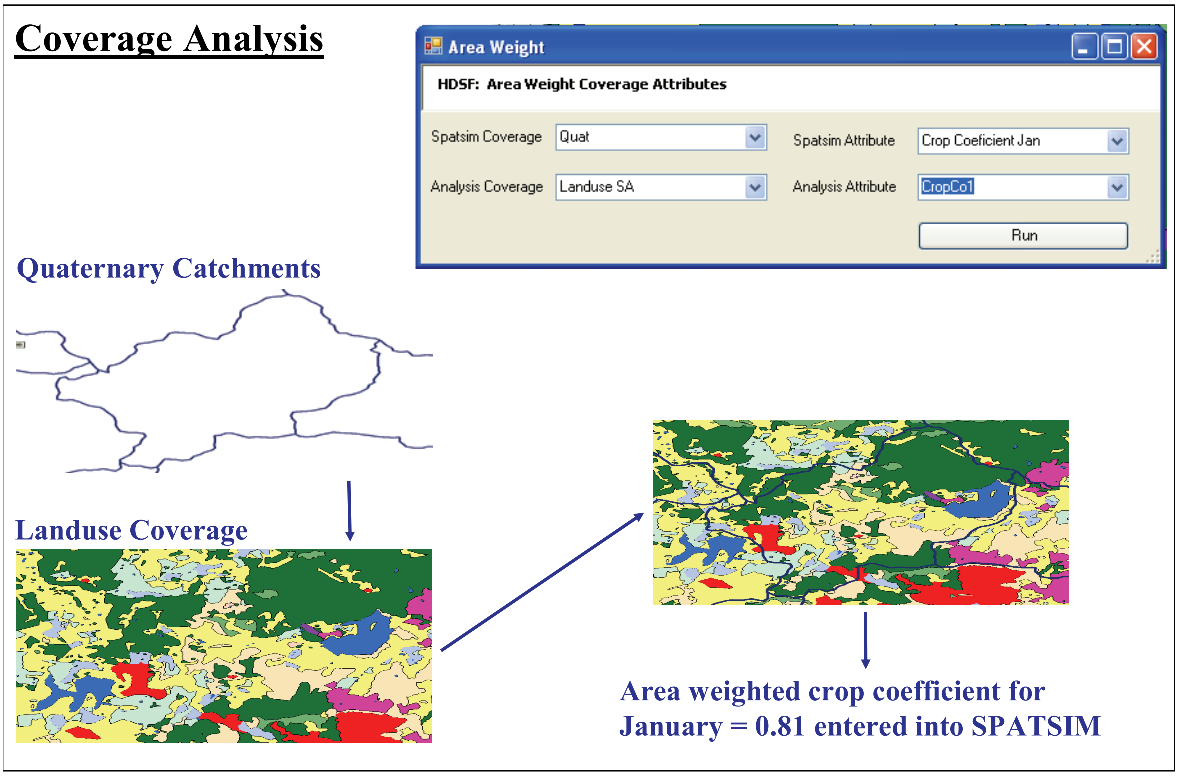


Figure 2.43: An example of the HDSF Coverage analysis tool for ArcGIS

2.5 Investigation of OpenMI

The OpenMI model linking framework is briefly described in Section 2.2.1 and is described in more detail in the Framework Review report (Appendix B.3). At the Framework Selection Workshop (Appendix B.4) it was decided that OpenMI be adopted as a possible means of linking models, and in particular, linking models is parallel. Therefore, an investigation into the use of the OpenMI model linking framework was undertaken by the project with the purpose of developing a simple proof-of-concept example of the use of OpenMI. The details of this investigation are described in more detail in Appendix B.9.

The proof-of-concept example aimed to achieve the following:

- Link two simple models, both written in C#, such that model variables calculated in one model are accessed and used by a second model
- Link two simple models running at two different time-steps, one daily and one hourly
- Link a simple model to a simple data source using an OpenMI wrapper for the data source.

In addition, a recommendation was sought regarding the feasibility of making a Java model OpenMI compliant and being able to link this model to an OpenMI compliant model written in a .Net programming language.

For the proof-of-concept example it was assumed that the simple models already existed and therefore the aim was to demonstrate wrapping of an existing model, as opposed to the development of a new model that implemented the OpenMI Standard interface directly. The two models to be linked were in the form of a simple rainfall-runoff model and a simple flow routing model. The proof-of-concept example was completed in the form of three C# code sets, one containing the simple rainfall-runoff model, one containing the simple flow routing model and the third containing a simple data source. Each of the code sets contained an OpenMI compliant wrapper enabling the relevant model or data source to be used with other OpenMI Linkable Components in a model run.

The OpenMI Guideline document (HarmonIT, 2005b) (Book 4, Section 4.4) was used as a guide when generating the model engine wrapper classes. The OpenMI software development kit (SDK) assemblies provided default implementations of many of the OpenMI wrapper classes as a useful starting point and much of the code in this proof-of-concept example used these classes directly, or was based on slightly modified versions of these wrapper classes.

In most instances wrapping a model to make it OpenMI compliant will require changes to the underlying model engine. The extent of these changes is dependent on the architecture of the model engine. The main requirement on the underlying engine is that it can expose the following functional hooks:

- An initialisation step, preparing for the model run and accepting initialisation parameters
- An exposed hook to allow the execution of a single time step, the engine must be able to execute one time step on request, and then stop and wait for the call to execute the next
- A finalisation step, closing file, persisting results etc
- A disposal step, cleaning up and releasing any system resources.

All of the other operations involved in the linking are taken care of in the wrapper class structure, particularly in the LinkableEngine implementation (HarmonIT, 2005b) (Book 4, Section 4.3.2). This includes publishing the definition of the types of data and data operations required and provided by the model engine, as well as facilitating model input, output, buffering and interpolation of values.

The proof-of-concept was successfully developed and there is no reason why more complex models written in a .Net programming language should not be made OpenMI compliant. The complexity of the process of wrapping a model to make it OpenMI compliant will depend on the architecture of the model.

A model written in the Java programming language could be wrapped in a similar manner as the OpenMI Standard is the same, though there is not a corresponding OpenMI SDK for Java so the implementation classes would have to be written from scratch. An OpenMI compliant Java model can be linked via OpenMI to another OpenMI compliant Java model. However, to provide an OpenMI interface for a Java model that enables it to be linked to a Win32 based OpenMI compliant model would require an extra level of abstraction. Normally when a model engine which is not crafted in managed code is wrapped, there is a Windows-API or COM interop layer that is introduced in the wrapping code to marshal the calls backwards and forwards between the model engine and the .Net wrapping component. Unfortunately, .Net code cannot interoperate directly with Java libraries in this way, and the only solution is to provide a messaging layer in the wrapper to enable communication to cross the boundary. There are existing technologies for implementing .Net-Java interop, such as XML web services, runtime bridges, message oriented middleware, integration brokers or a shared database. The potential problem of .Net-Java interop requires further investigation.

2.6 Summary

The framework development component of the HDSF project started with the identification of user requirements. Six modelling frameworks were reviewed in detail to determine their suitability for use in the HDSF. These modelling frameworks reviewed were evaluated against a list of requirements identified for the HDSF. The SPATSIM-HDSF modelling framework was selected to form the core of the HDSF and has been further developed to meet the requirements of the HDSF. This further development resulted in changes to the SPATSIM-HDSF database data model, changes to the main SPATSIM-HDSF user interface software including, translation of the code to Delphi for .Net, new attribute data editing forms, the data tabbed page showing a table of attribute values, enhancements to the internal GIS tools and development of a consolidated set of time-series analysis tools. In addition an important development was the creation of the external GIS tools in ArcGIS which also provide a link between ArcGIS and data stored in a SPATSIM-HDSF database. The OpenMI model linking framework was investigated as a means of linking models in parallel and as a means of linking models to a data source such as SPATSIM-HDSF database. The SPATSIM-HDSF modelling framework software developed in this project is included in Appendix D on the DVD accompanying this report.

3. MODEL DEVELOPMENT

The primary objective of the HDSF project was the implementation of a modelling framework within which integrated hydrological and related simulation modelling can take place. A secondary objective of the project was to provide for the modelling requirements of Catchment Management Agencies (CMAs) for the assessment and management of available water resources and their dynamics within a catchment. The development of the modelling framework became a larger component of the project than was initially anticipated, however, this was important for the provision of a foundation on which models and other modelling tools could be built. The SPATSIM-HDSF modelling framework is designed such that other models may be added after this project has been completed if required by CMAs. The purpose of this chapter is to describe the model development work carried out as part of the HDSF project. This work included a survey of modelling requirements for CMAs, conversion of existing SPATSIM compliant models, consolidation and further development of the *ACRU* model and modification of the *ACRU* model to run within SPATSIM-HDSF. Further detail of the model development work can be found in Appendices B.11, B.12, B.13 and B.14.

3.1 User Requirements

At the inception of the HDSF project an initial list of perceived modelling requirements by CMAs was compiled based on experience from members of the project team and from a user requirement survey reported in Appendix B.2. At the Inception Workshop for the project, reported in Appendix B.1, this list of modelling requirements was discussed and delegates were asked to rank each item in the list to assist in prioritising the extensive list of modelling requirements identified. Each item was assigned a rank from 1 (least important) to 10 (most important). The results of this exercise are shown in Table 3.1. The user requirements shown in Table 3.1 should be seen as a list of perceived modelling requirements by CMAs and not as a checklist of outcomes for the HDSF project in which the modelling component was not the primary objective. Requests from stakeholders that arose during the project were given priority where possible.

Table 3.1; Results of the ranking exercise on modelling requirements

Category	Requirement	Mean
General	Modelling at a range of spatial scales from Quaternary scale and finer	8.4
	Modelling at a daily temporal scale	7.9
	Represent catchment dynamics (non-stationarity)	7.5
	Acceptable modelling turn-around time	7.6
	Modelling algorithms must be transparent and appropriate	9.1
	Modelling outputs must be meaningful to stakeholders	8.5
Allocation	Evaluation of different water allocation strategies such as partitioning of yield and Fractional Allocation and Capacity Sharing (FACS)	7.9
	Need to be able to calculate river system/reservoir yield characteristics	8.2
	Include groundwater in calculation of assurance of supply	7.3
	Influence of small farm dams and irrigation abstractions from these on allocable water	7.2
	Water supply to: Reserve, irrigation, industry, bulk water suppliers, hydropower and international obligations	8.3
	Allow for trading of water allocations	7.3
	Assurance of supply for each licensed user (planning)	7.1

Category	Requirement	Mean
Reserve	Assessment of Resource Class options	7.1
	Tools for determination of the Reserve	7.8
	An audit methodology to ensure that Reserve is met	6.9
Management	Transparent, efficient and meaningful methodology is required to reconcile water supply and water demand (planning)	8.8
	Application of operating rules and curtailments for large reservoirs and upstream users	8.5
	Operational management of the Reserve	8.1
	Allow for inter-catchment transfers	7.7
	Realistically simulate demand patterns of water users	7.6
	Operational tracking and accounting of water	6.8
	Application of forecasting information	6.4
	Compliance monitoring methods (in-stream and abstraction auditing)	7.1
	Near real time use of monitoring information	6.6
Land Use	Evaluate effects of land use on water resources particularly Streamflow Reduction Activities (SFRAs) and alien vegetation.	7.1
	Water requirement modelling for a range of vegetation types	6.9
Irrigation	Estimation of irrigation demands, supply, scheduling, Best Management Practices (BMPs), Water Use Efficiency (WUE)	7.5
Surface Water	Determination of quantity of surface water available for allocation (consider spatial and temporal variability)	9.2
	Estimation of naturalised/baseline streamflows	7.5
	Need to model return flows from irrigation, industry and other water users	7.1
	Account for farm dams and if necessary model these in a lumped manner	6.6
	Allow off-channel storage in dams and other structures	7.2
	Account for wetlands and riparian zones	6.7
	Account for transmission losses	6.0
	Need to model hillslope processes	5.4
	Need to do flow routing down a river system	6.1
Groundwater	Determination of quantity of groundwater available for allocation (consider spatial and temporal variability)	7.1
	Interaction of groundwater with surface water	7.2
Water Quality	Model water quality considering point source and non-point source pollution.	7.2
	Model cascading water quality effects down river reservoir systems.	6.5
	Need to model soil erosion and sediment transport.	6.3
Other	Economic	5.6
	Socio-political	4.7

3.2 SPATSIM-HDSF Models

During this phase of the conversion from the previous version of SPATSIM to the new HDSF version of SPATSIM it was considered important to convert the models that are most commonly used within SPATSIM (either by the IWR or by other users), as well as to cover the main issues that are likely to arise in the conversion of all other models.

The Pitman Model

This model is used frequently by the IWR for both research and practical purposes. In terms of the generation of simulated natural flows it is equivalent to the new version of the standard Pitman model used in WR2005. However, in the simulation of development effects (reservoirs, abstractions, etc.) it uses a different approach.

Pitman Model Parameter Exploration

This is the same version of the basic Pitman model but includes the input of range values and steps for selected parameters. The program then runs the model multiple times and generates objective functions (statistical comparisons between observed and simulated flows) for all the possible parameter combinations (using the specified step sizes within the specified range of parameter values). This program is used by the IWR to investigate parameter sensitivities and results uncertainty associated with the Pitman model.

The Desktop Reserve Model

This model is the widely used Desktop model for the ecological Reserve and is used by a number of DWAF consultants as well as the IWR.

The Stressor-Response Model

This model is also widely used by DWAF consultants within the Reserve determination process (usually at intermediate and comprehensive levels).

General

The four models that have been converted represent most of the different approaches to data access that are used in all the other models currently associated with SPATSIM. The IWR therefore has example code that can be used as a basis for the future conversion of other models. These can therefore be converted quite quickly as they are required for use with the HDSF version of SPATSIM.

3.3 *ACRU* as a Model in SPATSIM-HDSF

Models will be required by the CMAs to assist in the assessment and management of available water resources, water quality and groundwater and their dynamics within a catchment, and it is likely that a physical conceptual model operating at a daily timestep will be needed to adequately represent certain hydrological and operational processes for the fulfilment of some of these modelling requirements. The *ACRU* model was selected for inclusion as a model to be run from SPATSIM-HDSF as part of the HDSF project for the following reasons:

- it is a physical conceptual model that operates at a daily time step
- it has been locally developed and applied in South Africa
- expertise exists within the School of BEEH to further develop and run the *ACRU* model
- it already contains much of the core modelling functionality required
- it has been developed with funding from the WRC
- is on the list of models recommended by DWAF for use in South Africa.

The *ACRU* 300 series version of the *ACRU* model is still in wide use, however, unless stated otherwise references to the *ACRU* model in this report should be taken as referring to the *ACRU*2000 version of the model.

3.3.1 *ACRU* Input Data File

The input data requirements of the *ACRU* model are considerably larger than the existing models within SPATSIM and thus careful consideration was required regarding how *ACRU* should be incorporated and run within SPATSIM-HDSF. Some of the considerations were as follows:

- The SPATSIM-HDSF modelling framework includes a tool enabling users of one of the models within SPATSIM-HDSF to associate an attribute in a SPATSIM-HDSF database with each variable required by the model. However, it would be onerous for *ACRU* users to set up the input data for an *ACRU* simulation in this variable-by-variable manner.
- The SPATSIM-HDSF database is a structured and efficient data repository for input data required by the *ACRU* model. However, existing methods of storing model configuration information in SPATSIM-HDSF were not suited to storing the model configuration information required by *ACRU*.
- The *ACRU*2000 version of the *ACRU* model was designed with an object-oriented structure as described in Kiker (2001) and Kiker et al. (2006). Although the *ACRU*2000 version of the *ACRU* model has an object-oriented structure, this could not be used to its full potential due the non-object-oriented text input data files being used. Therefore *ACRU* required an object-oriented input format for its object-oriented model structure to be used to its full potential.
- The *ACRU*2000 version of the *ACRU* model requires some means of mapping variable names in the input data file to Data classes in the model. The existing method of doing this was difficult to maintain and needed to be replaced.
- The *ACRU*2000 version of the *ACRU* model requires some means of the storing values for state variables at the end of a simulation period so that the model may be restarted from this state when being used for operational modelling
- There may be instances where it would be advantageous to be able to run the platform independent *ACRU* model without requiring a platform dependent RDBMS.

For these reasons it was decided that a new input data file format was required for *ACRU* to store model configuration information, provide an object-oriented input data structure, store links between model variables and data sources especially SPATSIM-HDSF databases, and also be able to store input data. It was decided that the most appropriate storage format would be XML [http://www.w3.org/XML/] which is text based and therefore human readable but also highly structured and platform independent.

The design of a suitable XML file has been through many iterations, as the finer design requirements became clearer and as experience was gained in the design of XML files and the associated programming tools available to read from and write to XML files. An overview of the XML schema design for the *ACRU* data input file is shown in Figure 3.1, Figure 3.2 and Figure 3.3. This design has been implemented but may be refined further. While this model input data XML file was designed primarily for use with the *ACRU* model there is no reason why this same design, or a very similar design, should not be applied to other models in SPATSIM-HDSF.

The root element *Model* shown in shown in Figure 3.1 contains several elements, most of which act as containers for more specific elements. The concept of Component, Process and Data classes used in the *ACRU*2000 version of the *ACRU* model as explained in Kiker (2001) and Kiker et al. (2006) has been carried through to the design of the model input data file. Component classes represent the physical components (e.g. catchment, river, vegetation) of the hydrological system being modelled, Data classes represent the attributes (e.g. area, flow rate, crop coefficient) of these components and Process classes represent the hydrological processes (e.g. runoff, transpiration) that occur on and in these components. The storage of model input data is mostly through *Component* elements shown in Figure 3.2 and *Data* elements shown in Figure 3.3 where the *Component* elements help to group the *Data* elements into logical sets. In addition the *Relationship* and *RelationshipType* elements are used to store information about relationships between *Component* elements, for example, catchment B is downstream of catchment A. The *DataRef* element is used to store information used to locate and connect to data sources referenced in the *Data* element, for example a SPATSIM-HDSF database or a text file containing time-series data. The *ModelData* element stores a list of *Data* elements containing model settings.

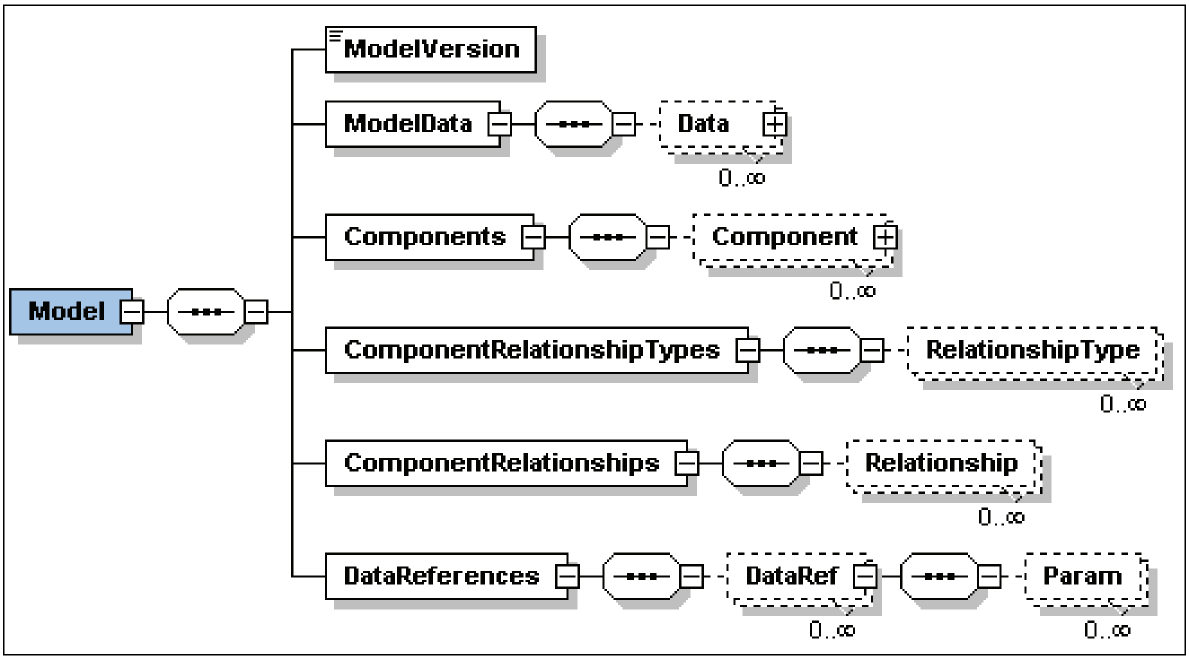


Figure 3.1: Main elements in the *ACRU* input file XML schema design

Each instance of a *Component* element is required to have a unique ID and consists of a list of *Data* elements, a list of subcomponent *Component* elements and a list of *Process* elements if the user of the model wants to state the *ACRU* Process classes explicitly. While it makes sense conceptually, and is possible in XML, to have *Component* elements nested as subcomponents of a parent *Component* element, this can add complexity to the operations of reading and writing the XML file. This aspect of the design may be changed so that *Component* elements are no longer nested but instead include an attribute storing the ID of the parent *Component* element. The *Component* element has a *Type* attribute which is used as a reference to information about the component type (similar to an *ACRU* Component class) that is stored in the *ACRU* ModelConfiguration file described in Section 3.3.2 of this report.

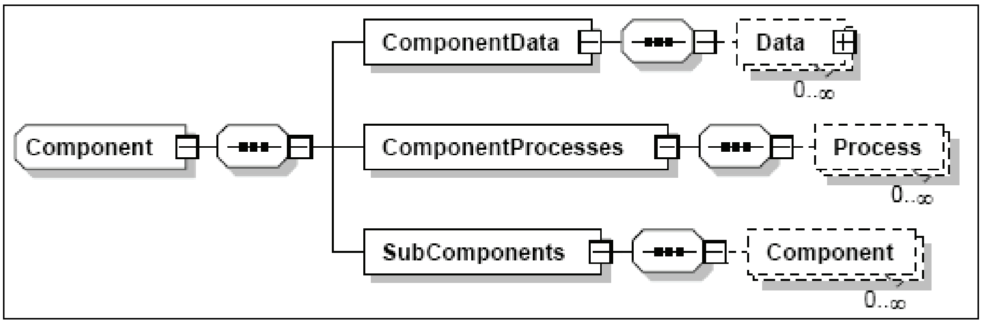


Figure 3.2: Component element in the *ACRU* input file XML schema design

The purpose of the *Data* element is to store data values and other useful metadata about these data values. The *Data* element has an *ID* attribute that is used to identify the model variable it represents. Each *Data* element may store either a single value, a record of values, for example an array, or a reference to a value stored in an outside data source such as a SPATSIM-HDSF database. In addition a *Data* element may store a time-series of values, a time-series of records or a reference to a time-series of values stored in an outside data source such as a SPATSIM-HDSF database. Generally it would not be advisable to store time-series data in an XML file as it is not a very efficient means of storing this type of data, but this is intended for the storage of a limited number of data points for state variables. The value stored in the *ID* attribute is used as a reference to information about the data variable (similar to an *ACRU* Data class) that is stored in the *ACRU* ModelConfiguration file described in Section 3.3.2 of this report. Each instance of a *Data* element may contain zero or more instances of the output reference *OutRef* element that specifies whether the variable represented by the instance of *Data* element should be included in the model output and which model output files it should be saved to. Currently the only output format is dBase (.dbf) files.

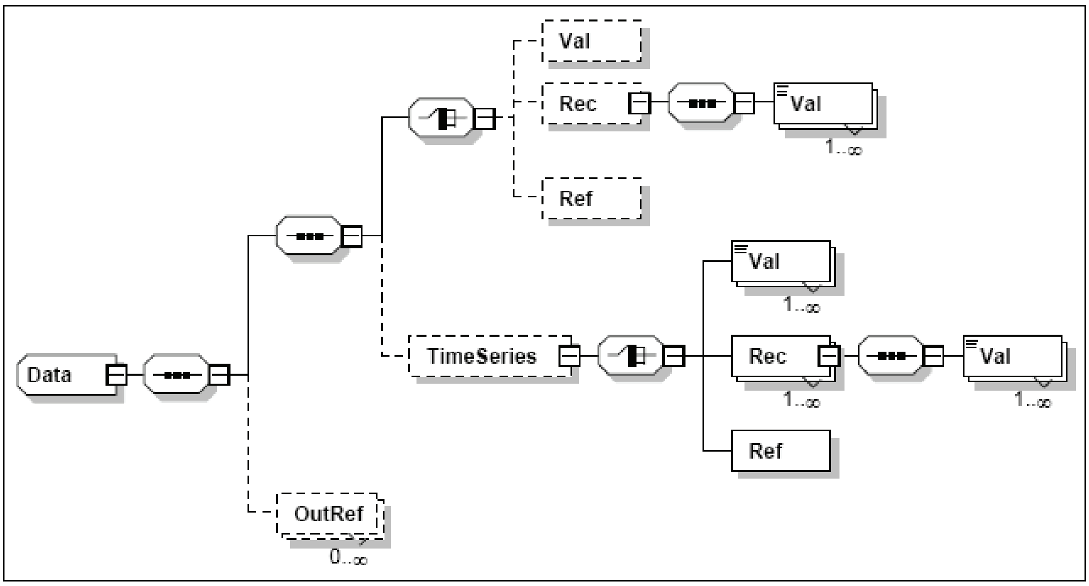


Figure 3.3: Data element in the *ACRU* input file XML schema design

A set of classes has been created in both the Java and C# programming languages to assist code developers with reading and writing these *ACRU* input data XML files.

3.3.2 *ACRU* Model Configuration File

The initial designs of the *ACRU* input data files described in Section 3.3.1 of this report included a lot of metadata type information that was not data but necessary for setting up the *ACRU* model. On further consideration it was realised that this information could be stored once in a separate file instead of being repeated in every instance of an *ACRU* input data file. This resulted in the development of an *ACRU* model input configuration file. An overview of the XML schema design for the *ACRU* model input configuration file is shown in Figure 3.4. Instances of the *ComponentType* element are conceptually similar to *ACRU* Component classes and contain an attribute specifying the *ACRU* Component class represented. Instances of the data definition *DataDef* element are conceptually similar to *ACRU Data* classes and contain an attribute specifying the *ACRU* Data class represented. The *DataDef* element has an attribute *Type* that is used to specify whether the *DataDef* element represents an input, output or state type variable. *DataDef* element also has an attribute *ValueType* that is used to specify the value type, for example, string, integer or double and an attribute that stores a reference to a unit of measure for the model variable represented.

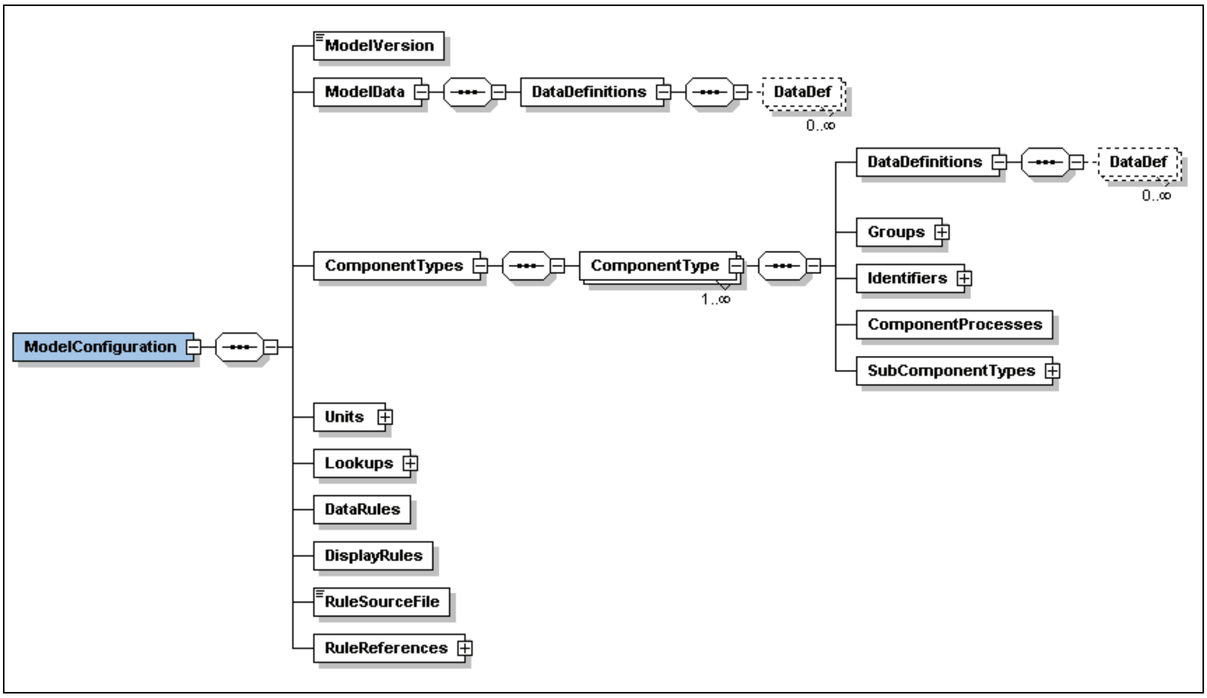


Figure 3.4: Main elements in the *ACRU* model configuration file XML schema design

During the course of the design of the model configuration XML file for *ACRU* it was found that it was very similar to the XML based configuration schema developed for the *ACRU* Configuration Editor described in Section 3.4 of the report. These two configuration files were therefore combined to produce the *ACRU* model configuration file shown in Figure 3.4.

While this model configuration XML file was designed primarily for use with the *ACRU* model, there is no reason why this same design, or a very similar design, should not be applied to other hydrological models.

3.3.3 Modifications to the *ACRU* Model

Several modifications to the *ACRU* model were required to implement the new model input data file and model configuration file described in Sections 3.3.1 and 3.3.2 of this report and to read data from a SPATSIM-HDSF database. These modifications were for the most part surprising easy to implement and confirmed the advantage of having an object-oriented input data file format for the object-oriented model. Further details of the modifications to the *ACRU* model can be found in Appendix B.14.

3.3.4 *ACRU* Input Data File Converter

It was anticipated that there will be users of the *ACRU* model that have *ACRU* model input data files stored in the text file format used by the *ACRU* 300 series version of the model and would like to store and run these legacy datasets using the SPATSIM-HDSF modelling framework. For this purpose the conversion program shown in Figure 3.5 was created. At present the conversion program includes two main conversion options: (1) to convert an *ACRU* 300 series version text input data file to an *ACRU*2000 version XML input data file with the data stored internally, and (2) to convert an *ACRU* 300 series version text input data file to an *ACRU*2000 version XML input data file with the data stored in a SPATSIM-HDSF database. The user is required to specify the *ACRU* 300 series version text input data file to be converted, the corresponding supplementary input data file if it exists and the name of the new XML input data file to which the converted model input data will be written. In addition, if the data is to be saved to a SPATSIM-HDSF database then the user is required to specify the initialisation (.ini) file for the SPATSIM-HDSF project containing the SPATSIM-HDSF database. The user has an option to select whether the time-series input data is to be imported or for the existing text based time-series input files to be referenced as data sources. The user may also select a start subcatchment and an end subcatchment if only a subset of the original input data is to be converted. Once the conversion has been completed, the user may run the *ACRU* model from the converted XML based input data file and associated SPATSIM-HDSF database.

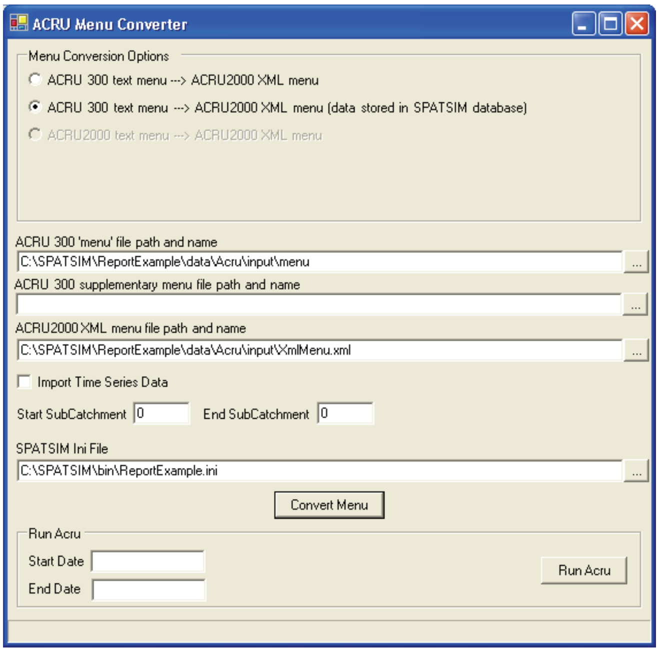


Figure 3.5: *ACRU* input data file converter

3.3.5 Running *ACRU* from SPATSIM-HDSF

The *ACRU* model may be run from the SPATSIM-HDSF modelling framework by selecting the *Application / ACRU Model* menu item to display the *ACRU Interface* form shown in Figure 3.6. The name of the initialisation file for the current SPATSIM-HDSF project is passed to the *ACRU Interface* form so that the default folder paths and the location of the SPATSIM-HDSF database can be determined. From the *ACRU Interface* form the user may run the *ACRU* input data file conversion program described in Section 3.3.4 of this report. The user could then select the output variable setup tool to specify which variables are to be included in the model output files. If an *ACRU* input dataset exists in the current SPATSIM-HDSF project then the user may run *ACRU* by selecting the *Run ACRU* button which will result in the *Run ACRU* form shown in Figure 3.7 being displayed. The path for the folder containing the input data file for *ACRU* is set automatically. A default folder to which the model output data files are to be saved is also set automatically, but may be changed by the user. The user may select a simulation start date and end date if the model is to be run for a shorter time period than the simulation time period set in the input data file. Selecting the *Run ACRU* button will result in the execution of the *ACRU* model and the progress of the simulation may be monitored in the Windows Command Prompt window that is displayed during execution of the model.

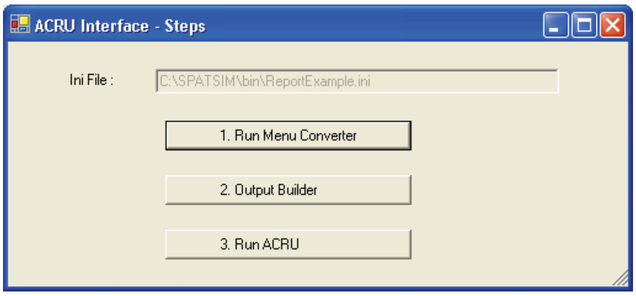


Figure 3.6: Example of the *ACRU Interface* form

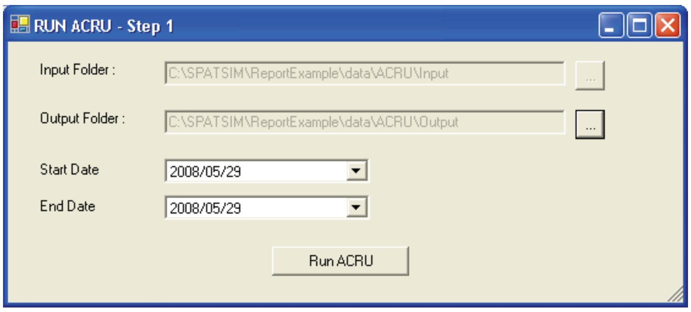


Figure 3.7: Example of the *Run ACRU* form

3.3.6 Conclusion

It has been demonstrated that XML is a suitable means of providing object-oriented input to an object-oriented model such as *ACRU*, and this is seen as an important step forward for the *ACRU* model. It has further been demonstrated that a complex model such as *ACRU* with many input variables can be run from the SPATSIM-HDSF modelling framework using data stored in a SPATSIM-HDSF database. The work reported in this section of the report has proved that the concepts work, however, further work is required to finalise the design of the XML based *ACRU* input data files and model configuration files. In addition further work is required to establish more efficient means of reading and writing XML files and the SPATSIM-HDSF database as these operations in their current form are too slow, especially for large datasets. The concepts tested here could be applied to other hydrological models.

3.4 *ACRU* Model Configuration Editor

A number of tools have been set up within SPATSIM-HDSF to edit data contained in a SPATSIM-HDSF database. Traditionally the models that have been run from SPATSIM have required relatively small sets of input variables and the data editing tools within SPATSIM were sufficient to edit information at this scale. In this project the SPATSIM database data model has been extended and the *ACRU* agrohydrological model has been added to the list of models that can be run from SPATSIM-HDSF. The input data for *ACRU* and other physically conceptual hydrological models can extend to many variables, parameters and time-series and, although the SPATSIM-HDSF tools can be used to edit this data, the process would be slow and users are likely to become frustrated with this form of editing. For this reason it was decided that, as part of this project, a program would be developed to edit these large sets of input variables. This program was termed the Configuration Editor. At the design stage of the Configuration Editor the following objectives were set for the program. It must be designed:

- so that new variables can be added, for editing, without having to change the code of the program
- so that data can be displayed in a user friendly manner and in a hierarchical structure to which they would belong (i.e. all soils data in one group and all vegetation data in another group etc.)
- to enable the editing of multiple lines of data in one instance (e.g. editing all catchments soils properties at the same time)
- to warn the user that entered values are outside of the prescribed range for that variable
- to read and write variable information and data into a file format which would optimise the speed of the system.

3.4.1 Structure of the Configuration Editor

The Configuration Editor was designed as a treeview control to display groups of variables and the data pertaining to those variables (Figure 3.8). The program can be run as a standalone program or can be executed as a dynamic link library (DLL) from a program such as SPATSIM-HDSF. The Configuration Editor requires a configuration file to stipulate the structure of Entities, Groups and Variables.

Entities
Entities in Figure 3.8 would be the hydrological entities such as Catchments, Rivers, Catchment Nodes etc.

Groups
In Figure 3.8 the variables QNUM, HEAD, CLAREA, ELEV, ALAT, ALONG, IHEMI and IQUAD belong to the group “Catchment Attributes” which falls under the entity “Catchments”. Groups can contain other groups to further discern specific processes. In the

example of Figure 3.8 the group “Rainfall” contains the groups “Rainfall Attributes” and “Rainfall Correction”.

Variables

The following attributes are applied to each variable entered in the XML based configuration schema:

- the variable ID
- an alias for displaying the variable
- a description of the variable
- some help text to assist the user when editing data
- the data type of the variable
- whether the variable should be available for editing.

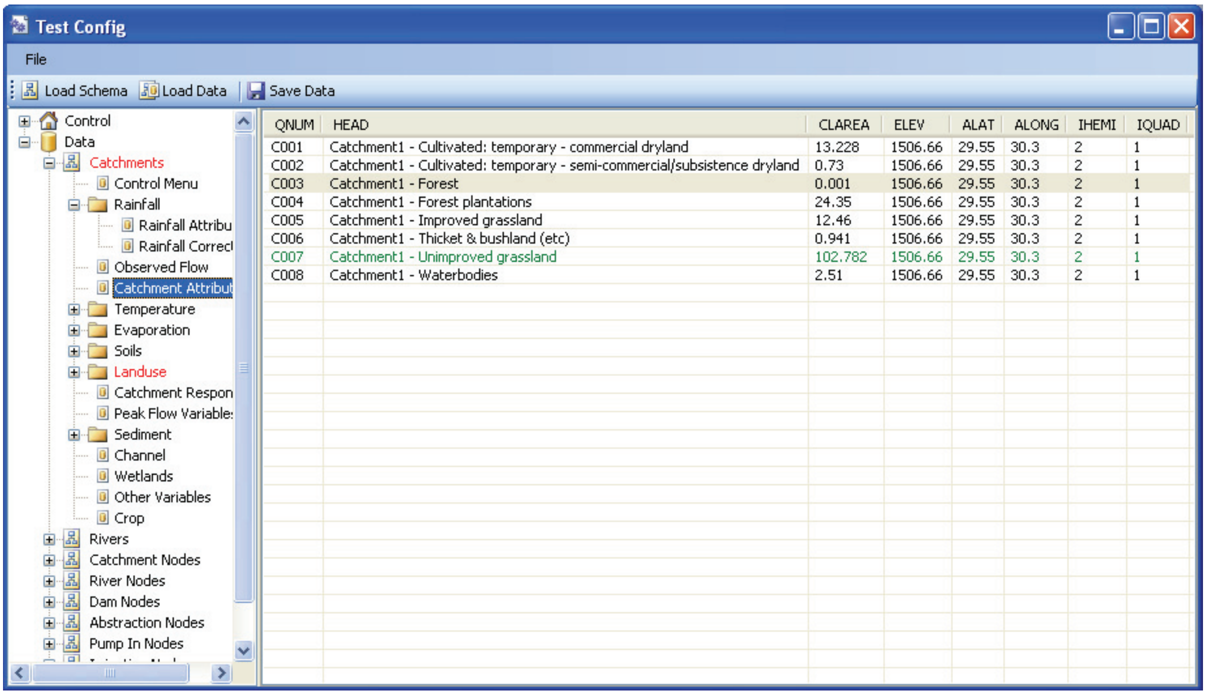


Figure 3.8: The Configuration Editor with *ACRU* variables and loaded data

3.4.2 Configuration Schema

The structure of the Entities, Groups and Variables is stored in an XML based configuration schema (example in Appendix A.3). The Configuration Editor loads the configuration schema and populates the tree view control of the Configuration Editor based on the group and variable data set up in the file.

The Configuration Editor is thus a program which sets up the Entities, Groups and Variables at runtime and based on information entered to the configuration schema. The benefit of compiling a data capture tool such as this is that by adding a variable to the configuration schema then that variable automatically becomes available to the program without having to change the code of the program.

3.4.3 Data Rules and Display Options

On loading a configuration schema the Configuration Editor is also directed to a set of data rules and display options which are contained in a code file written in the C# programming language (example Appendix A.4). These data rules and display options are executed at runtime and allow the Configuration Editor to provide warnings and errors for values entered outside of a stipulated range of information. In the example in Appendix A.4 if the value of variable ABRESP is either greater than 0.9 or less than 0.1 then a warning will be issued to the user and if the entered value is greater than 1 or less than 0 then an error will be displayed to the user. In Figure 3.8 the variable QNUM C007 is coloured green since one of the values entered for a variable is outside of the preferred range and with the entity Catchments being coloured red it shows that one of the variables which falls within one of its groups (in this case Landuse) is outside of the range of values applicable to the model.

A set of display rules can also be defined in the C# code file. The display rules are linked to variables and groups and provide the Configuration Editor with information pertaining to whether variables or groups should be displayed. In the example shown in Appendix A.4 the group CORPPT will only be displayed if the variable PPTCOR is not equal to zero. In this way the Configuration Editor will only display variables and groups which are to be displayed based on previously entered information.

3.4.4 Structure of Data

After importing the configuration schema to populate the treeview control of the Configuration Editor, the user has the option of loading data which pertains to this configuration schema and which is stored in a predefined storage format. An example of one format of stored data is provided in Appendix A.5. In this example the data is stored in CSV format, although the Configuration Editor could store or read data in other formats including XML and databases.

3.4.5 Editing Data using the Configuration Editor

The Configuration Editor has built in forms which enable the user to edit the data displayed. In Figure 3.9 the form for editing data shows that the value entered for CLAREA is outside the preferred range of values for this model.

Multiple entries can also be edited by selecting a number of rows of data and then opening the edit form. Any values entered into the edit form shown in Figure 3.10 will be applied to all the selected rows of the Configuration Editor.

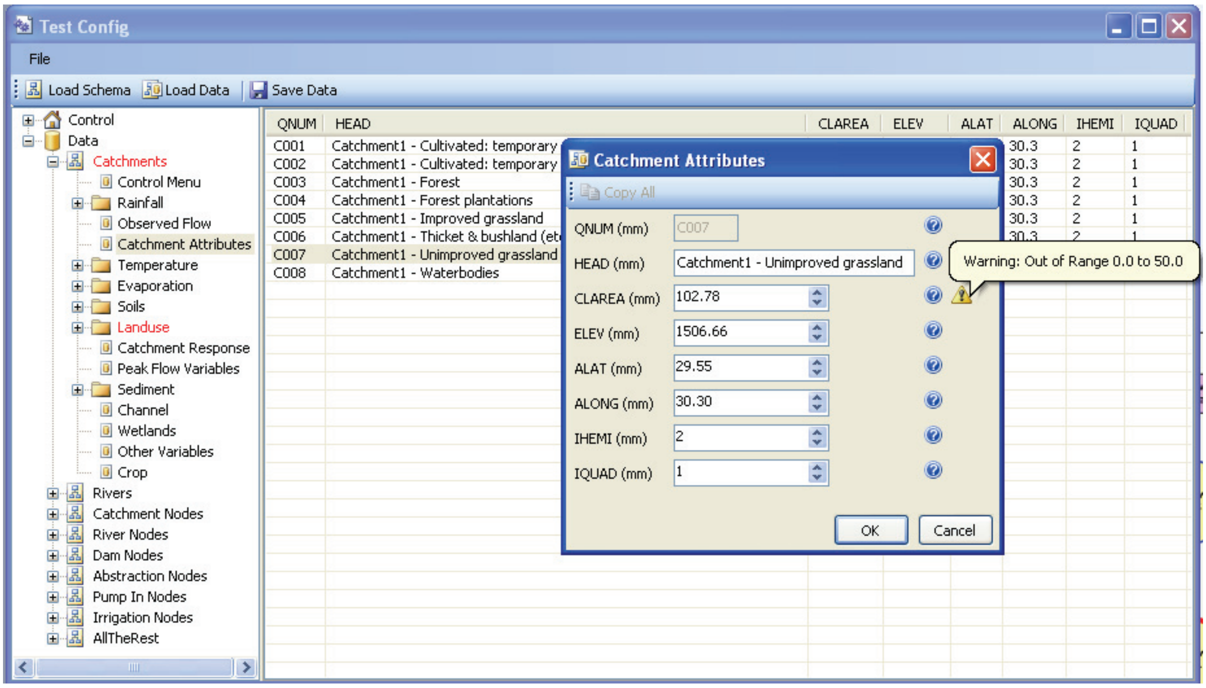


Figure 3.9: Editing data in the Configuration Editor

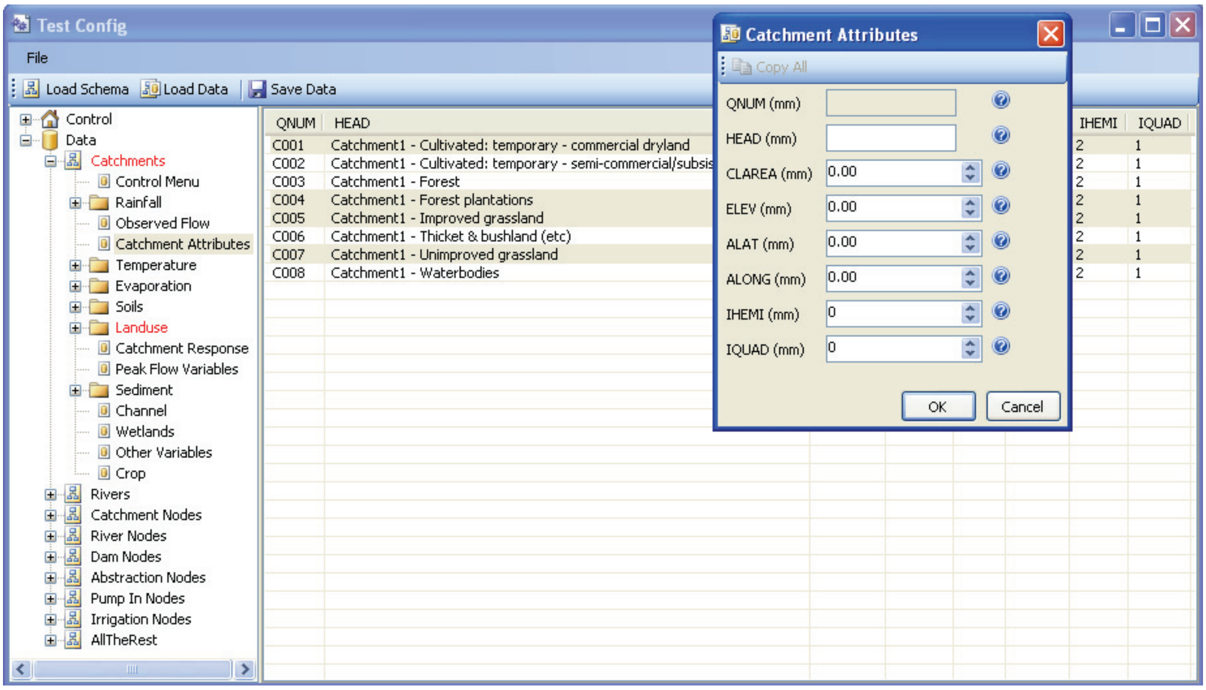


Figure 3.10: Editing multiple rows in the Configuration Editor

3.4.6 Setting up the Control of Entities

In a natural hydrological cycle entities such as catchments, rivers and dams are linked through the hydrological processes. In the Configuration Editor these entities are linked through the Control Menu shown in Figure 3.11. In Figure 3.11 Catchment C004 is linked with the downstream Dam Node DN1. The set up of the Control Menu is managed by the information on linkable entities entered into the configuration schema.

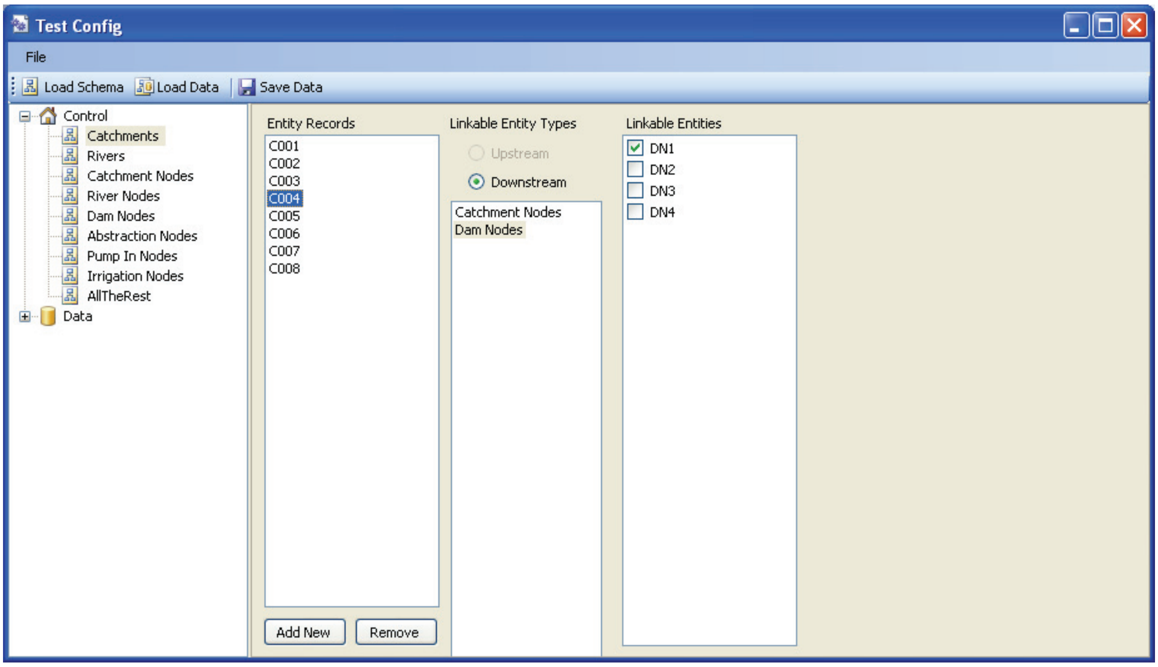


Figure 3.11: Control Menu options in the Configuration Editor

3.4.7 Conclusion

The Configuration Editor was developed as a prototype within this project to test the feasibility of developing such an editor that is designed in such a way that it is generic enough to be applied to almost any model. Thus far the development of the Configuration Editor has focussed on its use as an editor for setting up the *ACRU* model. However, the design of the Configuration Editor is generic enough that it could be used with other models if reader and writer objects were created for each specific model input file format. This prototype version of the Configuration Editor has been successful and warrants further development. The Configuration Editor is currently being further developed to be compatible with the new *ACRU* input data file format described in the Section 3.3.1 of this report, including being able to read data from and write data to a SPATSIM-HDSF database.

3.5 Hydrological Processes Incorporated into *ACRU*

In recent years WRC funded projects and other projects within the School of BEEH have lead to the development of modelling processes and modules being developed for the *ACRU* model. Many of these processes and modules are developed as part of MSc degrees and only exist in the form of research versions. These processes and modules are valuable additions to the *ACRU* model and require consolidation before being incorporated in the latest version of the *ACRU* model. This section of the report summarises the consolidation and further development of hydrological processes in the *ACRU* model that has taken place within this project, further details can be found in Appendices B.11, B.12 and B.13.

3.5.1 Consolidation of Dam and River Operating Rule Module

A set of Process classes modelling dam and river operating rules were developed for the *ACRU* model as part of an MSc Engineering project in the School of BEEH (Butler, 2001) and the developments are summarised by Butler et al. (2001). The National Water Act (NWA, 1998) of South Africa (Act 36 of 1998) includes a Reserve to provide a basic water supply to humans and the environment. The objective of the operating rule module developed for the *ACRU* model by Butler (2001) was to model the supply of water from a dam or river to multiple water users, including the Reserve. The operating rule module developed by Butler (2001) aims to:

- satisfy the requirements of water users in general
- satisfy the requirements of the basic human needs reserve
- satisfy the requirements of the ecological reserve by making artificial releases from a dam.

The operating rule module developed by Butler (2001) models the two components (human and environment) of the Reserve as water users (Butler et al., 2001). The framework developed by Butler (2001) identifies four types of water users:

- domestic users, representing the basic human needs component of the Reserve
- the environment, representing the ecological component of the Reserve
- industrial users
- irrigators.

Two different operating rule Processes were developed to supply water to the four types of water user:

- a generic dam operating rule, which models water abstracted from a dam
- a channel operating rule, which models water abstracted directly from a river (Butler, 2001).

These two operating rules determine the daily amounts each water user can receive through the use of a curtailment structure, where abstractions made by water users are limited based on the storage level in the dam or river (Butler, 2001). Each water user may request water from only one dam or river water source.

The computer code for the operating rule module was in the form of a research version and as part of the HDSF project was implemented in the most recent version of the *ACRU* model. The implementation of the operating rule module required some changes to the code of the framework and a thorough check that the system of management and curtailment level worked as expected. These changes included the concept of channel nodes to make it compatible with the most recent version of *ACRU*. Some of the Process classes developed in the operating rule module were specific to the case study at Paris Dam undertaken by Butler (2001) and these were omitted from this implementation in the *ACRU* model. A solicited presentation on the development of a dam operating rule module for the *ACRU* model to satisfy the requirements of the National Water Act in South Africa was made by Smithers et al. (2005).

3.5.2 Scoping of Extensions to the Dam and River Operating Rule Module

One limitation of the framework developed by Butler (2001) is that a request for water from a dam can only be supplied from a single dam. In order to operationalise the framework, it is necessary to be able to predict the lag and attenuation of flows in a river in order to supply a given volume at a given time at a point downstream of the dam. Therefore, a feasibility study was undertaken by the project in collaboration with Mr Butler from Partners in Development in order to:

- To investigate the feasibility of including functionality within the Operating Rule module to allow water users to be supplied from more than one water source
- To investigate the feasibility of including options in the Operating Rule module to account for the lags and attenuations associated with flow releases made from dams. This is of particular importance in meeting the requirements of the Ecological Reserve as required by the National Water Act (1998). Currently, the model is able to make releases from dams in an attempt to satisfy the Ecological Reserve at a Reserve site downstream from a dam. However, the releases do not take into account the flow lags and attenuations that will occur between the dam and the Reserve site.
- To investigate the latest developments regarding the implementation of environmental operating rules from dams specifically to meet the requirements of the Ecological Reserve
- To assess whether potential modifications made to the Operating Rule module would enable it to be used as an operational tool in the management of water resources within South Africa.

In order to assess the practicability of including the above functionality within the *ACRU* Operating Rule module, various models with some or all of the abovementioned capabilities were investigated in order to determine the best methods of handling the above problems. The models investigated included Mike Basin (developed by DHI Water and Environment), the Water Resources Yield Model (WRYM) and a model developed by Mallory and Van Vuuren (2007).

3.5.2.1 Description of the Current *ACRU* Operating Rule Module

The operating rule module, developed by Butler (2001), consists of four major components: the water user, the water request, the operating rule and the water transfer. Four types of generic water users are considered:

- Domestic user - the domestic user represents the Basic Human Needs Reserve, as defined by the National Water Act (1998)
- Environment - the environment represents the Ecological Reserve, as defined by the National Water Act (1998)
- Industrial user - the industrial user represents any industry that requires water in order to operate
- Irrigator - the irrigator represents any water user who requires water for irrigation purposes.

In order to obtain water, the four generic users must make a daily water request for their required quantity from the relevant water source (either a dam or a river). The methods for determining the quantity requested differ for each user. It is important to note however that currently the users may only request water from one water source during a simulation.

Two operating rule procedures are included within the Operating Rule module in order to determine whether or not the demands of each user can be met by the particular water source. The Generic Dam Operating Rule is responsible for supplying water from a dam while the Channel Operating Rule supplies water directly from a river. Both rules recognise two types of water present in the water source: unallocated and allocated water. Unallocated water has no owner and hence is available for use by any user abstracting from the relevant water source. However, allocated water has been assigned an owner and thus this water may only be abstracted by the particular water user that owns this volume of water. In the framework's current form, allocated water has most relevance concerning the release of high-flow events as part of the Ecological Reserve. The volume of the flood to be released is determined and this volume of water is then set as allocated storage. The flood is then released over a specified duration drawing from the allocated storage until this volume is drawn down to zero on the final day of the flood release.

The framework makes use of a curtailment structure in order to determine whether or not a user's demands can be met. Table 3.2 shows an example of a curtailment structure that would be used by the framework.

Table 3.2: Example of a curtailment structure used by the Operating Rule module

Management level (%)	**100 - 80**	**80 - 70**	**70 - 60**	**60 - 40**	**40 - 10**	**Priority**
Domestic	100	100	100	100	100	1
Environment	100	100	100	95	85	2
Industry	100	100	100	90	80	3
Irrigation user 1	100	100	90	80	50	4
Irrigation user 2	100	90	75	75	50	4

In Table 3.2, the percentage values in bold are termed management levels and represent a dam's current storage as a percentage of full capacity. Thus 100% represents full capacity and 10% in Table 3.2 represents dead storage. The values in the table represent the corresponding curtailment levels existing within a given management level. The curtailment level is the percentage of the user's request quantity that is likely to be supplied and hence is reliant on both the current level of storage in the dam and the priority of the user. The curtailment structure is only applied to requests from unallocated storage. Any water that has been allocated to a water user is not subjected to the curtailment structure and hence the full amount can be supplied to the user if requested.

Once the Operating Rule module has determined the quantity of water that can be supplied to a user, the water is transferred from the water source to the water user, either via a direct abstraction from the source, or by making a downstream release into the river reach below where the source is a dam.

3.5.2.2 Review of Existing Models with Capabilities for Supplying Water Users from Multiple Sources

Two of the most common techniques used when modelling complex water resource systems are linear programming and rule-based simulation. Linear programming is used by models such as the Water Resources Yield Model (WRYM) whereas rule-based simulation modelling is used by models such as Mike Basin. In both techniques, the water resource system is usually represented by a flow network, but the way in which water is allocated to water users differs. Broadly speaking, the linear programming approach employs penalties and seeks to obtain a solution that minimises the penalties incurred, whereas the rule-based approach prioritises the sources and supplies within the system and solves iteratively (Mallory and Van Vuuren, 2007). The principles of the WRYM, Mike Basin and a model developed by Mallory and Van Vuuren (2007) are discussed briefly in the following sections.

Water Resources Yield Model (WRYM)

The WRYM has been developed within the Department of Water Affairs and Forestry (DWAF) for the purpose of planning and operating South Africa's water resources (de Jager and Van Rooyen, (2007). The model employs a monthly time-step in order to model complex water resource systems using a flow network. Water users are able to draw from a number of water sources in order to meet their monthly requirements.

A network solver within the model is used to assess the water resource system's long- and short-term yield (de Jager and Van Rooyen, 2007). Penalties are dimensionless values that are assigned to all abstractions made from a water source (rivers or dams) as well as to the various routes throughout the network along which water can flow (Van Rooyen et al., 2003). The value of individual penalties is therefore not important but rather the relative value of penalties to one another. This is because the network solver attempts to find a solution where the smallest value of penalties is incurred (Van Rooyen et al., 2003). Penalties are also assigned for having units of water available within reservoirs. The solver again will attempt to minimise penalties incurred by comparing the benefit of having water in one reservoir with the benefit of having it in another, as well as considering penalties that might be incurred if the water is transported versus whether certain water users in the system are not supplied (Van Rooyen et al., 2003). The final solution opted for by the model will therefore result in the lowest total penalty being incurred and is calculated by adding up flows multiplied by penalties for all elements in the system (Van Rooyen et al., 2003).

The development of operating rules in the WRYM is therefore an iterative process. An initial set of operating rules is chosen and the model is run. The results produced are then evaluated, the operating rules are changed and the model is run again until a satisfactory output is produced (Van Rooyen et al., 2003). According to De Jager and Van Rooyen (2007), the main strength of the WRYM lies in the fact that the configuration of the system network and the relationships between its elements are defined by means of input data, rather than by fixed algorithms embedded within the complex source code.

Mike Basin

The Mike Basin model has been developed by DHI Water and Environment and is capable of modelling complex multi-source, multi-user water resource systems. It is a network model in which the river system is represented by a network of branches and nodes. The branches represent individual stream sections while the nodes represent confluences, bifurcations or locations where certain water activities may occur (CPH Water, 2007). Setting up and editing of the flow network is done on screen via ArcGIS software.

Input files are used to define the characteristics and operating rules for each reservoir (CPH Water, 2006). According to CPH Water (2006), allocation algorithms are used to determine how water is distributed among water users. Each water user represented within the water resource system is assigned a priority with the result that certain water users will receive priority over others. Each individual water user is also able to assign priorities to the water sources from which it is allowed to abstract water. This means that the user is able to specify the sequence in which abstractions should be requested from the available water sources. CPH Water (2006) suggest that water users will typically tend to first use water from those sources that do not provide them with a high level of assurance. The source with highest priority will supply the entire or partial demand of the user before the next highest priority source is considered. The facility to handle priorities from a user perspective as well as from the resource perspective makes the model extremely flexible in its application (CPH Water, 2006).

The Mike Basin model employs two different approaches to water resource management. The first is the "Curtailment" system where operating rules are based on the storage levels in the various available water sources. The second is the "Fractional Water Allocations and Capacity Sharing" or "FWACS" system where each water source is divided into storage volumes allocated to each water user. The water user is therefore able to use their allocation as and when required. Both systems have various advantages and disadvantages. However, CPH Water (2006) considers that the use of the "FWACS" system provides the water users with an incentive to conserve water when the levels of the water sources start to decline. Conversely, use of the "Curtailment" system means that it is to the water user's advantage to use as much water as possible before, for example, the levels of a dam drop into the next set of curtailment levels when the user's demand would be further curtailed. A disadvantage of the "FWACS" system is that it is extremely complex to set up as well as manage (CPH Water, 2006).

Additional pertinent features of the operating rules used by the Mike Basin model, taken from CPH Water (2006) are:

- Each reservoir is split into four operating zones. The uppermost portion is the flood control zone. If the dam level reaches this zone, water is automatically released from the dam to reduce the risk of dam wall failure. The normal operating zone falls below the flood zone and has a set of curtailment levels for each month of the year for each water user. The third zone is the minimum operational level. When the reservoir reaches this level, water may only be accessed from the reservoir for environmental or social reasons. Dead storage comprises the fourth zone and when this level is reached, no water can be abstracted from the reservoir.
- The model employs remote flow rules which allow interaction between nodes that are situated at different points in the catchment i.e. they are not neighbouring nodes. These rules can be used either where a control point exists far downstream from the reservoir, possibly with many intermediate inflows and offtakes, or where users are supplied by a reservoir that is far upstream.
- The storage demand rule allows two reservoirs to be operated in series or in parallel. If two reservoirs are located on the same river branch in series, it is usually an advantage to keep as much water as possible in the upstream reservoir. Alternatively, if two reservoirs are situated in parallel, the storage demand rule allows for flow between the two reservoirs depending on their water levels.

Other Models

Mallory and Van Vuuren (2007) have developed a model capable of handling complex water supply systems where users are able to draw water from multiple water sources. The model uses rule-based simulation and has been developed and tested in South Africa. Mallory and

Van Vuuren (2007) consider the model to be an extension of the techniques used by the Mike Basin model.

According to Mallory and Van Vuuren (2007), the model uses a cascading solution to solve the water resource system from upstream to downstream. Shortages experienced by each water user are determined in a first iteration while a second iteration supplies the water users' shortages from the available water sources such that a mass balance is achieved at each node for each time step. The model makes the assumption that users will first make use of river flow before requesting releases from dams (Mallory and Van Vuuren, 2007).

3.5.2.3 Recent Developments Regarding Operating Rules for the Ecological Reserve

The ability of a model to account for flow lags and attenuations is of particular importance if the needs of the Ecological Reserve as required by the National Water Act (1998) are to be met. The release of floods from dams to satisfy the instream flow requirements at a Reserve site downstream require that a particular flood peak be obtained at the site in as short a time interval from when a "real" event would have occurred naturally. However, if the Reserve site happens to be a distance downstream of the dam, attenuation of the flood will occur before the flood reaches the Reserve site. Thus, the flood peak at the Reserve site will be of smaller magnitude than what is required by the Ecological Reserve. A need therefore exists to be able to account for this attenuation before the flood is released from the dam to ensure that the desired peak is obtained at the Reserve site. According to Hughes et al. (2007), the accurate estimation of hydrograph attenuation and losses depends largely on the available information on the channel cross-section, longitudinal slope and hydraulic roughness characteristics between the release point and the Reserve site.

Hughes et al. (2007) consider the Muskingum method to provide a useful approximation of flood attenuations for a wide range of natural river systems. Hughes et al. (2007) state that the main problem with using the Muskingum method is the determination of the values of the storage (X) and routing (K) coefficients that are applicable to the specific reach. The most straightforward approach suggested by Hughes et al. (2007) is to use a set of trial releases to calibrate the coefficients of the model. While the coefficients are likely to vary with the size of the event, it should be possible to make an acceptable approximation of the value of the coefficients with three trial releases covering moderate sized events. Hughes et al. (2007) also note that if there are tributaries entering the river between the dam and the Reserve site, the tributary inflows can contribute to satisfying the Reserve if inflow events occur at the same time as releases from the dam are made into the river. A programme has been developed by Hughes et al. (2007), as part of the SPATSIM package, that is able to estimate the level of contribution that tributaries have made (or will make) under certain conditions.

Another fairly recent development regarding the release of floods from dams is to specify a range of floods for the year instead of a maintenance or drought flood of set peak and volume for particular months. According to Hughes et al. (2007), the high flow requirements of a Reserve determination are based on maintenance flow conditions occurring throughout a year. Therefore, in wetter years, larger (or more) events will occur, while in drier years the events may be curtailed with fewer events occurring or lower peaks being used. Each specified flood is classed according to its frequency of exceedance. A range of peak flows for each flood is also specified as well as the duration, the month on which the flood should be released and if the flood is not released for a particular month, the number of months for which the flood can be carried over. A scale factor can also be applied to the flood to account for losses / attenuation effects as well as contributions from tributaries entering the system between the dam and the downstream point at which the flood's peak flow is required.

Another recent development by Hughes et al. (2007) relates to the curtailment of demands on users who are abstracting water directly from a river. According to Hughes et al. (2007), in certain situations, it is not possible to achieve a satisfactory water balance within a model analysing the yield of a system without making allowance for seasonal variations in the curtailment rules. Therefore, a monthly set of curtailment values is used for each water user.

3.5.2.4 Discussion

The primary focus of this investigation has been to assess the feasibility of installing functionality within the *ACRU* Operating Rule module to allow water users to be supplied from more than one water source. However, before drawing conclusions on this issue, it is necessary to first discuss the techniques employed by the various models reviewed in order to assess which would provide the best solution for *ACRU*.

The WRYM has received widespread use throughout South Africa, especially by DWAF. However, the WRYM suffers from a number of limitations. Firstly, although the model has been set up specifically to deal with water resource management in South Africa, according to Mallory and Van Vuuren (2007), not many systems are actually operated as modelled in the WRYM. Secondly, the model runs off a monthly time step. This poses a problem when modelling small dams where variations in dam levels occurring within a month can be considerable. Apparently, a daily time-step version of the WRYM is currently under development which would eliminate this particular problem. Thirdly, the penalty structure employed by the WRYM is subjective and often does not relate well to the manner in which a catchment is actually operated (Mallory and Van Vuuren, 2007). Mallory and Van Vuuren (2007) consider the techniques used by the WRYM to be robust in that most conceivable situations can be solved. However, they note that a great deal of thought must be put into setting up the penalty structure and that intense testing of the system is required to ensure that the model is performing as intended. Mallory and Van Vuuren (2007) further suggest that the penalty structure approach employed by the WRYM is more applicable for planning purposes in catchments where the operating rules are not clearly defined, and that for operational analyses, where operating rules are either well understood or need to be developed, rule-based simulation offers a better solution.

In contrast to the approach taken by the WRYM, Mallory and Van Vuuren (2007) consider the advantages of rule-based simulation to be that it is easy to set up, especially for those less acquainted with numerical modelling and that it relates well to how catchments are managed and operated in practice. The Mike Basin model uses rule-based simulation in order to model complex water resource systems. The programme has received widespread use throughout the world, is easy to use in that it allows the user to set up very complex systems through the ArcGIS interface and also allows the user an amount of flexibility in terms of choosing how the water resource system should be managed. The model developed by Mallory and Van Vuuren (2007) also makes use of rule-based simulation and appears to be a very capable method of handling complex water resource systems. The fact that the model has been developed and tested in South Africa also means that it is well equipped to deal with local conditions.

Taking the abovementioned issues into account, when assessing the method to use from an *ACRU* perspective, and given the requirement that the Operating Rule module should aim to be useful not only as a planning tool but also for operational purposes, use of the linear programming / penalties approach used by the WRYM is therefore not recommended. Assuming that rule-based simulation was chosen as the most appropriate method to use in order to meet the requirements of this investigation, the next decision would be to determine whether such functionality should be included within *ACRU* itself or alternatively, whether *ACRU* should simply be linked to another model that already has these capabilities. *ACRU* has already been linked to the Mike Basin model by CPH Water (2007). However, for this link to be useful, model users would still have to purchase the Mike Basin model at a cost of approximately R75 000. Options of linking *ACRU* to the model developed by Mallory and Van Vuuren (2007) could also be investigated which would certainly provide a cheaper option than using Mike Basin. However, from the research conducted as part of this study, there does not appear to be any reason why it would not be possible to upgrade the *ACRU* Operating Rule module to be able to model more complex water resource systems.

From the investigations undertaken, all reviewed models with complex water resource system modelling capabilities have one thing in common i.e. they are all based on a flow network. The first alteration required to the existing Operating Rule module would therefore be to represent it as a flow network within *ACRU* which should be possible from a programming perspective. However, it is strongly recommended that this development be coupled with some form of graphical user interface that would assist in setting up the flow networks as part of the Operating Rule module. After consulting various individuals who have experience with both *ACRU* and other models such as the WRYM and Mike Basin, a major concern raised with installing complex water resource system functionality within the *ACRU* Operating Rule module, is the time required by the model user to set up the *ACRU* input menus. *ACRU* input menus are already fairly complex and are difficult to use if the user is not familiar with the model. The inclusion of additional functionality into the model to allow for modelling complex water resource systems would further complicate and lengthen the *ACRU* input menus. However, if a simple user interface could be developed for use with the Operating Rule module, it would not only save time when setting up a complex water system, but would also enable users who are not familiar with the model to run simulations fairly easily. Furthermore, if the Operating Rule module was to be marketed to potential users who would be making use of it from an operational point of view, the graphical user interface would form an essential tool in the daily operation of the water resource system.

Another issue to consider would be the method to be used by the Operating Rule module to manage the distribution of water within the water resource system. As mentioned above, two methods exist i.e. “Curtailment” and “FWACS”. The FWACS system is excellent conceptually. However, it has not yet been adopted as “the norm” where water resource modelling in South Africa is concerned. Furthermore, in the context of *ACRU,* considerable re-programming of the existing Operating Rule module would be required in order to implement this system. This option would therefore not be recommended. Since the current Operating Rule module is already based on the “Curtailment” system, this seems to be the best option for *ACRU*. Consideration should be given to altering the existing curtailment structure used by the Operating Rule module to be able to vary a user’s curtailment from month to month. This would also improve the method of meeting the Ecological Reserve requirements in the framework in cases where users are abstracting water directly from a river, because the curtailments on the users would account for seasonal variations in the river.

The Operating Rule module is currently made up to two main processes, the Generic Dam Operating Rule and the Channel Operating Rule. The Generic Dam Operating Rule would probably not need to be changed much in terms of the algorithms employed therein. However, the Channel Operating Rule algorithms would need to be modified. The current format of the Channel Operating Rule is oversimplified as users are curtailed based on the level in the river relative to the monthly maintenance and drought flows at a designated Reserve site that can be situated anywhere along the river within the catchment. Thus, the position of the abstraction point relative to the Reserve site is not taken into account. In order to be more realistic, the curtailment of the user should be based on the available water at the abstraction point (i.e. water not already allocated to the environment or other downstream users). This is a key issue as with most of the models reviewed, water users usually attempt to satisfy their demands from the river before requesting water from dams.

A method to account for flow lags and attenuations in the *ACRU* framework should also be possible to implement using a variation of the Muskingum routing algorithms which are already included within *ACRU*. However, one issue to consider in this regard would be the ability to "hot start" *ACRU*. If during a simulation, the criteria for releasing a flood from a dam were met, a variation of the Muskingum routing algorithms would have to then be activated such that the flood peak and duration to be released from the dam were determined in order to achieve the correct peak and duration of elevated flows at the Reserve site downstream. Having determined these parameters, the model would then have to switch back to the operating rule and release the flood. This functionality could also be used for releasing the daily low-flow for the Ecological Reserve and would eliminate problems of timing inherent within the existing framework where an environmental release is requested from a dam on one day but is only released on the following day. The ability to "hot start" *ACRU* would also be essential if the model was to be used in an operational capacity. A dam operator for example will not be able to run the model on a daily basis in order to determine the daily low-flow release required in order to meet the Ecological Reserve given that a simulation run over a number of years on a large catchment would take several hours to complete. However, if the model was able to store information from the previous day's simulation, then it would only be necessary to run the model for a single day.

3.5.2.5 Conclusions and Recommendations

The main conclusion to be drawn from this investigation is that it should be possible to further develop the *ACRU* Operating Rule module such that it is able to allow water users to draw water from more than one source as well as account for flow lags and attenuation of flow releases made from dams.

Assuming that this feasibility study is implemented, recommendations for future work would therefore be as follows:

- Conversion of the existing Operating Rule module algorithms into a format that is representative of a flow network should be carried out
- The existing Operating Rule module algorithms should be altered to account for the supply of users from multiple sources. Additional functionality should also be included to allow for varying curtailment levels on a monthly basis
- A graphical user interface should be developed within *ACRU* to assist with setting up the flow networks required by the Operating Rule module
- The Channel Operating Rule, used for abstracting water directly from rivers, should be altered to make it more representative of a user's position within a catchment

- Algorithms to account for flow lags and attenuations should be implemented within the new Operating Rule module
- The option to “hot start” *ACRU* should be implemented.

3.5.3 Consolidation of the Salinity Modelling Module

A salinity module (*ACRUSalinity)* was developed for the *ACRU* model as part of an MSc project in the School of Bioresources Engineering and Environmental Hydrology (BEEH) (Teweldebrhan, 2003; Teweldebrhan et al., 2003). The objective in developing the *ACRUSalinity* module was to provide a modelling tool to assist in the assessment of the impact of land and water use alternatives on the salt balance of large and complex hydrological systems at a catchment scale (Teweldebrhan, 2003).

The *ACRUSalinity* module was in the form of a research version coded by Teweldebrhan, (2003) and, as part of the HDSF project, was consolidated into the most recent version of the *ACRU* model. The implementation of the *ACRUSalinity* module required some changes to the code of the module. These additions included the concept of channel nodes into *ACRUSalinity* to make it compatible with the most recent version of *ACRU*.

During implementation of the *ACRUSalinity* module it was noted that when water was abstracted from rivers and reservoirs there was no corresponding abstraction of salts. Changes were made to *ACRUSalinity* to model these salt abstractions. When water is abstracted for irrigation these changes allow abstracted salt loads to be applied to an irrigated area. These salt abstractions occur in the following instances:

- domestic abstractions from a river (IDOMR = 1, DOMABS(i) > 0.0)
- draft from a dam (XDRAFT(i) > 0.0)
- irrigation from a river or dam (IRRIGN = 1, IRRAPL = 1 or 2).

Note that if inflows to a dam are specified using the PUMPIN variable there is at present no means of specifying a corresponding salt inflow. However, users may specify a daily time-series of water inflows to a dam or river and a corresponding daily time-series of inflow salinity values.

In addition, some code was added to several Process classes in the *ACRUSalinity* module to ensure that the salt processes were correctly synchronised with corresponding water processes. In *ACRU* corresponding water and salt processes are modelled in separate Process classes and it is necessary to maintain a “pseudo” water balance for use in certain salt processes.

3.5.4 Further Development of the Salinity Modelling Module

As part of the Water Research Commission (WRC) project K5/1301 titled, “Development of technical guidelines for water quality use allocation procedures under the NWA through application of the Berg River water quality information system”, several subsequent enhancements to the *ACRUSalinity* module were requested by Ninham Shand Consulting Services. These enhancements to the *ACRUSalinity* module were completed on a collaborative basis between project K5/1301 and the HDSF project. The enhancements to the *ACRUSalinity* module reported in this document have dealt mainly with catchment scale salinity processes. Similar enhancements to the salinity processes for irrigated areas should be considered as an item for future development.

3.5.4.1 Monthly Salt Generation and Saturation Input Variables

The *ACRUSalinity* module simulates the generation of dissolved salts from geological sources in soil horizons and the groundwater zone. Generation of dissolved salts ceases once a saturation level is reached. The salt generation process is modelled based on user specified salt generation rate and saturation concentration input variables. These input variables were initially constant values for each soil horizon and groundwater Component object and have been changed to enable users to specify different values for these variables for each month of the year.

3.5.4.2 Conditional Abstraction of Water from a Channel

A new Process class named *PConditionalChannelWaterAllocation* has been added to the *ACRUSalinity* module. The purpose of the *PConditionalChannelWaterAllocation* Process is to limit the quantity of water available for abstraction from a channel based on flow and salinity conditions in the channel. If the salinity modelling option is selected (SALINITY=1) then the *PConditionalChannelWaterAllocation* Process is called instead of the standard *PSimpleChannelWaterAllocation* Process for channel water allocation. If the conditional abstraction option variable for the river is turned off (CCONAB=0) then the *PConditionalChannelWaterAllocation* Process will calculate the same water abstraction quantity and allocations as the standard *PSimpleChannelWaterAllocation* Process. The *PConditionalChannelWaterAllocation* Process cannot currently be used if the flow routing option is selected (IROUTE>0).

The user may specify a set of flow bands, a corresponding set of allowed abstraction quantities and the months in which these conditions are to be applied. An example of a conditional abstraction table based on flow conditions for a channel is shown in Table 3.3, and the representation of this table as variables in a land segment menu is shown in Figure 3.12 The flow band and conditional abstraction quantity variable arrays must be the same size. The simulated salinity in the channel on the previous day is used to determine the salinity band.

Table 3.3: Example of conditional abstraction quantities based on flow bands for a channel

Flow Band		**Allowed Abstraction**	
Simulated flow >= 0 and <= 1000	m^3/day	0	m^3/day
Simulated flow > 1000 and <= 2000	m^3/day	800	m^3/day
Simulated flow > 2000 and <= 3000	m^3/day	1100	m^3/day
Simulated flow > 3000	m^3/day	2000	m^3/day

```
4  Channel flow abstraction bands (m3)
CFBAND(1) 0.0
CFBAND(2) 1000.0
CFBAND(3) 2000.0
CFBAND(4) 3000.0
 4 Channel flow abstraction quantities (allowed abstractions) (m3)
CFQTY(1) 0.0
CFQTY(2) 800.0
CFQTY(3) 1100.0
CFQTY(4) 2000.0
```

Figure 3.12: Representation of a conditional abstraction table in a land segment menu for a channel based on flow bands

Similarly the user may also specify a set of salinity bands, a corresponding set of allowed abstraction quantities and the months in which these conditions are to be applied. An example of a conditional abstraction table based on salinity conditions for a channel is shown in Table 3.4, and the representation of this table as variables in a land segment menu is shown in Figure 3.13. The salinity band and conditional abstraction quantity variable arrays must be the same size.

Table 3.4: Example of conditional abstraction quantities based on salinity bands for a channel

Salinity Band		**Allowed Abstraction**	
Simulated salinity >= 0 and <= 18	mg/ℓ	3000	m^3/day
Simulated salinity > 18 and <= 30	mg/ℓ	2000	m^3/day
Simulated salinity > 30	mg/ℓ	1000	m^3/day

```
3  Channel Salinity abstraction bands (mg/l)
CSBAND(1) 0.0
CSBAND(2) 18.0
CSBAND(3) 30.0
 3  Channel Salinity abstraction quantities (allowed abstractions) (m3)
CSQTY(1) 3000.0
CSQTY(2) 2000.0
CSQTY(3) 1000.0
```

Figure 3.13: Representation of a conditional abstraction table in a land segment menu for a channel based on salinity bands

It is assumed that if both flow and salinity abstraction conditions are applied then the abstraction quantity allowed is the minimum of the two values. If the available water in the channel is less that the allowed abstraction value then the allowed abstraction value is set equal to the available water. The conditional abstraction quantity is the total abstraction quantity from the river for the day being simulated and portions of this quantity will be allocated to the water users that have requested water from the channel.

3.5.4.3 Conditional Abstraction of Water from a Dam

A new Process class named *PConditionalDamWaterAllocation* has been added to the *ACRUSalinity* module. The purpose of the *PConditionalDamWaterAllocation* Process is to limit the quantity of water available for abstraction from a dam based on salinity conditions in the dam. If the salinity modelling option is selected (SALINITY=1) then the

PConditionalDamWaterAllocation Process is called instead of the standard *PSimpleDamWaterAllocation* Process for dam water allocation. If the conditional abstraction option variable for the dam is turned off (DCONAB=0) then the *PConditionalDamWaterAllocation* Process will calculate the same water abstraction quantity and allocations as the standard *PSimpleDamWaterAllocation* Process. The *PConditionalDamWaterAllocation* Process cannot be used if the flow routing option is selected (IROUTE>0).

The user may specify a set of salinity bands, a corresponding set of allowed abstraction quantities and the months in which these conditions are to be applied. An example of a conditional abstraction table based on salinity conditions for a dam is shown in Table 3.5, and the representation of this table as variables in a land segment menu is shown in Figure 3.14. The salinity band and conditional abstraction quantity variable arrays must be the same size. The simulated salinity in the dam on the previous day is used to determine the salinity band.

Table 3.5: Example of conditional abstraction quantities based on salinity bands for a dam

Salinity Band		**Allowed Abstraction**	
Simulated salinity >= 0 and <= 40	mg/ℓ	8000	m^3/day
Simulated salinity > 40 and <= 55	mg/ℓ	6000	m^3/day
Simulated salinity > 55	mg/ℓ	2000	m^3/day

```
3  Dam Salinity abstraction bands
DSBAND(1) 0.0
DSBAND(2) 40.0
DSBAND(3) 55.0
 3  Dam Salinity daily abstraction quantities (allowed abstractions)
DSQTY(1) 8000.0
DSQTY(2) 6000.0
DSQTY(3) 2000.0
```

Figure 3.14: Representation of a conditional abstraction table in a land segment menu for a dam based on flow bands

If the available water in the dam is less that the allowed abstraction value then the allowed abstraction value is set equal to the available water. The conditional abstraction quantity is the total abstraction quantity from the dam for the day being simulated and portions of this quantity will be allocated to the water users that have requested water from the dam.

3.5.4.4 Addition of a Soil Surface Layer

In initial versions of the *ACRUSalinity* code one of the base assumptions was that the stormflow had the same salinity as that of the rainfall. However, stormflow salinity can vary greatly from the rainfall salinity, especially after periods of no rainfall, when salinity in the soil near the surface increases due to evaporation. When a rainfall event occurs, accumulated salts near the surface of the soil mix with the stormflow resulting in a higher salinity than that of the rainfall.

In order to account for the above process, a thin soil surface layer Component and some new Process classes have been added to *ACRUSalinity*. The soil surface layer Component (CSoilSurfaceLayer) is a type of soil layer (CSoilLayer) Component. As a result of adding a soil surface layer some of the core *ACRU* Process and model structure classes as well as some of the *ACRUSalinity* Process classes needed to be modified. A schematic diagram of water flow processes related to soil layers is shown in Figure 3.15. The following assumptions were made regarding soil surface layers:

- When using *ACRUSalinity* soil water evaporation (Es) and transpiration (Et) are calculated separately (EVTR=2)
- When using *ACRUSalinity*, it is assumed that there is a thin soil surface layer approximately 10mm thick
- The soil surface layer does not contain any roots and therefore no transpiration takes place from the soil surface layer
- There is no salt uptake (i.e. salt generation from the geology) in the soil surface layer.

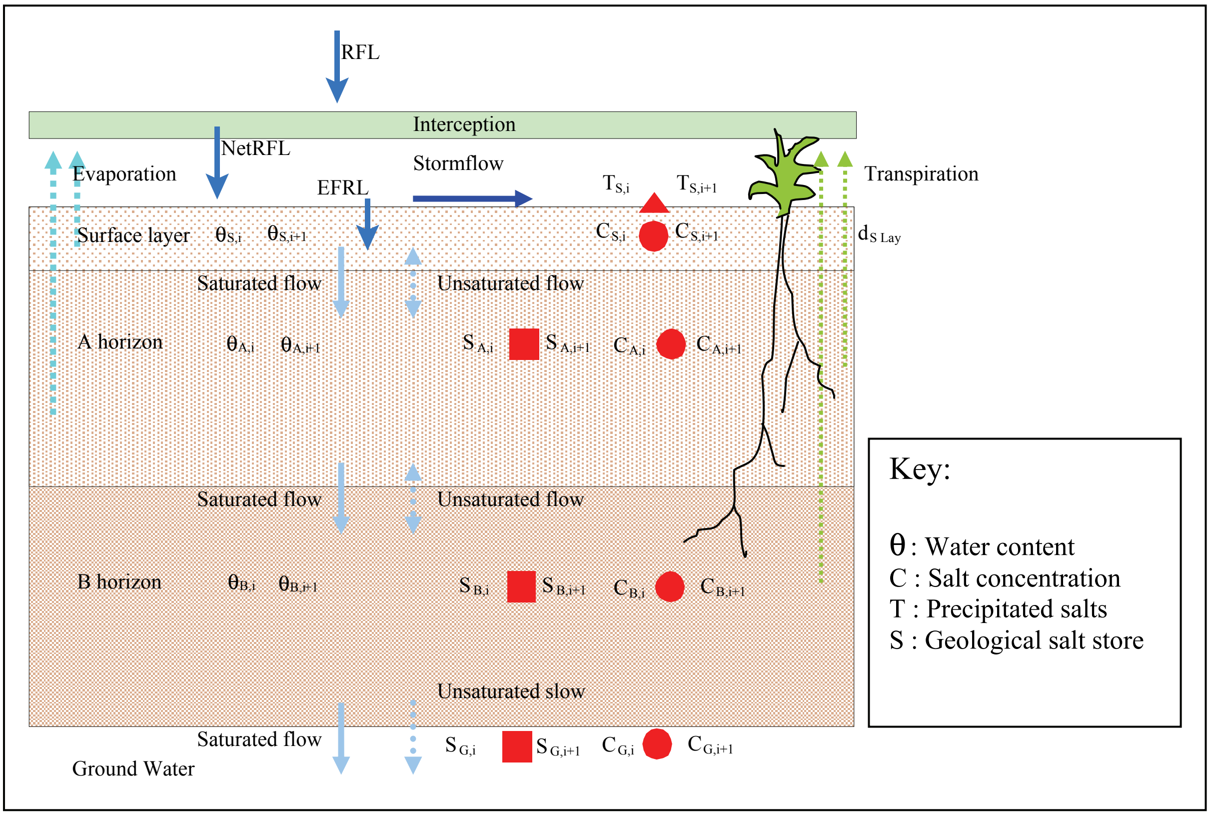


Figure 3.15: Schematic diagram of water flow processes related to soil layers

It is important to note that when using *ACRUSalinity* the processes the soil surface layer Component can cause significant changes to the simulated water quantity results if incorrectly parameterised. It is recommended that catchments first be modelled with *ACRUSalinity* turned off (i.e. without a soil surface layer). When the user is satisfied that the water related processes are being modelled correctly, then *ACRUSalinity* should be turned on and the results compared to the previous results to check that the water related processes are still being modelled correctly.

3.5.4.5 Mixing of Salts in the Soil Surface Layer with Stormflow

When a rainfall event occurs, accumulated salts near the surface of the soil mix with the stormflow resulting in a higher salinity than that of the rainfall. The PStormflowSalinity Process was modified to model this process. If the net rainfall after interception is greater than zero then the mixed stormflow salinity is calculated using Equation 3.1.

$$DISS = (SL_{RFL} + SL_{s\,lay} + SL_{inflows})/(\theta_{s,iVol} + NetRFL_{vol} + WV_{inflows}) \quad (3.1)$$

where
- $DISS$ = Mixed stormflow salinity [mg/ℓ]
- SL_{RFL} = Salt load of rainfall (rainfall volume * rainfall salinity) [mg]
- $SL_{s\,lay}$ = Dissolved salt load in soil surface layer [mg]
- $SL_{inflows}$ = Salt load from any other surface inflows [mg]
- $\theta_{s,iVol}$ = Water content of soil surface layer as a volume [ℓ]
- $NetRFL_{vol}$ = Volume of net rainfall (gross rainfall – interception) [ℓ]
- $WV_{inflows}$ = Water volume from any other surface inflows [ℓ].

Even though a portion of the rainfall may be intercepted, the salt load associated with the intercepted rainfall is assumed to still contribute to the system and is therefore included in the calculation of stormflow salinity.

The salinity of the stormflow and quickflow are set equal to the mixed stormflow salinity, however, the quickflow salinity is then readjusted based on the delayed stormflow and its corresponding salinity.

3.5.4.6 Precipitation of Salts in Soil Layers when Salt Saturation is reached

If the concentration of dissolved salts in the soil surface layer exceeds the saturation concentration of salts then these excess dissolved salts are precipitated out of solution and accumulated by *ACRUSalinity* in a precipitated salt store. The PSoilLayerSaltPrecipitation Process was created to model this process.

3.5.4.7 Dissolving of Salts in the Soil Surface Layer during Rainfall Events

During a rainfall event a portion of the precipitated salts accumulated in the soil surface layer dissolves back into the solute state, based on rainfall event contact time and soil characteristics. The PStormflowSalinity Process was modified to model this process. If the net rainfall after interception is greater than zero then the salt load dissolved from the soil surface layer's precipitated salt store is calculated using Equation 3.2 from Sharpley et al. (1981).

$$DISS1 = K_{diss} * T_{s,i} * (t^{\alpha}) * WR^{\beta} \quad (3.2)$$

where
- $DISS1$ = Salt load dissolved out of the precipitated salt store [mg], (If $DISS1 > T_{s,i}$ then $DISS1 = T_{s,i}$)
- K_{diss} = Salt dissolution constant
- $T_{s,i}$ = Salt load of the precipitated salt store [mg]
- t = Runoff event contact time [min]
- α = Alpha – salt dissolution constant
- β = Beta – salt dissolution constant
- WR = Water to solid ratio ($[NetRFL + \theta_{s,i}] / d_{s\,lay}$) [m/m]
- $\theta_{s,i}$ = Water content of soil surface layer as a depth [m]
- $d_{s\,lay}$ = Soil surface layer depth [m]
- $NetRFL$ = Net rainfall (gross rainfall – interception) [m].

3.5.4.8 Salt Movement with Unsaturated Redistribution of Water in Soil Layers

Initially *ACRUSalinity* only modelled the movement of dissolved salts associated with the movement of water between soil layers during saturated soil moisture conditions. The PUnsaturatedFlowSaltTra Process has been added to *ACRUSalinity* to model the movement of dissolved salts associated with upward and downward movement of water between soil layers during unsaturated soil moisture conditions. This Process is only applied if the unsaturated soil water redistribution option in *ACRU* has been switched on (IUNSAT = 1).

3.5.4.9 Salt Movement due to Evaporation of Water from Soil Layers

Evaporation of water from the soil results in the accumulation of salts near the surface of the soil. The PSoilWaterEvaporationSaltTrans Process has been added to *ACRUSalinity* to model this process. In *ACRUSalinity* evaporation of soil water takes place from both the soil surface layer and the A-horizon. The dissolved salts associated with water evaporated from the A-horizon are moved from the A-horizon to the soil surface layer.

3.5.5 Consolidation of Sugarcane Yield and Irrigation Module

In order to link the hydrology, irrigation scheduling and state-of-the-art sugarcane growth and yield modelling, a module (*ACRUCane*) was developed, verified and included in the *ACRU* model as part of an MSc Engineering project in the School of BEEH (Moult, 2005). A summarised description of the *ACRUCane* module can be found in a draft paper by Moult et al. (2006b), which is included in this report as Appendix B.12. The objective in developing the *ACRUCane* module was to provide a modelling tool to assist in the assessment and management of catchment water supply and demand interactions, and the associated impacts on the profitability of irrigated sugarcane (Moult, 2005). *ACRUCane* simulates the water budget of an irrigated field of sugarcane when irrigated with different types of irrigation systems and estimates the resulting yields (Moult, 2005). The algorithms used to model the water budget in *ACRUCane* are based on refinements of algorithms from FAO 56 (Allen et al., 1998) and the *ACRU* model. *ACRUCane* includes algorithms to simulate both sugarcane yield and quality to enable the impact of different management practices, type of irrigation system, water supply limitations and environmental conditions on sugarcane yield to be simulated.

New features integrated in the *ACRUCane* module developed by Moult (2005) include:

- evapotranspiration from a cropped surface using the dual crop coefficient methodology described by Allen et al. (1998)
- sugarcane root growth
- sugarcane canopy development
- interception associated with different types of irrigation system
- fraction of the soil surface wetted by different types of irrigation system
- irrigation uniformity index associated with different types of irrigation system
- irrigation scheduling option with different fixed-amount-fixed-cycle scheduling parameters for specified summer and winter periods
- irrigation scheduling option to irrigate a fixed amount at a specified soil moisture depletion level
- sucrose yield and biomass accumulation yield algorithms from CANEGRO (Singels and Bezuidenhout, 2002)

- Estimated Recoverable Crystal (ERC)
- sugarcane yield using the equation used in CANESIM (Singels et al., 1999).

This work has subsequently been presented at a number of conferences (SANCIAHS conferences in 2005, 79th SASTA conference in 2005, 80th SASTA conference in 2006, SANCID conference in 2006) and also written into papers (Moult et al., 2005a; Moult et al., 2005b; Moult, 2005; Lecler et al., 2006; Moult et al., 2006a; Moult et al., 2006b) and two papers have been submitted for publication in a journal (Moult et al., 2007a; Moult et al., 2007b).

In this project, further developments to the *ACRUCane* module has resulted from collaboration between Dr Neil Lecler, from the South African Sugarcane Research Institute (SASRI), and the project team at the School of BEEH. This collaboration involved the application and further debugging of the *ACRUCane* module developed by Moult (2005). The outcome of this collaboration is contained in the paper by Lecler et al. (2007) which is included in Appendix B.12 and is summarised below.

The objective of the collaboration was to demonstrate that the quantification of the water balance in an integrated irrigation systems context is essential to assessing the performance of irrigation and water management systems, in addition to the commonly used irrigation performance indices. Using the sugarcane yield model developed by Moult (2005), which is embedded in the *ACRU* model, the water balance of an irrigated field was simulated with water supplied by a dam. Four scenarios were simulated and the water balances and yields for each scenario was analysed:

(i) furrow irrigation with performance and management characteristics typically observed in practice
(ii) furrow irrigation with improved performance and management characteristics representative of top performing furrow irrigation systems
(iii) centre-pivot irrigation with performance and management characteristics typically observed in practice
(iv) centre-pivot irrigation with improved performance and management characteristics representative of top performing centre pivot irrigation systems.

In the study performed by Lecler et al. (2007) a perspective of irrigation systems performance is provided. To illustrate this perspective, the *ACRU* agrohydrological simulation model was used to estimate the water balance and various fates of applied irrigation water for typical and top performing furrow irrigation systems and these were compared to equivalent typical and top performing centre pivot irrigation systems. The characteristics of the different systems, such as the uniformity of applied water, were derived from representative in-field evaluations. Often, when irrigation efficiencies are quoted or interpreted, the fates of applied water are poorly specified or appreciated, leading to the widespread belief that water just disappears with low irrigation efficiencies and will re-appear with improvements. As demonstrated in the study, such beliefs are an over-simplification. Often the amount of water actually consumed in irrigation, i.e. the evaporated component, remains relatively unchanged at various levels of 'efficiency', or can even increase when changing to irrigation systems which are often perceived to be relatively more efficient. Simulations showed that reductions in non-beneficial water balance components were affected more by design and management considerations than by the type of irrigation system. Changing from furrow irrigation to centre pivots, which are often perceived to be more efficient, did result in slightly improved crop yields. However, the availability of water to other users supplied from a downstream dam was less, owing to the timing and magnitude of associated irrigation return flows and the higher evaporation losses simulated under the pivots. The study concluded that the quantification of the water balance in an integrated systems context is thus essential to assess the performance of irrigation and water management systems and should be promoted rather than promoting the plethora of often misinterpreted irrigation performance indices. This application was also a useful validation exercise and resulted in some corrections and adjustments being made to the *ACRUCane* module.

3.5.6 Root Colonisation

The concept of root colonisation was introduced to the *ACRU* 331 version of the *ACRU* model as a result of work done by Summerton (1995). Root colonisation is defined in Summerton (1995) as the fraction of the soil matrix under consideration to which roots have ready access to any available soil moisture. If a soil layer is 100% colonised then roots may utilise available soil moisture from 100 % of that layer. In the *ACRU* model the topsoil horizon is assumed to be 100% colonised and the percentage colonisation of the subsoil horizon is an input parameter. The potential transpiration or the potential evapotranspiration from a soil layer is multiplied by the fraction of the soil layer that is colonised and results in reduced transpiration from the soil layer. The concept of root colonisation has now been incorporated into the most recent version of the *ACRU* model.

3.6 *ACRU* User Support

Table 3.6 summarises the support provided by the project team to users of the *ACRU* model and other data and software related support. From the extensive list in Table 3.6 the hydrological modelling expertise within the project team and School of BEEH is evident and crucial to hydrological and water quality modelling in South Africa. The extensive use of the agrohydrological information generated by the School of BEEH is also evident.

Table 3.6: Summary of AC*RU* and other user support

User	Type of Support
EVN Africa Consulting Services (Pty) Ltd	Configuration of the *ACRU* model for a catchment in Limpopo province to determine runoff, inflows and outflows used in estimating siltation of a dam
Irrigation and Drainage Services CC	• AAHMS & Daily rainfall installation and basic operation • AAHMS Quaternary Extraction Query • Configuration of the *ACRU* model in Quaternary Catchment V20H to determine a suitable dam size
Partners in Development	Dam and river operating rules
South African Sugarcane Research Institute	Use of the *ACRUCane* module
Umgeni Water	• Configuration of the *ACRU* model • Correcting *ACRU* menus • Reformatting data • Use of QC database
WRP Consulting Engineers	Configuration of the *ACRU* model in the Mhlathuze Catchment
Department of Geoinformatics, Hydrology and Modelling Institute of Geography Friedrich-Schiller- University Jena Germany	Configured *ACR*U for the Limpopo and Letaba Catchments

Environmental and Geographical Science University of Cape Town	Using Quaternary Catchments version of *ACRU* in the Berg River Catchment with GCM data to assess adequacy of GCM's
DWAF	• Impacts of land use on hydrological responses in WMA12 • Course on the *ACRU* model, held in Pretoria (Aug 2007)
Bergstan Consultants	Irrigation in the Thune Catchment, Botswana
Futureworks/Institute of Natural Resources	Maluti Ecosystems Goods and Services project
Pieter van Heerden	Data for WRC SAPWAT project
Pegram and Associates	Data for WRC Project K5/1683 – "Soil Moisture from Satellites"
Ninham Shand Consulting Services	• Assistance with *ACRU* data format related query • Assistance with *ACRU*2000 conditional abstractions • Use of the *ACRUSalinity* module and configuration of Berg River Catchment
Projects in School of BEEH, UKZN	General user support to staff and students working on several other research projects, in particular: • WRC Project K5/1318 – "Development and assessment of a continuous simulation modelling system for design flood estimation" • WRC Project K5/1428 - "An Investigation and Formulation of Methods and Guidelines for the Licensing of SFRA's with Particular Reference to Low Flows" • WRC Project K5/1489 - "Development of Interactive, Updated and Updateable, Extended Atlas of South African Agroclimatology and Agrohydrology" • WRC Project K5/1562 - "Regional Aspects of Climate Change and their Secondary Impacts on Water Resources" • WRC Project K5/1646 - "Applications of Rainfall Forecasts for Agriculturally Related Decision Making in Selected Catchments" • National Department of Agriculture Project - "Potential First to Fourth Order Impacts of Climate Change on Agricultural Production in South Africa, and Adaptation Options" • International Water Management Institute Project: "Smallholder System Innovations in IWRM"
Undergraduate teaching in School of BEEH, UKZN	Teaching undergraduate students and Hydrology Honours students.
Swedish Meteorological and Hydrological Institute	Course on the *ACRU* model
Various consultants	Course on the *ACRU* model, held in Pietermaritzburg (July 2007)
Jeffares & Green Inc	Compilation of *ACRU* files and running model
Climate Systems Analysis Group (CSAG) Environmental & Geographical Science University of Cape Town	Recommendations to correct software bug
MBB Consulting Services (PMB) (Pty) Ltd.	Changes to catchment area for SCS-SA model
Partners in Development	Setup of and running *ACRU*2000
SRK Consulting Engineers	• Configuration and operating *ACRU* in Zululand catchments • Query re outputting monthly data and use of ACRUView
CSIR	*ACRU* input file preparation and use of AAHMS
eThekwini Municipality	Climate change and streamflows

3.7 Summary

Although model development was a secondary objective of the project, significant model development work was completed. Four of the most commonly existing models within SPATSIM, namely the Pitman model, Pitman model for parameter exploration, Desktop Reserve model and Stressor-Response model, were converted to work with the new SPATSIM-HDSF database data model. In addition the *ACRU* model was modified to run from the SPATSIM-HDSF modelling framework. Modifications to *ACRU* to run from SPATSIM-HDSF included the development of XML based model input data and model configuration files. These model input data and model configuration files have been designed such that the same file design could be used for other hydrological models. A prototype model configuration editor was developed for the *ACRU* model, but was also designed such that it could potentially be used for other models if a suitable configuration schema file is created for each model.

As part of this component of the project several hydrological processes were incorporated into the *ACRU* model. Research versions of three modules were consolidated and incorporated into *ACRU*, namely the dam and river operating rule module, the *ACRUSalinity* module and the *ACRUCane* module. The dam and river operating rule module aims to model the supply of water to general water users taking into account the basic human needs reserve and ecological reserve requirements of the National Water Act (NWA, 1998) of South Africa (Act 36 of 1998). This framework has some limitations with regard to its use for operational modelling, therefore an investigation was conducted to determine the feasibility of: (1) modelling water supply to a user from more than one water source, (2) accounting for lags and attenuations associated with flow releases made from dams, (3) updating the dam operating rules related to meeting the ecological reserve, and (4) using the framework as an operational tool in the management of water resources in South Africa. The *ACRUSalinity* module was further developed in collaboration with another WRC funded project. More detailed processes were added to *ACRUSalinity* to better model salinity in the hydrological system and a means of setting rules regulating the abstraction of water from dams and rivers based on flow and salinity conditions. User support was provided by the project team to users of the *ACRU* model and other data and software related support.

4. DATABASES

One of the objectives of this project was to compile a collection of readily available datasets in order to support the use of the SPATSIM-HDSF modelling framework and the models that run within it. The objective was not to develop a national database of hydrology in South Africa, as this was beyond the scope of this project, but to rather package with the SPATSIM-HDSF modelling framework relevant datasets which are available from the School of BEEH including products generated by other WRC projects undertaken by the School of BEEH. The data provided by this project consists of two main components: a database of climate related time-series data per quaternary catchment, and the *ACRU* Quaternary Catchments Database (AQCD), both containing data compiled for the South African Atlas of Climatology and Agrohydrology described in Schulze (2007a).

4.1 Climate Time-series Database

One of the primary inputs required by all hydrological models is rainfall. Another important variable in hydrological modelling is the estimation of evapotranspiration which is usually calculated from observed or estimated reference potential evaporation, soil moisture availability and physiological characteristics of the vegetation in the study area. The climate time-series database consists of daily time-series of rainfall, maximum temperature, minimum temperature, solar radiation, maximum relative humidity, minimum relative humidity and Penman-Monteith reference crop evaporation for each of the 1946 Quaternary Catchments in South Africa. Each time-series contains 50 years of daily data for the period from 1950 to 1999. A brief description of each variable is included below and more detailed descriptions can be found in Schulze (2007a). Obtaining or generating quality controlled time-series of climate related data is often a difficult and time consuming component of any modelling project. It is expected that this database of time-series data and information will be widely used by users of the SPATSIM-HDSF modelling framework. This data has been packaged in two forms: in a SPATSIM-HDSF database for ease of use by SPATSIM-HDSF users and as *ACRU* Composite (Y2K) Input Data files.

4.1.1 Description of Data

This section contains brief descriptions of the daily time-series datasets included in the databases stored in Appendices C.2 and C.3 on the DVD accompanying this report.

4.1.1.1 Rainfall

The daily rainfall dataset included in this database was derived from the comprehensive daily rainfall database for South Africa compiled by Lynch (2004) for the period 1950 to 1999 using data from 12 153 daily rainfall stations. More detail can be found in Lynch and Schulze (2007). Extensive error checking of the dataset and infilling of missing data was completed by Lynch (2004). The selection of a representative rainfall station from this dataset for each Quaternary Catchment is reported in Schulze et al. (2007b). In summary, the centroid for each catchment was determined and then the 10 closest rainfall stations to the centroid of each Quaternary Catchment were selected. From these rainfall stations one representative rainfall station was selected for the catchment based on several criteria including: mean annual precipitation (MAP), altitude, reliability of record, end year of record and distance of rainfall station from the centroid of the catchment. The result of this selection process was that 1248 rainfall stations were selected to represent the 1946 Quaternary Catchments (Schulze et al., 2007b). It is important to note that these rainfall time-series contain the station (i.e. uncorrected) rainfall for each Quaternary Catchment and should be used together with a correction factor such as the *CORPPT* variable in *ACRU* to calculate corrected rainfall values for a specific Quaternary Catchment.

4.1.1.2 Temperature

Temperature is a basic climatological parameter and as it is easy to measure it is frequently used in the estimation of other energy related parameters such as solar radiation, relative humidity and potential evaporation (Schulze and Maharaj, 2007). The daily maximum and minimum dataset included in this database was derived from a database of gridded daily maximum and minimum temperatures for Southern Africa described in Schulze and Maharaj (2004) and Schulze and Maharaj (2007). This database was compiled using records from over 970 temperature stations for the period 1950 to 1999 and where necessary missing data was infilled. The database was compiled at a spatial resolution of 1' latitude by 1' longitude for the area including South Africa, Lesotho and Swaziland. As described in Schulze et al. (2007b) a representative altitude was determined for each Quaternary Catchment and time-series of daily maximum and minimum temperatures were extracted from the gridded database for each Quaternary Catchment based on the representative altitude.

4.1.1.3 Solar Radiation

Potential evaporation and crop yields are strongly related to solar radiation (Schulze and Chapman, 2007). The daily solar radiation dataset included in this database was developed for use in the South African Atlas of Climatology and Agrohydrology (Schulze, 2007a). As explained in Chapman (2004), there is a relatively sparse and unrepresentative network of solar radiation measurement stations in South Africa, and records are frequently of short duration and unsuitable quality. Therefore, as described in Schulze and Chapman (2007), the daily solar radiation dataset was estimated using daily maximum and minimum temperature data in a modified version of the Bristow and Campbell (1984) equation.

4.1.1.4 Relative Humidity

Relative humidity is an important parameter affecting evaporation and transpiration. The daily maximum and minimum relative humidity dataset included in this database was developed for use in the South African Atlas of Climatology and Agrohydrology (Schulze, 2007a). The computation of this dataset is described in Schulze et al. (2007a).

4.1.1.5 Penman-Monteith Reference Crop Evaporation

The daily Penman-Monteith reference crop evaporation dataset included in this database was developed for use in the South African Atlas of Climatology and Agrohydrology (Schulze, 2007a). The computation of the of daily Penman-Monteith reference crop evaporation dataset and a brief summary of the background to the Penman-Monteith method is described in Schulze et al. (2007c). It is important to note that if this dataset is used in the *ACRU* model then values will have to be adjusted to A-pan equivalent reference potential evaporation values by setting an appropriate conversion factor (CORPAN parameter) in *ACRU*.

4.1.2 Description of Data Formats

4.1.2.1 Quaternary Catchments Shapefile

An ESRI shapefile named *Quaternaries.shp* using the WGS84 projection has been provided in Appendix C.1 as a spatial reference to the Quaternary Catchments. The *BEEH_ID* field in the *Quaternaries* shapefile contains the integer numbers from 1 to 1946 used to identify the individual Quaternary Catchments field and the *CATNUM* field containing the Quaternary Catchment number (e.g. *BEEH_ID*=1 corresponds to *CATNUM*=A10A).

4.1.2.2 SPATSIM-HDSF Database

A SPATSIM-HDSF database named *DailyClimateParameters* was created in Microsoft Access to store these time-series of climate parameters. A spatial feature named *Quaternary Catchments* was created and the shapefile named *Quaternaries.shp* is referenced in the *FeatureSource* field of the *data_dict1* table, where the *BEEH_ID* field in the *Quaternaries* shapefile is used as the identity field and the *CATNUM* field containing the Quaternary Catchment number is used as the description field. An attribute group named *Climate Data* was created in the *data_dict2* table. Attributes named *Rainfall*, *MaxTemperature*, *MinTemperature*, *SolarRadiation*, *MaxRelativeHumidity*, *MinRelativeHumidity* and *PenmanMonteithEvaporation* were created in the *data_dict3* table. The time-series datasets were imported into the database and stored in the *TimeSeriesData* table. The SPATSIM-HDSF database named *DailyClimateParameters* is stored in Appendix C.2.

4.1.2.3 *ACRU* Composite (Y2K) Files

The format of the *ACRU* Composite (Y2K) format files is described in (Smithers and Schulze, 2004). This format has been modified slightly as this database contains parameters (Penman-Monteith reference crop evaporation, and maximum and minimum relative humidity) that are not standard *ACRU* variables. The Penman-Monteith reference crop evaporation values are stored in the column previously used to store A-pan reference potential evaporation. The maximum relative humidity values are stored in the column previously used to store mean relative humidity, and the minimum relative humidity values are stored in the column previously used to store sunshine duration. The station identity is an 8 character string consisting of the latitude and longitude in minutes of a degree arc of the data point selected to represent the quaternary. A portion of one of the *ACRU* Composite (Y2K) format files is shown in Figure 4.1 and details of the format for each row are shown in Table 4.1.

The set of 1946 *ACRU* Composite (Y2K) format files containing daily climate parameters is stored in Appendix C.3. These *ACRU* Composite (Y2K) format files are named *obssim_XXXX.txt*, where *XXXX* is the Quaternary Catchment identity *BEEH_ID* number from 1 to 1946 (e.g. *obssim_0001.txt* is the file for Quaternary Catchment number *A10A*). These files are packaged as two self extracting WinRAR files named *obssim0000.exe* (containing files for Quaternary Catchments 0001 to 0999) and *obssim1000.exe* (containing files for Quaternary Catchments 1000 to 1946).

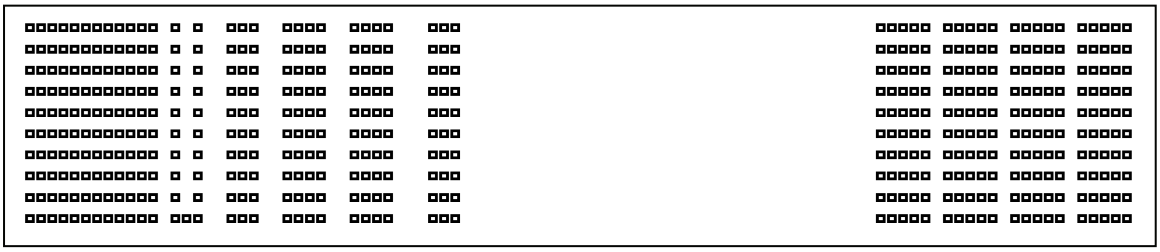


Figure 4.1: A portion of one of the *ACRU* Composite (Y2K) format files

Table 4.1: Format of the modified *ACRU* Composite (Y2K) format file

Character	Data Type	Description
1-8	8 character alphanumeric	Station identity
9-12	4 digit integer	Year
13-14	2 digit integer	Month
15-16	2 digit integer	Day
17-21	5 digit float with 1 decimal place	Observed rainfall (mm)
22	1 character alphanumeric	Data quality flag
23-27	5 digit float with 1 decimal place	Maximum temperature (°C)
28	1 character alphanumeric	Data quality flag
29-33	5 digit float with 1 decimal place	Minimum temperature (°C)
34	1 character alphanumeric	Data quality flag
35-39	5 digit float with 1 decimal place	Penman-Monteith reference crop evaporation values (mm)
40	1 character alphanumeric	Data quality flag
98-102	5 digit float with 2 decimal places	Solar radiation ($MJ/m^2/d$)
103	1 character alphanumeric	Data quality flag
104-108	5 digit float with 2 decimal places	Maximum relative humidity (%)
109	1 character alphanumeric	Data quality flag
110-114	5 digit float with 2 decimal places	Minimum relative humidity (%)
115	1 character alphanumeric	Data quality flag

4.2 *ACRU* Quaternary Catchments Database

The *ACRU* model described in Schulze (1995) has been modified to run in the SPATSIM-HDSF modelling framework as it is believed that a physical conceptual model operating at a daily time-step will be required to meet the modelling requirements of CMAs. The objective of the *ACRU* Quaternary Catchments Database (AQCD) is to provide a pre-populated database of default inputs for the *ACRU* model at a Quaternary Catchment scale. As stated in Schulze et al. (2007b), the AQCD enables the use of the *ACRU* model to perform simulations of, for example,

- stormflow
- baseflow, or
- total runoff, as well as
- impacts of land use change on hydrological responses, or of
- climate change on hydrological responses, of
- crop yields
- sediment yield
- irrigation water demand, or

- hydrological risk analyses

at a spatial scale of either

- individual Quaternary Catchments or, in the case of streamflows, for
- hydrologically linked, cascading catchments, thereby facilitating the simulation of cumulated flows from a series of individual Quaternary Catchments that make up a larger area.

The AQCD has been packaged in two forms: in a SPATSIM-HDSF database named *AcruQCDatabase* (Appendix C.4) for ease of use by SPATSIM-HDSF users and as an *ACRU* 300 series version text based input data file (Appendix C.5) with associated climate data files in *ACRU* Composite (Y2K) format. The reason for including the AQCD as an *ACRU* 300 series version text based input data file is that this will enable users to use the *ACRU* input data file converter program described in Section 3.3.4 of this report to populate a SPATSIM-HDSF database with a subset of the 1946 Quaternary Catchments. The *ACRU* 300 series version input data file in Appendix C.5 contains references to the *ACRU* Composite (Y2K) files in Appendix C.3 from which the rainfall and maximum and minimum temperature time-series are read.

4.2.1 Background

South Africa, Swaziland and Lesotho have been delineated by the South African Department of Water Affairs and Forestry (DWAF) into a hierarchical system of catchments starting with Primary Catchments, which in turn are disaggregated into Secondary, then Tertiary and finally into 1946 interlinked, cascading Quaternary Catchments (Schulze et al., 2005). The *ACRU* model requires various inputs, such as information on soils, land use and climate for each catchment being simulated. Meier (1997) established a database structure for the *ACRU* Input Database using a set of direct access ASCII files to store and reference the input data required for simulating hydrological responses from the Quaternary Catchments using the *ACRU* model. The data stored in the *ACRU* Input Database developed by Meier (1997) could then be extracted using a Fortran program into an *ACRU* input menu (file) which was in the form of a formatted ASCII file.

However, the structure of the *ACRU* Input Database was found to be limiting particularly with regard to the storage of data for additional catchments in instances where, for example, a Quaternary Catchment was further spatially disaggregated into Quinary Catchments (Schulze et al., 2005). Therefore the database which became known as the *ACRU* Quaternary Catchments Database (AQCD) was restructured as described in Schulze et al. (2005). This restructuring included reorganising the *ACRU* input information into a single spreadsheet file, enabling more flexible editing and manipulation of the *ACRU* input information in a familiar format. Each row in the spreadsheet referred to a catchment and each column to an *ACRU* input variable. Once the database had been established it could be exported from the spreadsheet in comma delimited format and subsequently converted into an *ACRU* input menu using a Fortran program.

The next phase of development involved the transfer of the AQCD into a database in Microsoft Access together with the associated Fortran programs required to create an *ACRU* input menu from data stored in Comma Separated Value (CSV) format (Schulze et al., 2005). At this point Microsoft Access was used to develop a Graphical User Interface (GUI) to assist users in the configuration of the *ACRU* model. This system in Microsoft Access including the AQCD and the GUI for configuring *ACRU* input menus became known as the *ACRU*

Agrohydrological Modelling System (AAHMS). The AAHMS includes a variety of options that can be specified for running simulations for the Quaternary Catchments using the *ACRU* model. More information regarding the AAHMS can be found in Smithers and Schulze (2004).

4.2.2 Data Selected for Baseline Quaternary Catchments Database

A baseline set of *ACRU* model inputs was developed by WRC project K5/1430, as reported in Schulze et al. (2005), and further refined in WRC project K5/1489, reported in Schulze (2007a). It is this baseline set of *ACRU* model inputs for each Quaternary Catchment that has been stored in a SPATSIM-HDSF database thus providing a populated baseline AQCD for users of the SPATSIM-HDSF modelling framework. The AQCD is intended primarily for use with the *ACRU* model. However, many of the variables, which are fully described in Schulze (1995) and Smithers and Schulze (2004), are variables that may be used in other hydrological models. The following is a summary of some of the more critical inputs included in the database:

- *Daily Rainfall* – The same daily rainfall dataset as was described in Section 4.1.1.1 was used in the AQCD

- *Daily Temperature* - The same daily maximum and minimum temperature datasets as were described in Section 4.1.1.2 were used in the AQCD

- *Estimation of Daily Reference Potential Evaporation* - The *ACRU* model enables the user to select from several different methods of inputting or estimating daily A-pan reference potential evaporation. According to Schulze et al. (2007b), observed daily reference potential evaporation was not available for all 1946 Quaternary catchments, and therefore the daily equation developed by Hargreaves and Samani (1985) was used due to the availability of temperature data.

- *Land Cover* - Some form of pristine baseline land cover type is required against which to evaluate the hydrological impact of new land uses or management (Schulze et al., 2007b). The 70 Acock's (1988) Veld Types for South Africa, Lesotho and Swaziland were selected as the baseline land cover and the dominant Veld Type for each Quaternary Catchment was determined (Schulze et al., 2007b). The relevant *ACRU* input variables for each Veld Type were determined by Schulze (2004).

- *Soils* - The soils parameters used were derived from the Land Types associated with the 84 Broad Natural Homogeneous Soils Zones delimited by the Institute for Soil, Climate and Water (ISCW) (Schulze, 1997). The soils parameters used in the *ACRU* model were determined for each of the Land Types as described in Schulze (2007b). The soils parameters to be used for each Quaternary Catchment were determined using area weighted averages.

The AQCD contains a complete set of default inputs for the *ACRU* model for each Quaternary Catchment and these values should be replaced by the user if better data and information are available. In addition, the AQCD contains inputs enabling users of the *ACRU* model to simulate crop yield (for maize, wheat, sugarcane and primary production), irrigation water demand and supply, and sediment yield (Schulze et al., 2007b).

4.2.3 Conversion to SPATSIM-HDSF Database Structure

The conversion of the AQCD from an *ACRU* 300 series version text based input data file to a SPATSIM-HDSF database was completed using the *ACRU* input data file converter program described in Section 3.3.4 of this report. The converter program automatically assigns each variable to an *ACRU* Component object each with a unique identity. Sets of *ACRU* Component objects with the same Component type are automatically represented as non-spatial features in the SPATSIM-HDSF database by the converter program. However, to enable all the variables to be viewed in a spatial context within the SPATSIM-HDSF modelling framework software, each of these non-spatial features has been manually edited to make them spatial features in which the shapefile named *Quaternaries.shp* is referenced in the *FeatureSource* field of the *data_dict1* table, where the *BEEH_ID* field in the *Quaternaries* shapefile is used as the identity field and the *CATNUM* field containing the Quaternary Catchment number is used as the description field.

5. CASE STUDY

A case study was conducted in which the SPATSIM-HDSF modelling framework was applied to a selected catchment as a demonstration and evaluation of the usefulness of the SPATSIM-HDSF modelling framework. It was agreed at the Reference Group meeting on 17 March 2006 that the project should concentrate on just one case study to demonstrate the use of the modelling framework developed. This decision was based on the realization that the case study needed to test the improvements made by the HDSF project to configure and model the hydrology in a catchment rather than to verify hydrological models on several different catchments. The Mgeni System was selected to leverage off work done concurrently by Umgeni Water (Summerton and Gillham, 2007; Summerton, 2008), who are also part of the project team for the HDSF project. The configuration of the *ACRU* model for the Mgeni System was updated in the study by Summerton (2008), and it was a subset of this configuration of the Mgeni System that was used for the case study. The unique relationship between BEEH and Umgeni Water is guided by principles set out in a Memorandum of Understanding between the two institutions.

The Mgeni catchment is situated in the Mvoti to Umzimkulu Water Management Area and is an economically important catchment ranging from the Drakensberg Mountains to the Indian Ocean at Durban (Kiker et al., 2006). Schulze et al. (2004) state that while the Mgeni catchment constitutes just 0.33% of South Africa, it supplies water to nearly 15% of the country's population and produces approximately 20% of the country's gross domestic product. The Mgeni catchment, shown in Figure 5.1, includes three major dams and two smaller water supply dams. These dams act as important water sources for the Pietermaritzburg and Durban municipal areas, as well as for many informal settlements and for agriculture (Kienzle et al., 1997). The Mgeni System includes water transfers from the Mooi River in the Thukela water management area, and in future could include water transfers from the Mkomazi River and Umzimkulu River (DWAF, 2004a). The Mgeni catchment contains a range of landuses including: natural veld, dry-land agriculture, irrigated agriculture, forestry, urban areas and informal settlements (Kiker et al., 2006). The Mgeni catchment is key to the local bulk water services provider, Umgeni Water, since most of the raw water used in operations is sourced from this catchment to supply the main water demand centres of Durban and Pietermaritzburg. These characteristics highlight the suitability of the Mgeni catchment for use in the case study component of the HDSF project.

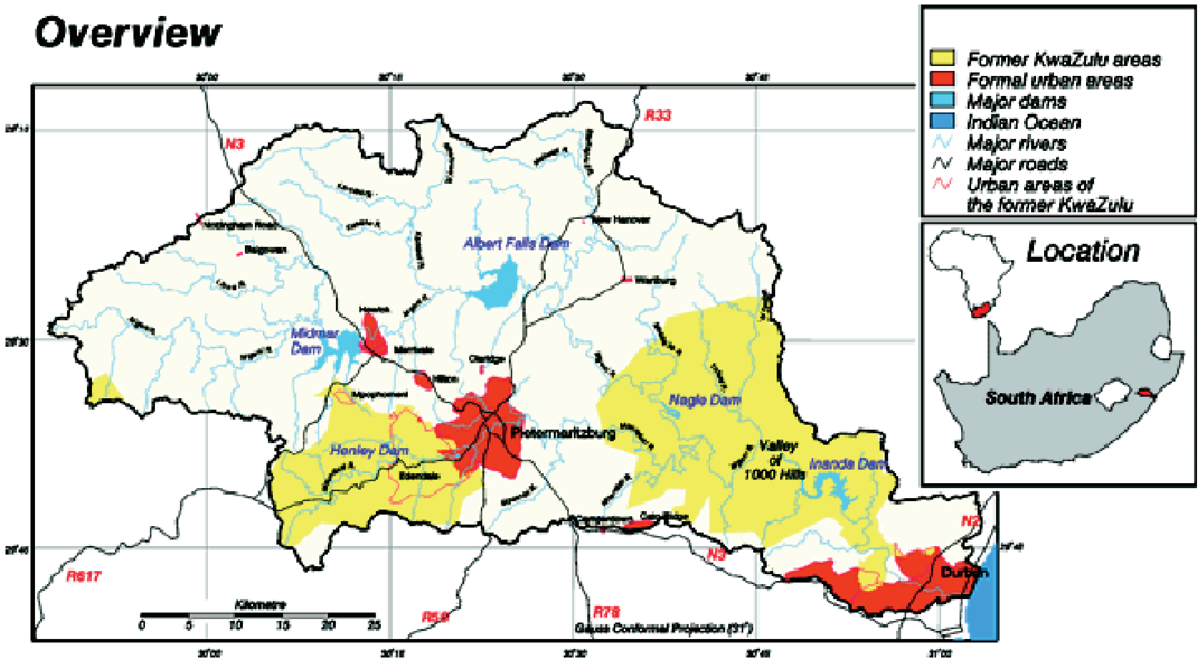


Figure 5.1: Overview of the Mgeni catchment (Schulze et al., 2004)

The last time the Mgeni System was fully configured for daily hydrological modelling was in 1994, using climate data up to the early 1990s and land use information from 1986 satellite imagery (Kienzle et al., 1997). Since this study the NWA with all its decision requirements has been promulgated, significant land use and population developments have taken place in the Mgeni catchment and the water demand/supply balance has shifted. Possibly the most important reason for re-configuring the model is the need to improve scenario planning for future water resources e.g. the impacts of climate change, in this strategic catchment, as identified at a workshop convened in September 2005 by Umgeni Water to identify the impacts of climate change on its operations.

Since *ACRU* was selected as the hydrological model, and the Mgeni as the test catchment, the aim of the case study was therefore to explicitly identify the value added to the hydrological modelling process by the HDSF project by comparing new and updated HDSF tools to conventional tools using the Mgeni catchment as an example.

5.1 Base Configuration

A daily hydrological modelling system has been configured by Summerton (2008), for the Mgeni catchment at a finer resolution than ever before to model the impacts of a changing climate on water resources. This includes 13 management catchments, subdivided into 145 hydrologically homogenous catchments (Figure 5.2). Ten landcover classes, mapped at a 1:5000 scale from orthophotos were included in each hydrological catchment, resulting in 1450 modelling sub catchments. Each of these has been configured within the *ACRU* hydrological model to cascade into each other in a logical sequence. The flow directions for these main sub-catchments are shown in Figure 5.3. The altitude ranges from 0-2013 m across the catchment (Schulze et al., 2004). The main rivers and the altitude across the catchment are shown in Figure 5.4. The mean annual precipitation for the Mgeni catchment is 902 mm and varies from 680 mm near the coast to 1220 mm in the mountains (Schulze et al., 2004).

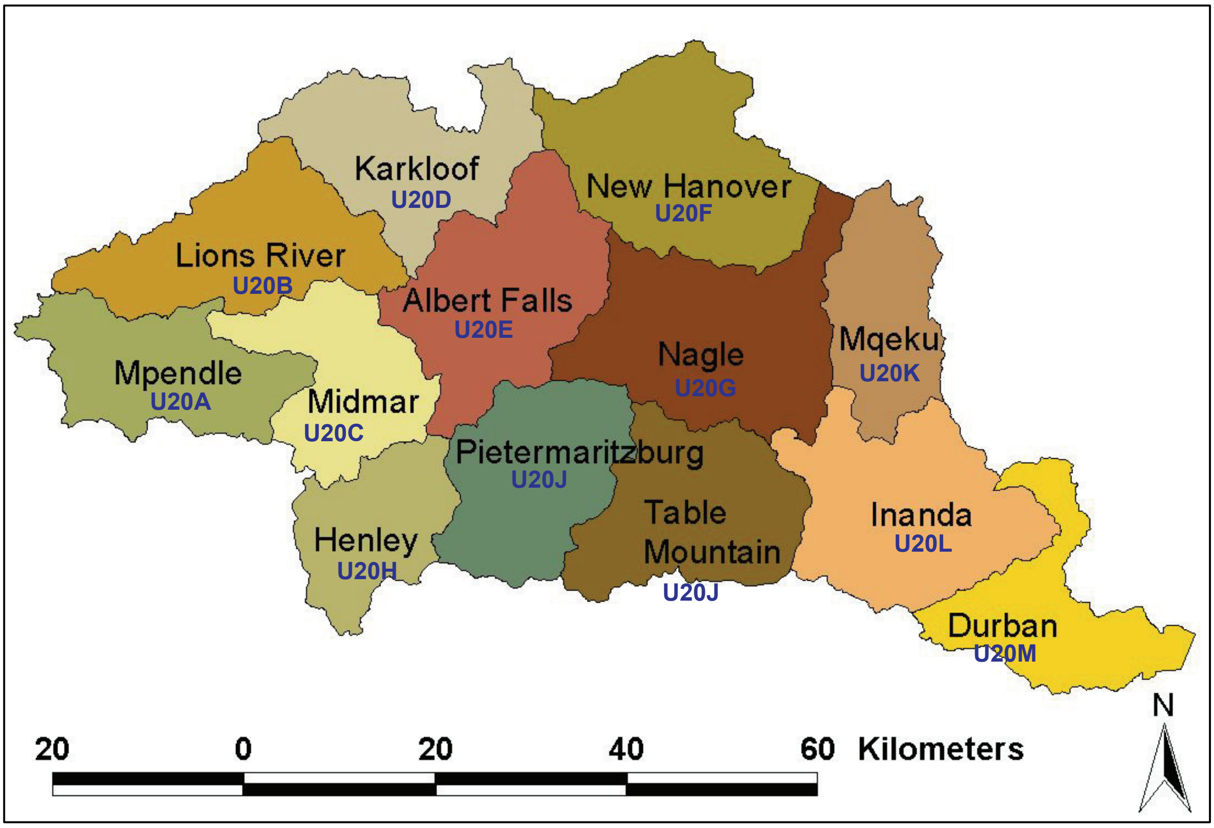


Figure 5.2: Main sub-catchments in the Mgeni catchment (Kiker et al., 2006)

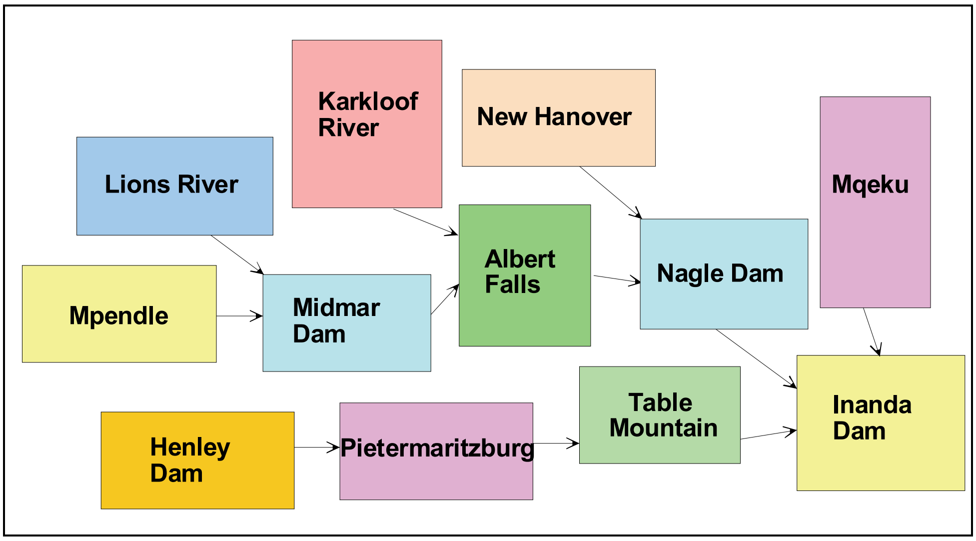


Figure 5.3: Flow directions for the main sub-catchments in the Mgeni catchment (Kiker et al., 2006)

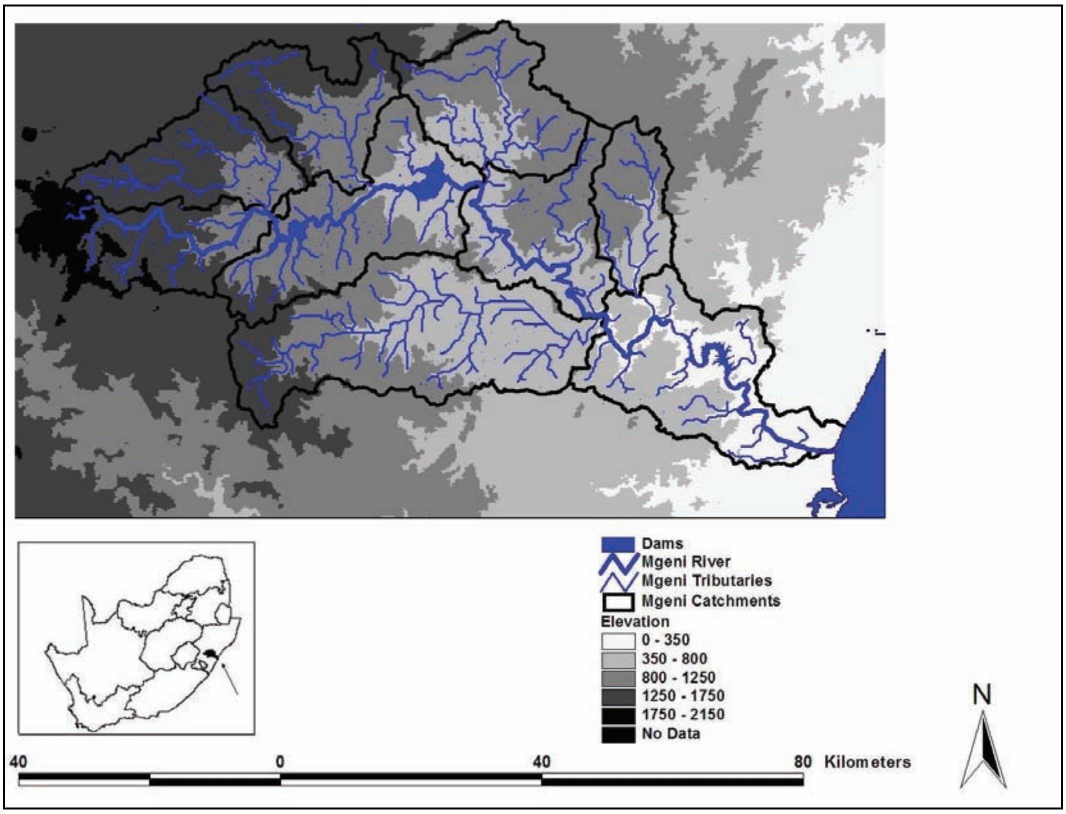


Figure 5.4: Altitudes and main rivers in the Mgeni catchment (Kiker et al., 2006)

An initial map of land cover was created in 1986 from SPOT satellite imagery (Bromley, 1989, cited by Kiker et al. 2006). An updated landcover map, shown in Figure 5.5, has recently been captured in GIS using ortho-photography dated 2003. Areas where interpretation was difficult were verified in the field. Expertise within the School of BEEH was utilised to assist with determination of suitable crop numbers and related hydrological model parameters, such as monthly crop coefficients, root distributions, and interception capacities, which define how each different land cover will use water in the hydrological model. Runoff, dam abstraction and rainfall data sets have been obtained from the Department of Water Affairs and Forestry, and were converted into the correct formats.

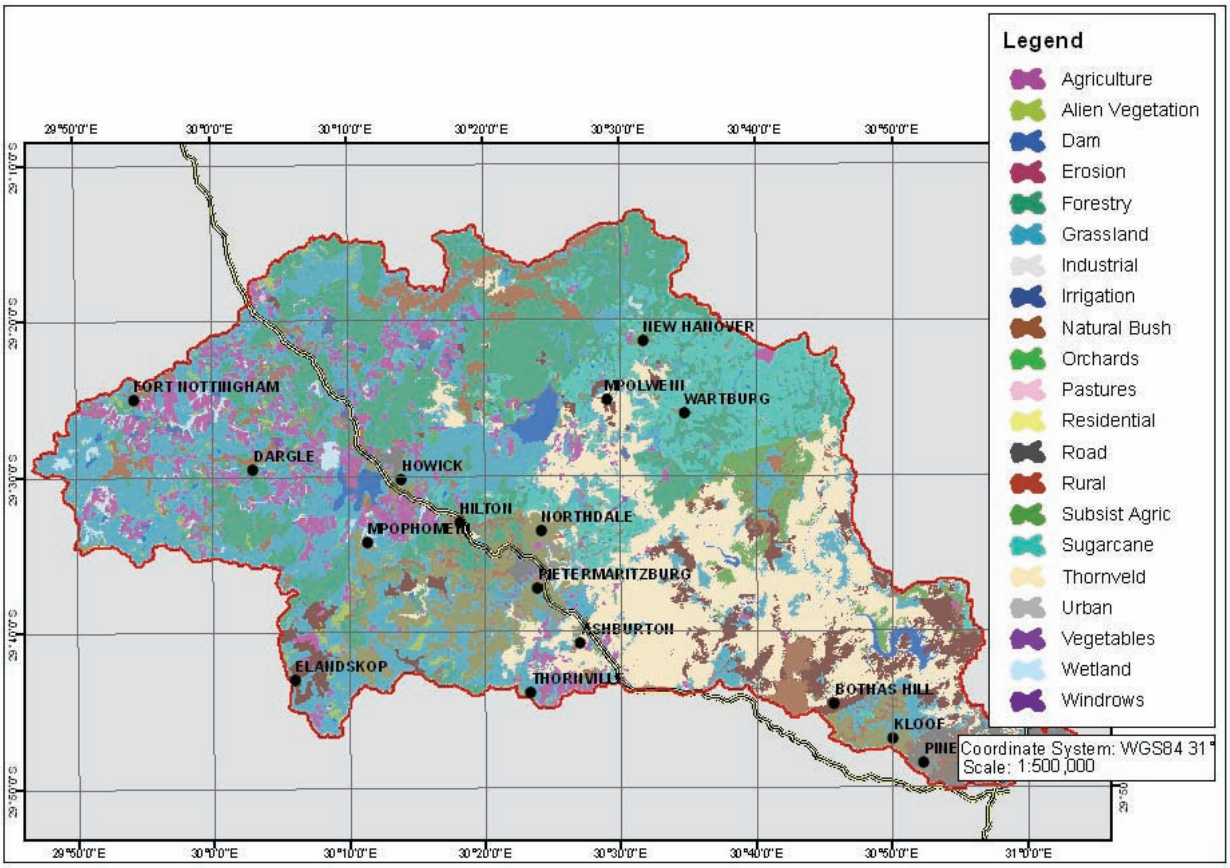


Figure 5.5: Updated landcover in the Mgeni catchment

The process used by Summerton (2008) is briefly summarised as follows and elaborated on below:

- The Amalgamation of Agro-Hydrological Modelling Groups (AAMG) software (Pott et al., 2007) was used to build a base configuration input dataset in the *ACRU* CSV file format
- These CSV files were imported into the AAHMS and converted to a standard *ACRU* 300 series input "menu" file
- The *ACRU* Menu Converter tool was used to select relevant catchments and simulation period for conversion to *ACRU*2000 input "menu" files
- The *ACRU*2000 version of the *ACRU* model was run from a batch file
- Output data files in dBase (.dbf) format were analysed in Microsoft Excel and *ACRUView* for the statistical analysis.

The AAMG (Figure 5.6) is an ArcGIS extension that enables *ACRU* menus to be configured from pre-existing GIS data. Since data such as landuse, soils, temperature, evaporation and rainfall are readily available for South Africa from Schulze (1997) and Lynch (2004), the AAMG is particularly useful in saving time and effort in configuring a base input dataset for the model. Essentially the AAMG delineates modelling subcatchments based on landuse, and then assigns standard variables that describe water use by the landuse, water movement in the soil profile, temperatures and A-pan evaporation to each of these modelling subcatchments. Rainfall driver stations are assigned to these modelling subcatchments in the GIS. These "de facto" variables are then adjusted using more detailed information where available. Since the AAMG doesn't have knowledge regarding the flow paths of water (yet), linking of modelling catchments is a manual process that is done in the CSV files. The AAMG is particularly useful for configurations with numerous subcatchments such as the Mgeni with 1450 modelling subcatchments. The time taken to set up this configuration is conservatively estimated to have been approximately 30% less when compared to original configurations by Tarboton and Schulze (1992) and later by Kienzle et al (1997).

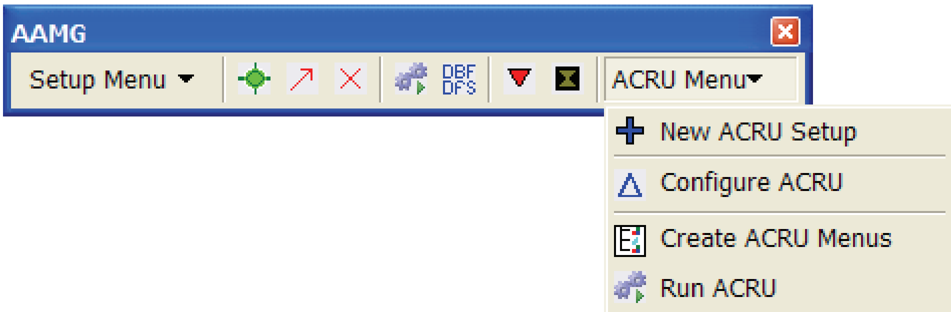


Figure 5.6; The Amalgamation of Agro-Hydrological Modelling Groups application

The *ACRU* Agrohydrological Modelling System (AAHMS) software (Version 1.014, December 2007) developed by the School of BEEH was used to view and edit model input data, generate model input "menu" files and run the *ACRU*2000 version of the *ACRU* model. The Microsoft Excel based *ACRUView* tool developed by the School of BEEH was used to run statistical analyses on the output from the *ACRU* model.

5.2 Requirements of Case Study

The deliverables of the case study task changed considerably from the conception of the project. When compared to the original requirements, the case study has been controlled by the final deliverables to be an assessment of an alpha version of the SPATSIM-HDSF modelling framework requiring further development and testing before being released for general use. Consequently, an assessment is made of the input utilities, ability to run models and time-series analysis utilities in SPATSIM-HDSF, as compared to the tools that would be used for a "conventional" *ACRU* configuration such as those described in Summerton (2008). Although the case study evaluation was performed utilising the *ACRU* model, the evaluation of the SPATSIM-HDSF tools is applicable to the configuration and analysis of any model in SPATSIM-HDSF.

The case study focussed on only new developments in this project, and consequently did not evaluate any of the inherent SPATSIM functionality that has migrated to the new framework. Furthermore, the external GIS tools have not been evaluated as part of the case study as the functionality of the AAMG, which is specific for the *ACRU* model and which was used in the configuration, is similar to the external GIS tools which can be utilised for any model in SPATSIM-HDSF.

5.3 Case Study Findings

This case study is based on an alpha version of the SPATSIM-HDSF framework received in May 2008 and, in the interim, some of the findings have already resulted in improvements to the framework. This alpha version of the framework required expert knowledge from the developers for proper installation to occur, and it is necessary to have the .Net framework and Java Runtime Environment installed. On start up, the user is confronted with a project selection screen which is useful for management of modelling projects and scenarios. To facilitate the case study assessment, 3 main areas are reported on including (1) data input, manipulation and model configuration, (2) ability to run models, and (3) outputs.

5.3.1 Data Input, Manipulation and Model Configuration

On using the SPATSIM-HDSF, several new features are immediately apparent including new attribute data editing forms, the data tabbed page showing a table of attribute values, enhancements to the internal GIS tools and development of a consolidated set of time-series analysis tools. The start-up *Project Selection* form of SPATSIM-HDSF includes options to create a new project or access an existing one. Creating a new project is a simple process where all the required data folders and an empty SPATSIM-HDSF database are built automatically. The main *ACRU* model utilities are accessed through the *Application* / *ACRU Model* menu option on the main SPATSIM-HDSF user interface form. Selection of the *Run Menu Converter* option opens the *ACRU* Menu Converter utility where *ACRU* input "menus" in *ACRU* 300 series version format can be imported into a SPATSIM-HDSF database and an *ACRU* run initiated. Notably, the options to select a range of catchments and a user defined start and end date are very useful functions that did not work well in the AAHMS. The *ACRU* Menu Converter utility provided in the alpha version of SPATSIM-HDSF was only able to convert from *ACRU* menus in *ACRU* 300 series format and needs to be expanded to be able to convert from *ACRU* menus in CSV and *ACRU*2000 text format files. Furthermore, although the *ACRU* Menu Converter utility is able to import time-series data into a SPATSIM-HDSF database, at the time of the case study, it did not appear to be possible to import time-series in *ACRU* Composite format.

The SPATSIM-HDSF database has been structured such that it is suitable for storing data and information required by any model. This has been successfully tested through the development and implementation of *ACRU* input menus using an XML input data file which references *ACRU* model input data and information stored in a SPATSIM-HDSF database. This is a major development since interrogation and interpretation of the original text files was cumbersome. Once data is stored in a SPATSIM-HDSF database, the main benefit of this project becomes apparent. Due to the Relational Database Management Systems (RDBMS) structure of the database, it is now possible to interact and manipulate base data more readily. For example, variables can now be globally changed.

A problem encountered when running the *ACRU* model from SPATSIM–HDSF in the alpha version is that there is an inconsistent error which appears to be related to the interaction between Microsoft Access and the *ACRU* code and is apparent only when reading large datasets from a SPATSIM-HDSF database.

5.3.2 Ability to Run Daily Hydrological Models

A daily time step simulation was successfully completed using the *ACRU* model for the period 1960 to 1990 using the functionality available in the *ACRU* Menu Converter utility provided in SPATSIM-HDSF, using input data from a SPATSIM-HDSF database. Run times comparable to those of the conventional *ACRU* simulation were achieved. Due to current limitations mentioned above, this configuration was for 2 subcatchments of quaternary catchment U20B in the Lions River area. Streamflow at the lower of the 2 catchments compared favourably (Figure 5.7) to that obtained by Summerton (2008) which confirmed the reliability of the *ACRU* Menu Converter.

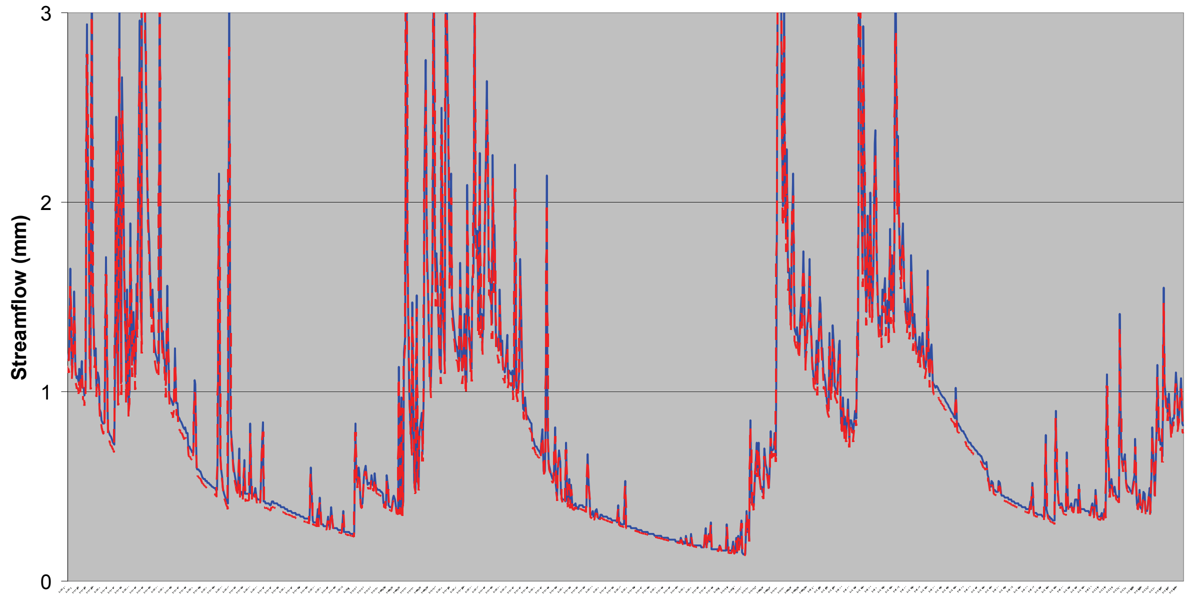


Figure 5.7: Streamflow (CELRUN variable) at Lions River catchment from *ACRU*2000 (solid blue line) (after Summerton, 2008) and SPATSIM–HDSF (dotted red line)

5.3.3 Outputs

The TSOFT and *ACRUView* applications have been successfully consolidated into a set of tools for the analysis and display of time-series data in SPATSIM-HDSF. Although the display of time-series is currently extremely slow, there are adequate tools for the graphing and comparison of time-series data and statistical analyses. Unfortunately, model outputs from the *ACRU* model are not currently accessible to the time-series analysis tools since the *ACRU* model does not currently enable model output to be written back to the SPATSIM-HDSF database, and the time-series analysis tool cannot access the output files from the *ACRU* model which are currently in dBase (.dbf) format. For this evaluation, only the input time-series for the *ACRU* model were accessible to the time-series analysis tools, if they were imported into the SPATSIM-HDSF database using the *ACRU* Menu Converter utility. Consequently, the assessment of this functionality was very limited. The addition of functionality in the *ACRU* Menu Converter utility to enable references to shapefiles to be specified thus providing a link to GIS would be useful. With the ability to assess model outputs, the SPATSIM-HDSF has the potential to be far superior to legacy tools used for the same purpose.

5.4 Recommendations

The development of the SPATSIM-HDSF has resulted in the alpha version of a modelling framework that includes internal and external GIS tools, a generic and extensible data model, and consolidated time-series analysis and graphing tools. In particular, the incorporation of a legacy model (*ACRU*) into SPATSIM-HDSF and several conceptual changes for the running of the *ACRU* model, including the use of the XML format for structuring input data and use of a relational database to store *ACRU* input data and information, were apparent in the case study. A major deliverable that was not originally envisaged has been the conversion of SPATSIM code to the .Net framework. The modelling framework has been improved to be more user friendly, particularly with additions to the attribute editing and viewing functionality. The *ACRU* model has been successfully included as a model that may be run from within the SPATSIM-HDSF framework. However further development and testing is required, including the streamlining of the programming code that will improve stability and the time taken to run various processes. Furthermore, without the inclusion of AAMG type tools and the configuration editor, setting up *ACRU* from scratch will remain a challenging task. It is recommended that functionality be included that will allow the user to:

- import larger configurations, such as the 1450 subcatchments of the Mgeni into a SPATSIM-HDSF database
- store outputs from hydrological models such as *ACRU* to a SPATSIM-HDSF database,
- access *ACRU* output files in dBase (.dbf) format from the time-series analysis tools in SPATSIM-HDSF
- link data to the GIS when importing *ACRU* input "menus" using the *ACRU* Menu Converter tool in SPATSIM-HDSF
- convert other *ACRU* input data formats including *ACRU*2000 and CSV files using the *ACRU* Menu Converter tool
- setup a base configuration using readily available data by accessing AAMG type tools from the SPATSIM-HDSF directly
- access and edit input data and information for the *ACRU* model more easily, by completing and including the *ACRU* Configuration Editor
- perform logic and range checking of data.

Some of these recommendations are beyond the scope of this study, however should be included in future for the SPATSIM–HDSF to truly be of value to practitioners.

6. DISCUSSION, CONCLUSIONS AND RECOMMENDATIONS

Integrated water resources management is a key requirement of the National Water Act (NWA, 1998) of South Africa (Act 36 of 1998) and this has created a requirement for integrated water resources modelling where a set of domain specific models need to be integrated to represent all facets of the hydrological system. The purpose of hydrological modelling frameworks is to provide a modelling environment facilitating the integration of models, providing a common data model and common data editing and analysis tools.

A user requirements survey was conducted prior to the inception workshop to identify general user needs. The framework related responses received indicated the need for a GIS interface for model configuration, a well designed and extensible database, user friendly graphical interfaces, tools for post-processing and display of model output, the ability to incorporate legacy models in the framework and to link and run models in parallel, i.e. on a time step by time step basis. Summarised in Table 2.1, a number of these requirements were either fully or partially addressed in this project.

Several national and international modelling frameworks were reviewed and the SPATSIM modelling framework was selected for further development. A comprehensive framework design was undertaken which included the design of the data model, GIS tools, framework interfaces, data analysis tools, project management tools and help facilities.

This software development undertaken in this project has resulted in a significantly enhanced modelling framework with regard to the database data model, data editing tools, internal GIS tools and time-series analysis tools. The SPATSIM database data model was expanded to include: non-spatial features, attribute groups, storage of metadata; a more comprehensive system of units of measure, storage of relationships between feature objects, storage of time-series data quality flags, storage of attributes and attribute values for processes and the storage of information used by the internal and external GIS tools. The data editing tools include a set of newly designed attribute value editing forms and the new *Data* tabbed page which enables *TextData*, *IntegerData* and *RealData* type attribute values for all the objects in a selected feature and attribute group to be viewed as a data grid. Enhancements to the internal GIS tools include: functionality to setup themes for a spatial feature, use of these themes for rendering based on feature attribute values and displaying these themes in a legend display control. A consolidated set of time-series analysis tools based on the external TSOFT application previously accessed from SPATSIM and the Microsoft Excel based *ACRUView* utility for the *ACRU* model includes graphing, statistics, comparative statistics and frequency analysis tools.

In addition, the conversion of the code to .Net programming platform has ensured that future development of SPATSIM-HDSF can take advantage of recent developments in software development technology. The use of the ADO.Net database access framework should enable SPATSIM-HDSF databases to be implemented in RDBMSs other than Microsoft Access, which was used in this project. An important component of the HDSF is the external GIS tools developed in this project as, in addition to the analysis tools, it includes an interface between ArcGIS and SPATSIM-HDSF which will be extremely useful for model configuration and analysing model output.

The ability to link models in parallel will enable two or more expert models to be integrated without needing to integrate the code for these models. The OpenMI model linking framework (Gijsbers, 2003; Blind and Gregersen, 2004; HarmonIT, 2005a) was developed in the European Commission funded HarmonIT project for this purpose. An initial investigation of OpenMI conducted as part of the HDSF project concluded that the adoption of OpenMI would have many potential benefits. Models running on the same operating platform (e.g. .Net, or Java) may be linked using OpenMI. However, this investigation showed that the interoperability between the .Net platform and the Java platform is difficult, even though technologies are available that may make this possible. This has significant implications for the *ACRU* model which is written in Java. Performance of models linked via OpenMI is another issue that needs to be further investigated.

The *ACRU* model was successfully implemented as a model that may be run within the SPATSIM-HDSF modelling framework. The ability to implement a complex physical conceptual model such as *ACRU* within the SPATSIM-HDSF demonstrates the robustness of the design of the SPATSIM-HDSF modelling framework. As part of this *ACRU* implementation, XML based model input data and model configuration files, and a prototype Configuration Editor for population of model input data files, were developed. These were primarily developed for the *ACRU* model, but have been designed in such a way that they could easily be applied to other hydrological models. During this process new technologies and concepts were learned which will be useful in the further development of the SPATSIM-HDSF modelling framework.

Although process modelling was a secondary objective of the project, significant consolidation and further development of hydrological processes in the *ACRU* model was achieved. These processes include aspects of water management, operational modelling, water quality, irrigation efficiency and crop water use, all of which are important components in the implementation of the National Water Act (NWA, 1998).

The dam and river operating rule module developed for the *ACRU* model by Butler (2001) was an important step towards providing the modelling capability in the *ACRU* model for modelling the Ecological Reserve on a daily basis. Therefore the consolidation of the Operating Rule module within the most recent version of the *ACRU* model was important. However, several limitations were recognised and an investigation was conducted into the feasibility of extending the Operating Rule module to facilitate multiple water sources per water user, account for flow lags and attenuations when supplying water, improve modelling of the Ecological Reserve and be suitable for use as an operation tool for management of water resources. This investigation included a review of the techniques used in the WRYM, Mike Basin and a model developed by Mallory and Van Vuuren (2007) and concluded that a rule-based system should be used and that it would be feasible to implement such a system in the *ACRU* model.

Implementation of the National Water Act (NWA, 1998) requires that water quality also be taken into account in the management of water resources. The *ACRU* model includes the *ACRU_NP* nitrogen and phosphorous module and as part of this project the *ACRUSalinity* module developed for *ACRU* by Teweldebrhan (2003) was consolidated within the most recent version of the *ACRU* model. The consolidated *ACRUSalinity* module was then used in WRC Project K5/1301 and, at the request of this project, several enhancements were made to the *ACRUSalinity* module. These include improved modelling of salts at the soil surface and the implementation of rules enabling the abstraction of water from dam and river water sources to be modelled in an improved operational manner based on water availability and salinity levels.

The *ACRUCane* module developed by Moult (2005) was also consolidated within the most recent version of the *ACRU* model. Irrigation is one of the dominant users of water resources in South Africa and the *ACRUCane* module is an important step towards being able to model in more detail the link between hydrology, irrigation scheduling, irrigation efficiency and crop yield. The *ACRUCane* module is based on the FAO56 method and although focussed on sugarcane, could be easily extended to model other irrigated crops.

At the conclusion of this project both the SPATSIM-HDSF and the *ACRU* software are at a stage where significant development has taken place and new designs and concepts have been proven. However, both need to be thoroughly beta tested before being released for general use. It is anticipated that as the new version of SPATSIM-HDSF is used there will be requests and suggestions from users for further development of additional functionality, such as time-series analysis and GIS related tools.

Some recommendations for possible further development of the SPATSIM-HDSF modelling framework include:

- further development of the external GIS tools to enable catchment delineation and the setting up of linkages/relationships between features
- further development of the internal GIS tools to enable the setting up of linkages/relationships between features and rendering based on array attribute elements
- improvement of the speed of display of time-series data in the attribute data editing forms
- development of additional time-series analysis tools including: extreme value analysis, run length analysis and time-series transformation tools
- modification of time-series analysis tools so that graph legends are displayed on a separate control that the user can choose to hide or display
- providing an option to access time-series files in dBase (.dbf) format from the time-series analysis tools
- development of tools to import data from one SPATSIM-HDSF database to another SPATSIM-HDSF database
- development of tools to import and export other data formats
- creation of an OpenMI wrapper around the SPATSIM-HDSF database.

Some recommendations for possible further development of the *ACRU* model and related tools for the SPATSIM-HDSF modelling framework include:

- storage of outputs from the *ACRU* model to the SPATSIM-HDSF database
- linking data to the GIS when importing *ACRU* input "menus" using the *ACRU* Menu Converter tool in SPATSIM-HDSF
- conversion of other *ACRU* input data formats including *ACRU*2000 and CSV files using the *ACRU* Menu Converter tool
- development of a tool to enable *ACRU* users to select variables to be included in the model output, where these variables are to be stored and in which format
- development of tools to setup a base configuration using readily available data by accessing GIS type tools from the SPATSIM-HDSF directly
- development of a full working version of the *ACRU* Configuration Editor
- implementation of OpenMI in *ACRU*.

An important outcome of this project has been the collaboration of two key hydrological modelling groups in South Africa, namely the School of Bioresources Engineering and Environmental Hydrology at the University of KwaZulu-Natal and the Institute for Water

Research at Rhodes University. The collaboration between these modelling groups will impact positively on the sustainability of detailed hydrological modelling in South Africa.

7. REFERENCES

Acocks, JPH. 1988. Veld Types of Southern Africa. Botanical Research Institute, Pretoria, RSA. *Botanical Survey of South Africa Memoirs*, 57.

Ahuja, LR, David, O and Ascough, JC II. 2004. Developing natural resource models using the object modeling system: feasibility and challenges. Transactions of the 2nd Biennial Meeting of the International Environmental Modelling and Software Society, iEMSs 2004: Osnabrück, Germany.

Allen, RG, Pereira, LS, Raes, D and Smith, M. 1998. Guidelines for computing crop water requirements. Irrigation and Drainage Paper No. 56. FAO, Rome, Italy.

Argent, RM and Rizzoli, AE. 2004. Development of Multi-Framework Model Components. Transactions of the 2nd Biennial Meeting of the International Environmental Modelling and Software Society, iEMSs 2004: Osnabrück, Germany.

Blind, M and Gregersen, JB. 2004. Towards an Open Modelling Interface (OpenMI): The HarmonIT Project. Transactions of the 2nd Biennial Meeting of the International Environmental Modelling and Software Society, iEMSs 2004. Osnabrück, Germany.

Bristow, KL and Campbell, GS. 1984. On the relationship between incoming solar radiation and daily maximum and minimum temperature. *Agricultural and Forest Meteorology*, 31, 159-166.

Bromley, KA. 1989. Land Cover Classes in the Mgeni River Catchment. Supplement to CSIR Report AAL 21. Institute of Natural Resources, University of Natal, Pietermaritzburg, RSA.

Butler, AJE. 2001. The development and evaluation of an operating rule framework to satisfy the requirements of the National Water Act. Unpublished MSc Engineering dissertation. Department of Agricultural Engineering, University of Natal, Pietermaritzburg, RSA. 146 pp.

Butler, AJE, Smithers, JC, Jewitt, GPJ and Clark, DJ. 2001. The development of an operating rule framework to satisfy the requirements of the National Water Act. Proceedings of Tenth South African National Hydrology Symposium: 26 - 28 September, 2001. School of BEEH, University of Natal, Pietermaritzburg, RSA.

Catchment Modelling Toolkit. 2005. Catchment modelling toolkit. [Internet]. Catchment modelling toolkit. Available on webpage: http://www.toolkit.net.au [Accessed: 14 February 2005].

Chapman, RD. 2004. Estimation of Vapour Pressure and Solar Radiation in South Africa. MSc dissertation, School of Bioresources Engineering and Environmental Hydrology, University of KwaZulu-Natal, Pietermaritzburg, RSA.

Clark, DJ and Smithers, JC. 2006. Hydrological Decision Support Framework (HDSF) - Initial Design. Water SA, Vol. 32, No.4, October 2006, 465-472. Water Research Commission, Pretoria, South Africa.

CPH Water. 2006. Mike Basin Training Examples. WRC project K5/1300. Water Research Commission, Pretoria, RSA.

CPH Water. 2007. Technology Transfer and Integrated Implementation of Water Management Models in Commercial Farming. WRC project K5/1481/4. Water Research Commission, Pretoria, RSA.

David, O, Schneider, IW and Leavesley, GH. 2004. Metadata and modeling frameworks: The object modeling system example. Transactions of the 2nd Biennial Meeting of the International Environmental Modelling and Software Society, iEMSs 2004: Osnabrück, Germany.

De Jager, FGB and Van Rooyen, PG. 2007. Water Resources Yield Model User Guide – Release 7.1. Internal report. Department of Water Affairs and Forestry, Pretoria, RSA.

DWAF. 2004a. National Water Resources Strategy. Department of Water Affairs and Forestry, Pretoria, RSA. First Edition – September 2004.

DWAF. 2004b. Vaal Hydrological Information Management System – Software Design Document. The Strategic Planning Systems Sub-Directorate, Department of Water Affairs and Forestry (DWAF), Pretoria, South Africa.

ESRI. 2005. Environmental Systems Research Institute (ESRI) – GIS and Mapping Software webpage: http://www.esri.com [Accessed: 22 March, 2005].

Gijsbers, PJA. 2003. OpenMI – Harmonizing linkages between water related models. [Internet] Presented at Integrated Modelling User Group (IMUG) 2003 - International Conference on Application of Integrated Modelling, Tillburg Netherlands. Available from: http://www.wrcplc.co.uk/imug/html/imug_2003.htm [Accessed: 12 March, 2005].

Hargreaves, GH and Samani, ZA. 1985. Reference Crop Evapotranspiration From Temperature. *Transactions of the American Society of Agricultural Engineers*, 1, 96-99.

HarmonIT. 2002. State of the Art Review. Document version: 9. Hutchings C. (ed.) [Internet]. Document produced for the HarmonIT project and available to members on registration on the HarmonIT website: www.harmonit.org [Accessed: 26 January 2004].

HarmonIT. 2005a. The OpenMI Architecture - Report A: Scope for the OpenMI. Document version: 1.0. Editor: Tindall I. [Internet]. Document produced for the HarmonIT project and available to members on registration on the HarmonIT website: www.harmonit.org [Accessed: 20 September 2005]

HarmonIT. 2005b. The OpenMI Document Series Part B – Guidelines for OpenMI. Document version: 1.0. Editor: Tindall I. [Internet]. Document produced for the HarmonIT project and available to registered members on the HarmonIT website: www.harmonit.org [Accessed: 15 July 2005]

HarmonIT. 2005c. The OpenMI Document Series Part C – The org.OpenMI.Standard interface specification. Document version: 1.0. Editor: Gijsbers P J A. [Internet]. Document produced for the HarmonIT project and available to registered members on the HarmonIT website: www.harmonit.org [Accessed: 15 July 2005]

Havnø, K, Sørensen, HR and Gregersen, JB. 2002. Integrated water resources modelling and object oriented code architecture. Proceedings of the 1st Asia-Pacific DHI Software Conference, Bangkok, Thailand.

Hughes, DA. 2002. The development of an information modelling system for regional water resource assessments. Proceedings of the 4th International FRIEND Conference. IAHS Publication 274.2002, 43-49.

Hughes, DA (Ed). 2005. SPATSIM, an Integrating Framework for Ecological Reserve Determination and Implementation: Incorporating Water Quality and Quantity Components for Rivers. WRC Report TT 245/04, Water Research Commission, Pretoria, South Africa.

Hughes, DM, Mallory, SJL and Louw, D. 2007. Methods and Software for the Real-time Implementation of the Ecological Reserve – Explanations and User Manual. WRC project K5/1587. Water Research Commission, Pretoria, RSA.

ISO. 2004. ISO/TC 211/WG 4/PT 19136 Geographic information – Geography Markup Language (GML). GML Version 3.1.0 [Internet]. Available from: http://www.opengis.net/gml/ [Accessed: 21 March 2005].

IWR. 2005. SPATSIM - Spatial and Time series Information Modelling Software. [Internet] Institute for Water Research (IWR), Rhodes University, Grahamstown, 6140. Available from: http://www.rhodes.ac.za/institutes/iwr/software/spatsim.html [Accessed: 23 February, 2005].

Kienzle, SW, Lorentz, SA and Schulze RE. 1997. Hydrology and water quality of the Mgeni Catchment. Water Research Commission, Pretoria, South Africa, Report TT87/97.

Kiker, GA. 2001. *ACRU*2000 Model. In: Lynch, SD and Kiker, GA. *ACRU* Model Development and User Support. Water Research Commission, Pretoria, South Africa, Report 636/1/00.

Kiker, GA, Clark, DJ, Martinez, CJ. and Schulze, RE. 2006. A Java-based, Object-Oriented Modeling System for Southern African Hydrology. Transactions of the ASABE, 49(5): 1419 - 1433.

Kisters. 2005. Hydstra – Engineering and Environmental Data Management Solutions webpage: http://www.hydstra.com/english [Accessed: 21 March, 2005].

Kralisch, S, Krause, P and David, O. 2004. Using the object modeling system for hydrological model development and application. Transactions of the 2nd Biennial Meeting of the International Environmental Modelling and Software Society, iEMSs 2004, Osnabrück, Germany.

Lecler, N, Clark, DJ, Schulze, RE, Griffiths, BAK and Smithers, JC. 2007. Performance of irrigation systems and water user of sugarcane: A field to catchment scale perspective. Paper presented at XXVI Conference of ISSCT. International Society of Sugar Cane Technologists, Durban, South Africa.

Lecler, NL, Moult, NG and Smithers, JC. 2006. Application of a Catchment Scale Irrigation Systems Model. Proceedings 80th SASTA Conference. South African Sugar Association, Mount Edgecombe, South Africa, Mount Edgecombe, SA.

Lynch, SD. 2004. *Development of a Raster Database of Annual, Monthly and Daily Rainfall for Southern Africa*. Water Research Commission, Pretoria, RSA, WRC Report 1156/1/04. 78pp.

Lynch, SD and Schulze, RE. 2007. Rainfall Database. In: Schulze, RE (Ed). 2007. *South African Atlas of Climatology and Agrohydrology*. Water Research Commission, Pretoria, RSA, WRC Report 1489/1/06, Section 2.2.

Maidment, D (Ed). 2002. Arc Hydro - GIS for Water Resources. ESRI Press, Redlands, California, USA.

Mallory, SJL and Van Vuuren, SJ. 2007. Water Resource Modelling of Complex Catchments: The Development of a Rule-based Modelling Technique for South African Conditions. Unpublished paper. Department of Civil Engineering, University of Pretoria, RSA.

Marston, F, Argent, R, Vertessy, R, Cuddy, S and Rahman, J. 2002. The Status of Catchment Modelling in Australia. Technical Report, Report 02/4. Cooperative Research Centre for Catchment Hydrology (CRCCH), Australia. Available from: http://www.toolkit.net.au [Accessed: 2 February 2005].

Meier, KB. 1997. *Development of a Spatial Database for Agrohydrological Model Applications in Southern Africa*. Unpublished MSc dissertation, Department of Agricultural Engineering, University of Natal, Pietermaritzburg, RSA.

Moult, NG. 2005. The development of a catchment scale irrigation systems model for sugarcane. MSc Eng. Dissertation. School of Bioresources Engineering and Environmental Hydrology, School of Bioresources Engineering and Environmental Hydrology, University of KwaZulu-Natal, Pietermartizburg, South Africa.

Moult, N, Lecler, N, Bezuidenhout, C and Clark, D. 2005a. A Catchment Scale Irrigation Systems Model for Sugarcane. Proceedings 79th SASTA Conference. South African Sugar Association, Mount Edgecombe, RSA, Mount Edgecombe, RSA. 452-455.

Moult, NG, Lecler, NL and Smithers, JC. 2006a. A Catchment Scale Irrigation Systems Model For Sugarcane II: Model Application. The Changing Face of Irrigation in South Africa. South African National Committee on Irrigation and Drainage, Pretoria, RSA, Swadini, RSA.

Moult, NG, Lecler, NL and Smithers, JC. 2007a. A Catchment Scale Irrigation Systems Model For Sugarcane II: Model Application. Submitted to Water SA.

Moult, N, Lecler, N, Smithers, J, Bezuidenhout, C and Clark, D. 2005b. A Catchment Scale Irrigation Model for Sugarcane. Proceedings of the Twelfth South African National Hydrological Symposium. SANCIAHS, Pretoria, RSA, Johannesburg, RSA.

Moult, NG, Lecler, NL, Smithers, JC and Clark, DJ. 2006b. A Catchment Scale Irrigation Systems Model For Sugarcane I: Model Development. The Changing Face of Irrigation in South Africa. South African National Committee on Irrigation and Drainage, Pretoria, RSA, Swadini, RSA

Moult, NG, Lecler, NL, Smithers, JC and Clark, DJ. 2007b. A Catchment Scale Irrigation Systems Model For Sugarcane I: Model Development. Submitted to Water SA.

Murray, N, Perraud, J-M, Rahman, J, Seaton, S, Hotham, H and Watson F. 2004. Introduction to TIME. Workshop notes. [Internet] Cooperative Research Centre for Catchment Hydrology (CRCCH), Australia. Available from: http://www.toolkit.net.au/time [Accessed: 2 February 2005].

NWA. 1998. National Water Act, Act No. 36 of 1998. Government Printer, Pretoria, RSA.

OMS. 2005. OMS – Central Collaboration Platform. [Internet]. OMS webpage: http://oms.ars.usda.gov [Accessed: 12 March 2005].

Pott, AJ, Van der Merwe, RJ, Meier, KB and Summerton, MJ. 2007. GIS Integrated linking of the *ACRU* and Mike Basin model. In: Proceedings of the Thirteenth South African National Hydrological Symposium. Cape Town, South Africa.

Rahman, JM, Seaton, SP and Cuddy, SM. 2002. Making frameworks more useable: Using model introspection and metadata to develop model processing tools. Transactions of the 1st Biennial Meeting of the International Environmental Modelling and Software Society, iEMSs 2002, Manno, Switzerland.

Rahman, JM, Seaton, SP, Perraud, J-M, Hotham, H, Verrelli, DI and Coleman, JR. 2003. It's TIME for a New Environmental Modelling Framework. In: Post D A (Ed), MODSIM 2003 International Congress on Modelling and Simulation: Townsville, Modelling and Simulation Society of Australia and New Zealand inc., pg 1727 – 1732.

SABS. 1973. The International Metric System (SI): Guide to the Use of the SI in South Africa. South African Bureau of Standards, Private Bag X191, Pretoria, South Africa.

Schulze, RE. 1995. *Hydrology and Agrohydrology: A Text to Accompany the ACRU 3.00 Agrohydrological Modelling System*. Water Research Commission, Pretoria, RSA, Report TT 69/9/95. pp 552.

Schulze, RE. 1997. *South African Atlas of Agrohydrology and -Climatolgy*. Water Research Commission, Pretoria, RSA, Report TT82/96. 276pp.

Schulze RE (Ed). 2002. Modelling as a Tool in Integrated Water Resources Management: Conceptual Issues and Case Study Applications. WRC Report No 749/1/04. Water Research Commission, Pretoria, RSA.

Schulze, RE. 2004. Determination of Baseline Land Cover Variables for Applications in Assessing Land Use Impacts on Hydrological Responses in South Africa. *In*: Schulze, RE and Pike, A. *Development and Evaluation of an Installed Hydrological Modelling*

System. Water Research Commission, Pretoria, RSA, WRC Report 1155/1/04. Chapter 2, 37-50.

Schulze, RE (Ed). 2007a. *South African Atlas of Climatology and Agrohydrology*. Water Research Commission, Pretoria, RSA, WRC Report 1489/1/06.

Schulze, RE. 2007b. Soils: Agrohydrological Information Needs, Information Sources and Decision Support. In: Schulze, RE (Ed), *South African Atlas of Climatology and Agrohydrology,* Section 4.1. WRC Report 1489/1/06, Water Research Commission, Pretoria, RSA.

Schulze, RE and Chapman, RD. 2007. Estimation of Daily Solar Radiation over South Africa. *In:* Schulze, RE (Ed). 2007. *South African Atlas of Climatology and Agrohydrology*. Water Research Commission, Pretoria, RSA, WRC Report 1489/1/06, Section 5.2.

Schulze, RE, Chapman, RD and Maharaj, M. 2007a. Daily Minimum, Maximum and Mean Relative Humidity. *In:* Schulze, RE (Ed). 2007. *South African Atlas of Climatology and Agrohydrology*. Water Research Commission, Pretoria, RSA, WRC Report 1489/1/06, Section 12.2.

Schulze, RE, Hallowes, LA, Horan, MJC, Lumsden, TG, Pike, A, Thornton-Dibb, S and Warburton, ML. 2007b. South African Quaternary Catchments Database. *In:* Schulze, RE (Ed). 2007. *South African Atlas of Climatology and Agrohydrology*. Water Research Commission, Pretoria, RSA, WRC Report 1489/1/06, Section 2.3.

Schulze, RE, Lorentz, SA, Kienzle, SW and Perks, L. 2004. Case Study 3: Modelling the Impacts Of Land Use And Climate Change On Hydrological Response In The Mixed, Underdeveloped/Developed Mgeni Catchment, Southern Africa. In: Kabat, P, Claussen, M, Dirmeyer, PA, Gash, JHC, Bravo de Guenni, L, Meybeck, M, Pielke, Sr., RA, Vörösmarty, CJ, Hutjes, RWA. and Lütkenmeier, S (eds.), *Vegetation, Water, Humans and the Climate: A New Perspective on an Interactive System*. Springer, Heidelberg, Germany.

Schulze, RE and Maharaj, M. 2004. *Development of a Database of Gridded Daily Temperatures for Southern Africa*. Water Research Commission, Pretoria, RSA, WRC Report 1156/2/04.

Schulze, RE and Maharaj, M. 2007. Temperature Database. *In:* Schulze, RE (Ed). 2007. *South African Atlas of Climatology and Agrohydrology*. Water Research Commission, Pretoria, RSA, WRC Report 1489/1/06, Section 2.1.

Schulze, RE, Maharaj, M and Moult, N. 2007c. Reference Crop Evaporation by the Penman-Monteith Method. *In:* Schulze, RE (Ed). 2007. *South African Atlas of Climatology and Agrohydrology*. Water Research Commission, Pretoria, RSA, WRC Report 1489/1/06, Section 13.3.

Schulze, RE, Warburton, M, Lumsden, TG and Horan, MJC. 2005. The Southern African Quaternary Catchments Database: Refinements to, and Links with, the *ACRU* System as a Framework for Modelling Impacts of Climate Change on Water Resources. In: Schulze, RE (Ed), *Climate Change and Water Resources in Southern Africa: Studies on Scenarios, Impacts, Vulnerabilities and Adaptation*. Water Research Commission, Pretoria, RSA, WRC Report 1430/1/05. Chapter 8, 111 - 139.

Sharpley, AN, Ahuja, LR, Yamamoto, M and Menzel, RG. 1981. The kinetics of phosphorus desorption from soils. Soil Science Society of America Journal, 45:493-496.

Singels, A and Bezuidenhout, CN. 2002. A new method of simulating dry matter partitioning in the Canegro sugarcane model. *Field Crops Research* 78:151-164.

Singels, A, Bezuidenhout, CN and Schmidt, EJ. 1999. Evaluating strategies for scheduling supplementary irrigation of sugarcane in South Africa. In: *Proceedings of the Australian Society for Sugar Cane Technology*, 21:219-226. ASSCT, Brisbane, Australia.

Smithers, JC, Butler, AJE, Jewitt, GPJ, Clark, DC and Schulze, RE. 2005. The development of a dam operating rule framework for the *ACRU* model to satisfy the requirements of the national water act in South Africa. Enhancing equitable livelihood benefits of dams using decision support systems. CGIAR Challenge Programme on Water and Food, IWMI, Addis Abbaba, Ethiopia, Nazareth, Ethiopia.

Smithers, JC and Schulze, RE. 2004. *ACRU Agrohydrological Modelling System*: *User Manual Version 4.00*. School of Bioresources Engineering and Environmental Hydrology, University of KwaZulu-Natal, Pietermaritzburg, RSA.

Summerton, MJ. 1995. Process and modelling studies in forest hydrology. Unpublished MSc dissertation, Department of Agricultural Engineering, University of Natal, Pietermaritzburg, RSA. 153pp.

Summerton, MJ. 2008. A Preliminary Assessment of the Impact of Climate Change on the Water Resources of the Mgeni Catchment. Umgeni Water, Planning Services, Pietermaritzburg, RSA, Report Number 160.8/R001/2008 (DRAFT).

Summerton, MJ and Gillham, SW. 2007. A Water Utility's approach to addressing the potential impacts of Climate Change. In: Proceedings of the Thirteenth South African National Hydrological Symposium. Cape Town, South Africa.

Tarboton, KC and Schulze, RE. 1992. Distributed Hydrological modelling system for the Mgeni catchment. Water Research Commission, Pretoria, RSA, WRC report 234/1/92.

Teweldebrhan, AT. 2003. The Hydrosalinity Module of *ACRU* Agrohydrological Modelling System (*ACRUSalinity*): Module Development and Evaluation. Unpublished MSc Dissertation, School of Bioresources Engineering and Environmental Hydrology, University of Natal, Pietermaritzburg, RSA.

Teweldebrhan, AT, Lorentz, SA and Schulze, RE. 2003. The Hydrosalinity Module of *ACRU* Agrohydrological Modelling System (*ACRUSalinity*). *ACRU*cons Report, 47. School of Bioresources Engineering and Environmental Hydrology, University of Natal, Pietermaritzburg, RSA.

USEPA. 2001. BASINS – Better Assessment Science Integrating Point and Nonpoint Sources (Version 3.0). User Manual. Office of Water, United States Environmental Protection Agency (USEPA), 401 Mth Street SW, Washington, DC 20460, USA. Available from: http://www.epa.gov/waterscience/basins/bsnsdocs.html [Accessed: 25 February, 2005].

Van Rooyen, PG, De Jager, FCB and Swart, HS. 2003. Water Resources Yield Model: Procedural Manual. Internal report. Department of Water Affairs and Forestry, Pretoria, RSA.

WL Delft Hydraulics. 2005a. HYMOS - Information System for Water and Environment webpage: http://www.wldelft.nl/soft/hymos/int/index.html [Accessed: 21 March, 2005].

WL Delft Hydraulics. 2005b. HYMOS – Hydrological Database System. Presentation. [Internet]. Available from: http://www.wldelft.nl/soft/hymos/doc/index.html [Accessed: 21 March, 2005].

Zeiler, M. 1999. Modeling Our World - The ESRI Guide to Geodatabase Design. Environmental Systems Research Institute, Inc. Redlands, USA.

8. APPENDICES

There are four sets of appendices associated with this report. Appendix A is an integral part of this report and includes details of SPATSIM-HDSF database design specifications, units of measure added to SPATSIM-HDSF and examples of Model Configuration Editor files. Appendix B is mostly a collection of reports for deliverables submitted as part of this project. The reports in Appendix B include details about the user requirement survey, framework design, framework development and modelling that may not have been reported in detail in the main body of this report. Appendix C contains the datasets packaged as part of this project to support users of the SPATSIM-HDSF modelling framework, as described in Chapter 4 of this report. Appendix D contains the alpha version of the SPATSIM-HDSF modelling framework developed in this project and includes the models that may currently be run from SPATSIM_HDSF. Appendix B, Appendix C and Appendix D are not integral parts of this report and have been included in electronic form on the DVD accompanying this report.

Appendix A

A.1 Specifications for the SPATSIM-HDSF Data Model

Table A.1.1: Description of data types

Data Type	Description
COUNTER	Auto-incrementing 32-bit integer counter field
SMALLINT	16-bit integer field
INTEGER	32-bit integer field
DOUBLE	Double precision floating-point number field (15 digit precision)
VARCHAR(n)	String field. Maximum size of n characters where n < 255
LONGCHAR	String field up to 2.0 Gigabytes
LONGBINARY	Binary field up to 2.0 Gigabytes
DATETIME	Date and time field
BOOLEAN	Boolean field (1 bit)

Feature Reference Table: *data_dict1*

Table A.1.2: Description of the fields in the *data_dict1* table (after Hughes, 2005; IWR, 2005)

Field	Description
FeatureCode	A unique ID number associated with a single feature (shape file).
FeatureName	The name of the feature for display and selection purposes.
FeatureSource	The path and filename of an ESRI shape file, image or the name of a *NonSpatialFeatureTable* table.
IDField	The field name in the shape file or *NonSpatialFeatureTable* table that uniquely identifies an object (integer data).
DescField	The field name in the shape file or *NonSpatialFeatureTable* table that is selected to describe the objects (text or number data).
FeatureType	An integer number identifying the type of feature: 1 = Spatial, 2 = Image, 3 = Non-spatial, 4 = GRID.
Metadata	A memo field in which metadata describing the feature may be recorded.

Table A.1.3: Specifications for the *data_dict1* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
FeatureCode	COUNTER	✓	✓			
FeatureName	VARCHAR (40)		✓			
FeatureSource	VARCHAR (100)		✓			
IDField	VARCHAR (40)		✓			
DescField	VARCHAR (40)		✓			
FeatureType	SMALLINT		✓			
Metadata	LONGCHAR					

Feature Table: *NonSpatialFeatureTable*

Table A.1.4: Description of the fields in a *NonSpatialFeatureTable* table

Field	Description
ObjectID	*A unique integer ID used to identify the non-spatial object.*
ObjectDesc	A unique text name describing the non-spatial object.

Table A.1.5: Specifications for a *NonSpatialFeature* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
ObjectID	COUNTER	✓	✓			
ObjectDesc	VARCHAR (80)		✓			

Feature Reference Table: *data_dict2*

Table A.1.6: Description of the fields in the *data_dict2* table

Field	Description
FeatureCode	A unique ID number associated with a single feature. References the *FeatureCode* field in the *data_dict1* table.
GroupCode	A unique ID number associated with an attribute group.
GroupName	The name of an attribute group.

Table A.1.7: Specifications for the *data_dict2* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
FeatureCode	INTEGER	✓	✓			
GroupCode	COUNTER	✓	✓			
GroupName	VARCHAR (30)		✓			

Feature Reference Table: *data_dict3*

Table A.1.8: Description of the fields in the *data_dict3* table (after Hughes, 2005; IWR, 2005)

Field	Description
GroupCode	A unique ID number associated with an attribute group. References the *GroupCode* field in the *data_dict2* table.
AttributeCode	A unique ID number associated with the attribute.
AttributeName	The name of the attribute associated with the feature.
DataType	An integer number identifying the type of data associated with the attribute that also defines the type of attribute data table that the attribute data will be stored in. The data types are: 0 = Text, 1 = Integer, 2 = Real, 3 = Time-series, 4 = Picture, 5 = Array, 6 = Memo, 7 = Link.

UnitID	A unique ID number associated with the unit of measure. References the *UnitID* field in the *UnitsOfMeasure* table.
Metadata	A memo field in which metadata describing the attribute may be recorded.

Table A.1.9: Specifications for the *data_dict3* table

Field	**Type**	**Primary Key**	**Required**	**Maximum Value**	**Minimum Value**	**Default Value**
GroupCode	INTEGER	✓	✓			
AttributeCode	COUNTER	✓	✓			
AttributeName	VARCHAR (30)		✓			
DataType	SMALLINT		✓			
UnitID	INTEGER		✓			
Metadata	LONGCHAR					

Feature Reference Table: *data_dict4*

Table A.1.10: Description of the fields in the *data_dict4* table (after Hughes, 2005; IWR, 2005)

Field	**Description**
AttributeCode	A unique ID number associated with the attribute. References the *AttributeCode* field in the *data_dict3* table.
ObjectRecord	A unique ID number associated with an object of the relevant feature. References the Object*ID* field in the relevant feature's *ShapefileDbfTable* or *NonSpatialFeatureTable* table.
TableRecord	A unique ID number associated with the attribute data for the attribute. References the *RECID* field of the attribute data table associated with the attribute.

Table A.1.11: Specifications for the *data_dict4* table

Field	**Type**	**Primary Key**	**Required**	**Maximum Value**	**Minimum Value**	**Default Value**
AttributeCode	INTEGER	✓	✓			
ObjectRecord	INTEGER	✓	✓			
TableRecord	INTEGER	✓	✓			

Attribute Value Metadata Table: *AttValMeta*

Table A.1.12: Description of the fields in the *AttValMeta* table

Field	**Description**
TableCode	A unique ID number associated with each attribute data table. References the *TableCode* field in the *data_dict5* table.
TableRecord	A unique ID number associated with the attribute data for the attribute. References the *RECID* field of the attribute data table associated with each attribute.
Metadata	A memo field in which metadata describing an attribute value may be recorded.

Table A.1.13: Specifications for the *AttValMeta* table

Field	**Type**	**Primary Key**	**Required**	**Maximum Value**	**Minimum Value**	**Default Value**
DataType	SMALLINT	✓	✓			
TableRecord	INTEGER	✓	✓			
Metadata	LONGCHAR					

Process Reference Table: *proc_run*

Table A.1.14: Description of the fields in the *proc_run* table

Field	Description
PRCode	A unique ID number associated with each model application.
Description	A description of the model application.
RecDate	The date on which the model application was created or last edited.
EXE	The name of the model's executable file.
EXE_Version	The version number of the model.
No_Objects	The number of feature objects included in the modelling application.
No_Attributes	The number of feature attributes included in the modelling application.
Object_List	A list of the objects (and the features they belong to) included in the modelling application.
Attribute_List	A list of the attributes (and the features they belong to) included in the modelling application.
Object_Links1	A list of object links.
Object_Links2	A second list of object links.
StartDate	The simulation start date for the model application.
EndDate	The simulation end date for the model application.
Metadata	A memo field in which metadata describing a model application may be recorded.

Table A.1.15: Specifications for the *proc_run* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
PRCode	COUNTER	✓	✓			
Description	VARCHAR (40)		✓			
RecDate	DATETIME		✓			
EXE	VARCHAR (40)		✓			
EXE_Version	VARCHAR (30)		✓			
No_Objects	SMALLINT		✓			
No_Attributes	SMALLINT		✓			
Object_List	LONGBINARY					
Attribute_List	LONGBINARY					
Object_Links1	LONGBINARY					
Object_Links1	LONGBINARY					
StartDate	DATETIME					
EndDate	DATETIME					
Metadata	LONGCHAR					

Process Reference Table: *data_dict_pr2*

Table A.1.16: Description of the fields in the *data_dict_pr3* table

Field	Description
PRCode	A unique ID number associated with a model application. References the *PRCode* field in the *proc_run* table.
GroupCode	A unique ID number associated with the attribute group.
GroupName	The name of the attribute group associated with a model application.

Table A.1.17: Specifications for the *data_dict_pr2* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
PRCode	INTEGER	✓	✓			
GroupCode	COUNTER	✓	✓			
GroupName	VARCHAR (30)		✓			

Process Reference Table: *data_dict_pr3*

Table A.1.18: Description of the fields in the *data_dict_pr3* table

Field	Description
GroupCode	A unique ID number associated with an attribute group. References the *GroupCode* field in the *data_dict_pr3* table.
AttributeCode	A unique ID number associated with the attribute.
AttributeName	The name of the attribute associated with a model application.
DataType	An integer number identifying the type of data associated with the attribute that also defines the type of attribute data table that the attribute data will be stored in. The data types are: 0 = Text, 1 = Integer, 2 = Real, 3 = Time-series, 4 = Bitmap (Graphic), 5 = Array, 6 = Memo, 7 = Link.
UnitID	A unique ID number associated with the unit of measure. References the *UnitID* field in the *UnitsOfMeasure* table.
Metadata	A memo field in which metadata describing the attribute may be recorded.

Table A.1.19: Specifications for the *data_dict_pr3* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
GroupCode	INTEGER	✓	✓			
AttributeCode	COUNTER	✓	✓			
AttributeName	VARCHAR (30)		✓			
Datatype	SMALLINT		✓			
UnitID	INTEGER		✓			
Metadata	LONGCHAR					

Process Reference Table: *data_dict_pr4*

Table A.1.20: Description of the fields in the *data_dict_pr4* table

Field	Description
AttributeCode	A unique ID number associated with the attribute. References the *AttributeCode* field in the *data_dict_pr3* table.
TableRecord	A unique ID number associated with the attribute data for the attribute. References the *RECID* field of the attribute data table associated with the attribute.

Table A.1.21: Specifications for the *data_dict_pr4* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
AttributeCode	INTEGER	✓	✓			
TableRecord	INTEGER	✓	✓			

Relationship Table: *obj_rel*

Table A.1.22: Description of the fields in the *obj_rel* table

Field	Description
RelationshipID	A unique ID number associated with the relationship.
FeatureA_ID	A unique ID number associated with the feature containing object A of the relationship. References the *FeatureCode* field in the *data_dict1* table.
ObjectA_ID	A unique ID number associated with object A of the relationship. References the *ObjectID* field in the relevant feature's *ShapefileDbfTable* table.
FeatureB_ID	A unique ID number associated with the feature containing object B of the relationship. References the *FeatureCode* field in the *data_dict1* table.
ObjectB_ID	A unique ID number associated with object B of the relationship. References the *ObjectID* field in the relevant feature's *ShapefileDbfTable* table.
RelationshipTypeID	A unique ID number associated with the relationship type. References the *RelationshipTypeID* field in the *obj_rel_type* table.
Proportion	A proportion value that can be applied to the relationship between object A and object B where object A has the same type of relationship with two or more other objects.

Table A.1.23: Specifications for the o*bj_rel* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
RelationshipID	COUNTER	✓	✓			
FeatureA_ID	INTEGER					
ObjectA_ID	INTEGER					
FeatureB_ID	INTEGER					
ObjectB_ID	INTEGER					
RelationshipTypeID	INTEGER					
Proportion	DOUBLE					1.0

Relationship Table: *obj_rel_type*

Table A.1.24: Description of the fields in the *obj_rel_type* table

Field	Description
RelationshipTypeID	A unique ID number associated with the relationship type.
RelationshipName	The name of the relationship.
Context	A description of the context of the relationship.
InverseContext	A description of the inverse of the context of the relationship.

Table A.1.25: Specifications for the o*bj_rel_type* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
RelationshipTypeID	+	✓	✓			
RelationshipName	VARCHAR (40)		✓			
Context	VARCHAR (40)		✓			
InverseContext	VARCHAR (40)		✓			

Units of Measure Table: *UnitsOfMeasure*

A system of units of measure based on the system used in OpenMI and described in HarmonIT (2005c) has been implemented in SPATSIM-HDSF. The OpenMI system refers to units and dimensions, where a unit has a magnitude and can be used to measure things and a dimension describes the type of thing being measured. In the SI system there are seven base dimensions of measure as shown in Table A.1.26 and currency has been added as another dimension by OpenMI. A unit of measure is described by its dimensions, a conversion factor to SI units and a conversion offset to SI units. For a given value the conversion to an SI value is done using Equation 1.

Table A.1.26: Base dimensions and units used in OpenMI (after HarmonIT (2005c))

Dimension Base	**SI Base Unit**	**Symbol Used**
Length	meter	m
Mass	kilogram	kg
Time	second	s
ElectricCurrent	ampere	A
Temperature	kelvin	K
AmountOfSubstance	mole	mol
LuminousIntensity	candela	cd
Currency	Euro	E

$$SI\_Value = ConversionFactorToSI * Value + ConversionOffsetToSI \qquad (1)$$

The fields in the new *UnitsOfMeasure* table are described in Table A.1.27. The *UnitID* field is the primary key field with an Autoincrement (+) data type and acts as a unique identifier to reference each unit of measure. The *UnitName* field will hold a short text representation of the unit of measure (e.g. “m^3/s”). The *UnitDesc* field will hold a longer text description of the unit of measure. The *UnitCategory* field will allow units of measure to be grouped into categories (e.g. “Flow Rate”) to enable users to more easily locate a required unit of measure. The *ConversionFactorToSI* field will hold values specifying the conversion factor to convert the unit of measure to SI units and would have a default value of “1.0”. The *ConversionOffsetToSI* field will hold values specifying the conversion offset to convert the unit of measure to SI units and would have a default value of “0.0”. The remaining fields represent the eight base dimensions shown in Table A.1.26 and store the power to which the SI base unit forming the unit of measure is raised, the default value for each of these fields is zero. For example, to represent the dimensions of cubic meters per second, the *Dim_Length* field would have a value of “3” and the *Dim_Time* field would have a value of “-1”. The dimensions of cubic meters per hour would be the same as for cubic meters per second but a conversion factor of “1/3600” to SI units would be specified. The *UnitsOfMeasure* table has been populated with the units of measure listed in Appendix A.2. The advantages of the new *UnitsOfMeasure* table are that it fully describes the units of measure, enables conversions between units of measure and allows for new units of measure to be easily added. However, for this system of units of measure to work it is important that all SPATSIM-HDSF databases use exactly the same units of measure table so that if data is moved between SPATSIM-HDSF project databases units of measure can simply be transferred. Users of SPATSIM-HDSF may not add or edit records in the *UnitsOfMeasure* table, if a new unit of measure is required by a user the developers of SPATSIM-HDSF will add it to the table which will be made available to all users.

Table A.1.27: Description of the fields in the *UnitsOfMeasure* table

Field	**Description**
UnitID	A unique ID number associated with the unit of measure.
UnitName	A short name for the unit of measure.
UnitDesc	A description of the unit of measure.
UnitCategory	The category to which the unit of measure belongs.
ConversionFactorToSI	The factor to be used to convert a value with this unit of measure to the SI unit of measure.

ConversionOffsetToSI	The offset to be used to convert a value with this unit of measure to the SI unit of measure.
Dim_Length	The power of the length dimension of the unit of measure.
Dim_Mass	The power of the mass dimension of the unit of measure.
Dim_Time	The power of the time dimension of the unit of measure.
Dim_ElectricCurrent	The power of the electric current dimension of the unit of measure.
Dim_Temperature	The power of the temperature dimension of the unit of measure.
Dim_AmountOfSubstance	The power of the amount of substance dimension of the unit of measure.
Dim_LuminousIntensity	The power of the luminous intensity dimension of the unit of measure.
Dim_Currency	The power of the currency dimension of the unit of measure.

Table A.1.28: Specifications for the *UnitsOfMeasure* table

Field	**Type**	**Primary Key**	**Required**	**Maximum Value**	**Minimum Value**	**Default Value**
UnitID	COUNTER	✓	✓			
UnitName	VARCHAR (20)		✓			
UnitDesc	VARCHAR (100)					
UnitCategory	VARCHAR (20)		✓			
ConversionFactorToSI	DOUBLE		✓			1.0
ConversionOffsetToSI	DOUBLE		✓			0.0
Dim_Length	SMALLINT		✓			0
Dim_Mass	SMALLINT		✓			0
Dim_Time	SMALLINT		✓			0
Dim_ElectricCurrent	SMALLINT		✓			0
Dim_Temperature	SMALLINT		✓			0
Dim_AmountOfSubstance	SMALLINT		✓			0
Dim_LuminousIntensity	SMALLINT		✓			0
Dim_Currency	SMALLINT		✓			0

Attribute Value Table: *TextData*

Table A.1.29: Description of the fields in a *TextData* table (after Hughes, 2005; IWR, 2005)

Field	**Description**
RECID	A unique ID number associated with each attribute data record.
TextValue	Text attribute data.

Table A.1.30: Specifications for the *TextData* table

Field	**Type**	**Primary Key**	**Required**	**Maximum Value**	**Minimum Value**	**Default Value**
RECID	COUNTER	✓	✓			
TextValue	VARCHAR (80)		✓			

Attribute Value Table: *IntegerData*

Table A.1.31: Description of the fields in an *IntegerData* table (after Hughes, 2005; IWR, 2005)

Field	**Description**
RECID	A unique ID number associated with each attribute data record.
IntValue	Integer attribute value.

Table A.1.32: Specifications for the *IntegerData* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
RECID	COUNTER	✓	✓			
IntValue	INTEGER		✓			

Attribute Value Table: *RealData*

Table A.1.33: Description of the fields in a *RealData* table (after Hughes, 2005; IWR, 2005)

Field	Description
RECID	A unique ID number associated with each attribute data record.
RealValue	Real attribute value.

Table A.1.34: Specifications for the *RealData* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
RECID	COUNTER	✓	✓			
RealValue	DOUBLE		✓			

Attribute Value Table: *TimeSeriesData*

Table A.1.35: Description of the fields in a *TimeSeriesData* table (after Hughes, 2005; IWR, 2005)

Field	Description
RECID	A unique ID number associated with each attribute data record.
GeogID	A text description of the data source location.
DataID	A text description of the data type (e.g. "Daily Rainfall").
DataObj	A data type code: 0 = fixed interval (data value only), 1 = variable interval (pairs of date/time and data value), 2 = variable interval (pairs of interval length and data value).
TimeUnits	The units of the time data: 1 = minutes, 2 = hours, 3 = days, 4 = months, 5 = years.
TimeInt	The time interval (when the data type code in the *Data Obj.* field is 0).
MDCode	The missing data code used in the time-series data.
UnitID	A unique ID number associated with the unit of measure. References the *UnitID* field in the *UnitsOfMeasure* table.
AggType	The aggregation type of the time-series data: 0 = should not be aggregated, 1= sum, 2 = average, 3 = maximum, 4 = minimum.
Scale	The value to be used to scale the data.
FlagKey	A unique ID number associated with the flag key. References the *FlagKey* field in the *tsflag* table.
StartDateTime	The start date and time of the time-series.
EndDateTime	The end date and time of the time-series.
NumRecords	The number of records in the time-series.
TSData	The time-series stored in a BLOB as a series of values for fixed interval data or stored as a series of time and value pairs.

Table A.1.36: Specifications for the *TimeSeriesData* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
RECID	COUNTER	✓	✓			
GeogID	VARCHAR (80)					
DataID	VARCHAR (80)					
DataObj	SMALLINT		✓			
TimeUnits	SMALLINT		✓			
TimeInt	SMALLINT		✓			
MDCode	DOUBLE		✓			
UnitID	INTEGER		✓			
AggType	SMALLINT		✓			
Scale	DOUBLE		✓			
FlagKey	INTEGER		✓			
StartDateTime	DATETIME		✓			
EndDateTime	DATETIME		✓			
NumRecords	INTEGER		✓			
TSData	LONGBINARY		✓			

Time-series Flag Table: *tsflag*

Table A.1.37: Description of the fields in a *tsflag* table

Field	Description
FlagKey	A unique ID number associated with each flag key.
Description	Text description of the flag key.
F1…F30	Flag.
FD1…FD30	Flag.

Table A.1.38: Specifications for the *tsflag* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
FlagKey	COUNTER	✓	✓			
Description	VARCHAR (40)		✓			
F1	VARCHAR (10)		✓			
FD	VARCHAR (40)		✓			
F2...F30	VARCHAR (10)					
FD2…FD30	VARCHAR (40)					

Attribute Value Table: *PictureData*

Table A.1.39: Description of the fields in a *PictureData* table (after Hughes, 2005; IWR, 2005)

Field	Description
RECID	A unique ID number associated with each attribute data record.
Title	A text title for the picture.
Xsize	The horizontal size of the picture in pixels.
Ysize	The vertical size of the picture in pixels.
RecDate	The date of creation or loading of the picture.
Type	A type code for the picture file: 0 = bmp, 1 = cgm, 2 = jpg, 3 = pcx, 4 = tif, 5 = gif, 6 = wmf, 7 = mpg, 8 = avi, 9 = mov
Picture	The picture stored as a BLOB.

Table A.1.40: Specifications for the *PictureData* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
RECID	COUNTER	✓	✓			
Title	VARCHAR (80)					
Xsize	SMALLINT					
Ysize	SMALLINT					
RecDate	DATETIME					
Type	SMALLINT		✓			
Picture	LONGBINARY		✓			

Attribute Value Table: *ArrayData*

Table A.1.41: Description of the fields in an *ArrayData* table (after Hughes, 2005; IWR, 2005)

Field	Description
RECID	A unique ID number associated with each attribute data record.
Title	A text title for the array.
RecDate	The date of creation or loading of the array.
NumRows	The number of rows in the array.
NumColumns	The number of columns in the array.
Format	The filename of a text file (*.*txp*) specifying the structure of the array.
ArrayData	The values in the array stored as a BLOB, where each value is represented by a 4 byte real number.

Table A.1.42: Specifications for the *ArrayData* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
RECID	COUNTER	✓	✓			
Title	VARCHAR (80)					
RecDate	DATETIME					
NumRows	SMALLINT					
NumColumns	SMALLINT					
Format	VARCHAR (80)		✓			
ArrayData	LONGBINARY		✓			

Attribute Value Table: *MemoData*

Table A.1.43: Description of the fields in a *MemoData* table (after Hughes, 2005; IWR, 2005)

Field	Description
RECID	A unique ID number associated with each attribute data record.
Title	A text title for the memo.
RecDate	The date of creation or loading of the memo.
MemoValue	Text data stored in a memo type field

Table A.1.44: Specifications for the *MemoData* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
RECID	COUNTER	✓	✓			
Title	VARCHAR (80)					
RecDate	DATETIME					
MemoValue	LONGCHAR		✓			

Attribute Value Table: *LinkData*

Table A.1.45: Description of the fields in a *LinkData* table (after Hughes, 2005; IWR, 2005)

Field	Description
RECID	A unique ID number associated with each attribute data record.
Feature	The ID number of the feature acting as the data source. References the *FeatureCode* field of the *data_dict1* table.
Attribute	The ID number of the attribute acting as the data source. References the *AttributeCode* field of the *data_dict3* table.
Type	The data type of the data source attribute as specified in the *data_dict3* table.
ObjectRecord	The ID number of the source object that will be used to locate the data. References the *IDField* of the *ShapefileDbfTable* or *NonSpatialFeatureTable* table for the relevant feature.

Table A.1.46: Specifications for the *LinkData* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
RECID	COUNTER	✓	✓			
Feature	INTEGER		✓			
Attribute	INTEGER		✓			
Type	SMALLINT		✓			
ObjectRecord	INTEGER		✓			

Internal GIS Feature Theme Table: FeatureTheme

Table A.1.47: Description of the fields in a *FeatureTheme* table

Field	Description
ID	String containing a GUID used to identify the feature theme.
FeatureCode	A unique ID number associated with a single feature. References the *FeatureCode* field in the *data_dict1* table.
ThemeName	The name of the feature theme for display and selection purposes.
ThemeData	A string storing a XML serialised theme definition object used to store information about the feature theme.
IsDefault	A Boolean value specifying whether this feature theme is the default theme for the feature to which it belongs.

Table A.1.48: Specifications for the *FeatureTheme* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
ID	VARCHAR(40)	✓	✓			
FeatureCode	INTEGER		✓			
ThemeName	VARCHAR(255)		✓			
ThemeData	LONGCHAR		✓			
IsDefault	BOOLEAN		✓			

External GIS Base Data Table: ETBaseData

Table A.1.49: Description of the fields in a *ETBaseData* table

Field	Description
ETBaseDataID	A unique ID number associated with a base dataset.
DataSetSource	The path and filename of a base dataset.
DataSetName	The name of the base dataset for display and selection purposes.
FeatureType	An integer number identifying the type of feature: 1 = Spatial, 2 = Image, 3 = Non-spatial, 4 = GRID.

Table A.1.50: Specifications for the *ETBaseData* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
ETBaseDataID	COUNTER	✓	✓			
DataSetSource	VARCHAR(255)		✓			
DataSetName	VARCHAR(50)		✓			
FeatureType	SMALLINT		✓			

External GIS Automatic Calculation Table: ETAutoCalc

Table A.1.51: Description of the fields in a *ETAutoCalc* table

Field	Description
ETAutoCalcID	A unique ID number assigned to each record.
AttributeCode	A unique ID number associated with the attribute. References the *AttributeCode* field in the *data_dict3* table.
ETBaseDataID	A unique ID number associated with a base dataset. References the *ETBaseDataID* field in the *ETBaseData* table.

Table A.1.52: Specifications for the *ETAutoCalc* table

Field	Type	Primary Key	Required	Maximum Value	Minimum Value	Default Value
ETAutoCalcID	COUNTER	✓	✓			
AttributeCode	INTEGER		✓			
ETBaseDataID	INTEGER		✓			

A.2 Units of Measure Included in the *UnitsOfMeasure* Table

Each unit of measure has been assigned to a category. The units of measure and their properties are shown in the tables below. The conversion factors and offsets to SI units were based on SABS (1973).

Table A.2.1: Units of measure for the Area category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
mm^2	square millimetre	Area	1.0 E-6	0.0	2	0	0	0	0	0	0	0
cm^2	square centimetre	Area	1.0 E-4	0.0	2	0	0	0	0	0	0	0
m^2	square metre	Area	1.0	0.0	2	0	0	0	0	0	0	0
ha	hectare	Area	1.0 E+4	0.0	2	0	0	0	0	0	0	0
km^2	square kilometre	Area	1.0 E+6	0.0	2	0	0	0	0	0	0	0
ft^2	square foot	Area	9.290304 E-2	0.0	2	0	0	0	0	0	0	0
acre	acre	Area	4.04686 E+3	0.0	2	0	0	0	0	0	0	0
mi^2	square mile	Area	2.589988 E+6	0.0	2	0	0	0	0	0	0	0

Table A.2.2: Units of measure for the Concentration category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
mg/ℓ	milligram per litre	Concentration	1.0 E-3	0.0	-3	1	0	0	0	0	0	0
g/ℓ	gram per litre	Concentration	1.0	0.0	-3	1	0	0	0	0	0	0
kg/m^3	kilogram per cubic metre	Concentration	1.0	0.0	-3	1	0	0	0	0	0	0
t/m^3	tonne per cubic metre	Concentration	1.0 E+3	0.0	-3	1	0	0	0	0	0	0
cwt/ft^3	hundredweight per cubic foot	Concentration	1.794068 E+3	0.0	-3	1	0	0	0	0	0	0
cwt/gal	hundred weight per gallon (UK)	Concentration	1.117496 E+4	0.0	-3	1	0	0	0	0	0	0
ton/ft^3	ton (long) per cubic foot	Concentration	3.588136 E+4	0.0	-3	1	0	0	0	0	0	0
ton/gal	ton (long) per gallon	Concentration	2.234991 E+5	0.0	-3	1	0	0	0	0	0	0

Table A.2.3: Units of measure for the Cost category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
R	Rand	Cost	1.0	0.0	0	0	0	0	0	0	0	1
R/ha	Rand per hectare	Cost	1.0 E-4	0.0	-2	0	0	0	0	0	0	1
R/m^3	Rand per cubic metre	Cost	1.0	0.0	-3	0	0	0	0	0	0	1
R/kℓ	Rand per kilo litre	Cost	1.0	0.0	-3	0	0	0	0	0	0	1
R/kg	Rand per kg	Cost	1.0	0.0	0	-1	0	0	0	0	0	1
R/t	Rand per ton	Cost	1.0 E-3	0.0	0	-1	0	0	0	0	0	1
R/kW h	Rand per kilo watt hour	Cost	2.777778 E-7	0.0	-2	-1	2	0	0	0	0	1

Table A.2.4: Units of measure for the Dimensionless category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstanc	Dim_LuminousIntensity	Dim_Currency
?	Unknown	Dimensionless	0.0	0.0	0	0	0	0	0	0	0	0
-	Dimensionless	Dimensionless	0.0	0.0	0	0	0	0	0	0	0	0
Fraction	Fraction	Dimensionless	1.0	0.0	0	0	0	0	0	0	0	0
Ratio	Ratio	Dimensionless	1.0	0.0	0	0	0	0	0	0	0	0
%	percent	Dimensionless	1.0 E-2	0.0	0	0	0	0	0	0	0	0
pH	pH	Dimensionless	1.0	0.0	0	0	0	0	0	0	0	0
Index	Index	Dimensionless	1.0	0.0	0	0	0	0	0	0	0	0

Table A.2.5: Units of measure for the Electrical category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
A	ampere	Electrical	1.0	0.0	0	0	0	1	0	0	0	0
V	volt	Electrical	1.0	0.0	2	1	-3	-1	0	0	0	0
Ω	ohm	Electrical	1.0	0.0	2	1	-3	-2	0	0	0	0
C	coulomb	Electrical	1.0	0.0	0	0	1	1	0	0	0	0
F	farad	Electrical	1.0	0.0	-2	-1	4	2	0	0	0	0
S	siemens	Electrical	1.0	0.0	-2	-1	3	2	0	0	0	0
MS/m	mega siemens per metre	Electrical	1.0	0.0	-3	-1	3	2	0	0	0	0

Table A.2.6: Units of measure for the Energy category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
J	joule	Energy	1.0	0.0	2	1	-2	0	0	0	0	0
W	watt	Energy	1.0	0.0	2	1	-3	0	0	0	0	0
kW	kilowatt	Energy	1.0 E+3	0.0	2	1	-3	0	0	0	0	0
MW	megawatt	Energy	1.0 E+6	0.0	2	1	-3	0	0	0	0	0
kWh	kilowatt hour	Energy	3.6 E+6	0.0	2	1	-2	0	0	0	0	0
MJ/cm^2	mega joules per square centimetre	Energy	1.0 E+10	0.0	0	1	-2	0	0	0	0	0
MJ/m^2	mega joules per square metre	Energy	1.0 E+6	0.0	0	1	-2	0	0	0	0	0
$MJ/cm^2/d$	mega joules per square centimetre per day	Energy	1.157407 E+5	0.0	0	1	-2	0	0	0	0	0
$MJ/m^2/d$	mega joules per square metre per day	Energy	1.157407 E+1	0.0	0	1	-2	0	0	0	0	0
J/kg	joule per kilogram	Energy	1.0	0.0	2	0	-2	0	0	0	0	0
J/kg.°C	joules per kilogram per degree Celsius	Energy	1.0	0.0	2	0	-2	0	-1	0	0	0

Table A.2.7: Units of measure for the Flux category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
mm/h	millimetre per hour	Flux	2.777778 E-7	0.0	1	0	-1	0	0	0	0	0
mm/d	millimetre per day	Flux	1.157407 E-8	0.0	1	0	-1	0	0	0	0	0
mm/month	millimetre per month	Flux	0.0	0.0	1	0	-1	0	0	0	0	0
mm/annum	millimetre per annum	Flux	0.0	0.0	1	0	-1	0	0	0	0	0
m/s	metre per second	Flux	1.0	0.0	1	0	-1	0	0	0	0	0
m/h	metre per hour	Flux	2.777778 E-4	0.0	1	0	-1	0	0	0	0	0
m/d	metre per day	Flux	1.157407 E-5	0.0	1	0	-1	0	0	0	0	0
km/h	kilometres per hour	Flux	2.777778 E-1	0.0	1	0	-1	0	0	0	0	0
km/d	kilometres per day	Flux	1.157407 E-2	0.0	1	0	-1	0	0	0	0	0
ft/s	foot per second	Flux	3.048 E-1	0.0	1	0	-1	0	0	0	0	0
mi/h	miles per hour	Flux	4.4704 E-1	0.0	1	0	-1	0	0	0	0	0
mi/d	miles per day	Flux	1.862667 E-2	0.0	1	0	-1	0	0	0	0	0

Table A.2.8: Units of measure for the Length category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
µm	micrometre	Length	1.0 E-6	0.0	1	0	0	0	0	0	0	0
mm	millmetre	Length	1.0 E-3	0.0	1	0	0	0	0	0	0	0
cm	centimetre	Length	1.0 E-2	0.0	1	0	0	0	0	0	0	0
m	metre	Length	1.0	0.0	1	0	0	0	0	0	0	0
km	kilometre	Length	1.0 E+3	0.0	1	0	0	0	0	0	0	0
mm/mm	millimetre per millimetre	Length	1.0	0.0	0	0	0	0	0	0	0	0
m/m	metre per metre	Length	1.0	0.0	0	0	0	0	0	0	0	0
in	inch	Length	2.54 E-2	0.0	1	0	0	0	0	0	0	0
ft	foot	Length	3.048 E-1	0.0	1	0	0	0	0	0	0	0
yd	yard	Length	9.144 E-1	0.0	1	0	0	0	0	0	0	0
mi	mile	Length	1.609344 E+3	0.0	1	0	0	0	0	0	0	0

Table A.2.9: Units of measure for the Mass category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
mg	milligram	Mass	1.0 E-6	0.0	0	1	0	0	0	0	0	0
g	gram	Mass	1.0 E-3	0.0	0	1	0	0	0	0	0	0
kg	kilogram	Mass	1.0	0.0	0	1	0	0	0	0	0	0
t	tonne	Mass	1.0 E+3	0.0	0	1	0	0	0	0	0	0
cwt (UK)	hundredweight (UK)	Mass	5.080235 E+1	0.0	0	1	0	0	0	0	0	0
ton (long)	ton (long)	Mass	1.016047 E+3	0.0	0	1	0	0	0	0	0	0
g/g	gram per gram	Mass	1.0	0.0	0	0	0	0	0	0	0	0

Table A.2.10: Units of measure for the Mechanical category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
N	newton	Mechanical	1.0	0.0	1	1	-2	0	0	0	0	0
Nm	newton metre	Mechanical	1.0	0.0	2	1	-2	0	0	0	0	0
N/m^2	newton per square metre	Mechanical	1.0	0.0	-1	1	-2	0	0	0	0	0
m/s^2	metres per second squared	Mechanical	1.0	0.0	1	0	-2	0	0	0	0	0

Table A.2.11: Units of measure for the Plane Angle category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
rad	radian	Plane Angle	1.0	0.0	0	0	0	0	0	0	0	0
°	degree	Plane Angle	1.745329 E-2	0.0	0	0	0	0	0	0	0	0
‘	minute	Plane Angle	2.908882 E-4	0.0	0	0	0	0	0	0	0	0
“	second	Plane Angle	4.848137 E-6	0.0	0	0	0	0	0	0	0	0

Table A.2.12: Units of measure for the Pressure category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
Pa	pascal	Pressure	1.0	0.0	-1	1	-2	0	0	0	0	0
kPa	kilopascal	Pressure	1.0 E+3	0.0	-1	1	-2	0	0	0	0	0
bar	bar	Pressure	1.0 E+5	0.0	-1	1	-2	0	0	0	0	0
Pa/°C	pascal per degree Celsius	Pressure	1.0	0.0	-1	1	-2	0	-1	0	0	0

Table A.2.13: Units of measure for the Production category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
g/m^2	grams per square metre	Production	1.0 E-3	0.0	-2	1	0	0	0	0	0	0
g/ha	grams per hectare	Production	1.0 E-7	0.0	-2	1	0	0	0	0	0	0
Mg/ha	mega grams per hectare	Production	1.0 E-1	0.0	-2	1	0	0	0	0	0	0
kg/ha	kilograms per hectare	Production	1.0 E-4	0.0	-2	1	0	0	0	0	0	0
t/ha	tonnes per hectare	Production	1.0 E-1	0.0	-2	1	0	0	0	0	0	0
ton/acre	ton (long) per acre	Production	2.51071 E-1	0.0	-2	1	0	0	0	0	0	0
m^3/ha	cubic metres per hectare	Production	1.0 E-4	0.0	1	0	0	0	0	0	0	0

Table A.2.14: Units of measure for the Rate category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
mℓ/d	millilitre per day	Rate	1.157407 E-11	0.0	3	0	-1	0	0	0	0	0
ℓ/s	litre per second	Rate	1.0 E-3	0.0	3	0	-1	0	0	0	0	0
m^3/s	cubic metre per second	Rate	1.0	0.0	3	0	-1	0	0	0	0	0
m^3/h	cubic metre per hour	Rate	2.777778 E-4	0.0	3	0	-1	0	0	0	0	0
m^3/d	cubic metre per day	Rate	1.157407 E-5	0.0	3	0	-1	0	0	0	0	0
m^3/month	cubic metre per month	Rate	0.0	0.0	3	0	-1	0	0	0	0	0
m^3/annum	cubic metre per annum	Rate	0.0	0.0	3	0	-1	0	0	0	0	0
Mm^3/d	million cubic metres per day	Rate	1.157407 E+1	0.0	3	0	-1	0	0	0	0	0
Mm^3/month	million cubic metres per month	Rate	0.0	0.0	3	0	-1	0	0	0	0	0
Mm^3/annum	million cubic metres per annum	Rate	0.0	0.0	3	0	-1	0	0	0	0	0
ft^3/s	cubic feet per second	Rate	2.831685 E-2	0.0	3	0	-1	0	0	0	0	0
gall/h	gallon (UK) per hour	Rate	1.262803 E-6	0.0	3	0	-1	0	0	0	0	0
gall/d	gallon (UK) per day	Rate	5.261678 E-8	0.0	3	0	-1	0	0	0	0	0

Table A.2.15: Units of measure for the Temperature category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
K	Kelvin	Temperature	1.0	0.0	0	0	0	0	1	0	0	0
°C	degrees Celsius	Temperature	1.0	2.7315 E+2	0	0	0	0	1	0	0	0
° F	degrees Farenheit	Temperature	5.5555556 E-1	2.5537222 E+2	0	0	0	0	1	0	0	0
°C/1000 m	degrees Celsius per 1000 metres	Temperature	1.0 E-3	2.7315 E+2	-1	0	0	0	1	0	0	0
°C days	growing degree days (°C)	Temperature	8.64 E+4	2.7315 E+2	0	0	1	0	1	0	0	0

Table A.2.16: Units of measure for the Time category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
s	second	Time	1.0	0.0	0	0	1	0	0	0	0	0
min	minute	Time	6.0 E+1	0.0	0	0	1	0	0	0	0	0
h	hour	Time	3.6 E+3	0.0	0	0	1	0	0	0	0	0
d	day	Time	8.64 E+4	0.0	0	0	1	0	0	0	0	0
week	week	Time	6.048 E+5	0.0	0	0	1	0	0	0	0	0
month	month	Time	0.0	0.0	0	0	1	0	0	0	0	0
year	year	Time	0.0	0.0	0	0	1	0	0	0	0	0

Table A.2.17: Units of measure for the Volume category

UnitName	UnitDesc	UnitCategory	ConversionFactorToSI	ConversionOffsetToSI	Dim_Length	Dim_Mass	Dim_Time	Dim_ElectricCurrent	Dim_Temperature	Dim_AmountOfSubstance	Dim_LuminousIntensity	Dim_Currency
mℓ	millilitre	Volume	1.0 E-6	0.0	3	0	0	0	0	0	0	0
ℓ	litre	Volume	1.0 E-3	0.0	3	0	0	0	0	0	0	0
kℓ	kilolitre	Volume	1.0	0.0	3	0	0	0	0	0	0	0
Mℓ	mega litre	Volume	1.0 E+3	0.0	3	0	0	0	0	0	0	0
dm^3	decimetre cubed	Volume	1.0 E-3	0.0	3	0	0	0	0	0	0	0
m^3	cubic metre	Volume	1.0	0.0	3	0	0	0	0	0	0	0
Mm^3	million cubic metres	Volume	1.0 E+6	0.0	3	0	0	0	0	0	0	0
ft^3	cubic foot	Volume	2.831685 E-2	0.0	3	0	0	0	0	0	0	0
acre ft	acre foot	Volume	1.233482 E+3	0.0	3	0	0	0	0	0	0	0
gall	gallon (UK)	Volume	4.54609 E-3	0.0	3	0	0	0	0	0	0	0
Mill.gall	million gallons (UK)	Volume	4.54609 E+3	0.0	3	0	0	0	0	0	0	0
m^3/m^3	cubic metre per cubic metre	Volume	1.0	0.0	0	0	0	0	0	0	0	0

A.3 Example a Configuration Schema Used in the Configuration Editor Program

Portion of a configuration schema file for the Configuration Editor program showing examples of the definitions for an entity (EntitySchema), a group (Group), a variable (FieldDefinition) and a data rule (DataRule).

A.4 Example of Configuration Editor Data and Display Rules

Examples of data validation and display rules used by the Configuration Editor program. These rules are written in the C# programming language.

Example of a Data Rule

[illegible]

Example of a Display Rule

[illegible]

A.5 Example of a CSV Data Storage Format Used by the Configuration Editor

Example of CSV data storage format used by the Configuration Editor program.

```
□□□□□□
□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□
□□□ □□□□□
□□□ □□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□
□□□□□□□□□□□ □□□□□
□□□□□□□□□□□ □□□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□□□
□□□□□□□□□
□□□□□□□□□
□□□□□□
```

Appendix B

B.1 Inception Report
See file AppendixB/B.1_InceptionReport.pdf on DVD accompanying this report.

B.2 User Requirement Survey Report
See file AppendixB/B.2_UserRequirementSurveyReport.pdf on DVD accompanying this report.

B.3 Framework Review Report
See file AppendixB/B.3_FrameworkReviewReport.pdf on DVD accompanying this report.

B.4 Framework Selection Report
See file AppendixB/B.4_FrameworkSelectionReport.pdf on DVD accompanying this report.

B.5 Framework Design Report
See file AppendixB/B.5_FrameworkDesignReport.pdf on DVD accompanying this report.

B.6 Framework Development Report 1
See file AppendixB/B.6_FrameworkDevelopmentReport1.pdf on DVD accompanying this report.

B.7 Framework Development Report 2
See file AppendixB/B.7_FrameworkDevelopmentReport2.pdf on DVD accompanying this report.

B.8 Framework Development Report 3
See file AppendixB/B.8_FrameworkDevelopmentReport3.pdf on DVD accompanying this report.

B.9 Framework Development Report 4
See file AppendixB/B.9_FrameworkDevelopmentReport4.pdf on DVD accompanying this report.

B.10 User Manual for External GIS Tools
See file AppendixB/B.10_ExternalGISTools_UserManual.pdf on DVD accompanying this report.

B.11 Modelling Report 1
See file AppendixB/B.11_ModellingReport1.pdf on DVD accompanying this report.

B.12 Modelling Report 2
See file AppendixB/B.12_ModellingReport2.pdf on DVD accompanying this report.

B.13 Modelling Report 3
See file AppendixB/B.13_ModellingReport3.pdf on DVD accompanying this report.

B.14 Modelling Report 4
See file AppendixB/B.14_ModellingReport4.pdf on DVD accompanying this report.

Appendix C

C.1 Quaternary Catchments Shapefile

See folder AppendixC/QCShapefile on the DVD accompanying this report.

C.2 Climate Time-series Database – SPATSIM-HDSF Database

See file AppendixC/ClimateTimeSeriesDatabase/SpatsimHdsfDatabase on the DVD accompanying this report.

C.3 Climate Time-series Database – *ACRU* Composite (Y2K) Files

See folder AppendixC/ClimateTimeSeriesDatabase/AcruCompositeFiles on the DVD accompanying this report.

C.4 Quaternary Catchments Database – SPATSIM-HDSF Database

See folder AppendixC/QCDatabase/SpatsimHdsfDatabase on the DVD accompanying this report.

C.5 Quaternary Catchments Database – *ACRU* 300 Series Input Data Files

See folder AppendixC/QCDatabase/AcruInputDataFiles on the DVD accompanying this report.

Appendix D

This Appendix contains the apha version of the SPATSIM-HDSF modelling framework and other tools developed in this project.

The SPATSIM website [http://www.ru.ac.za/institutes/iwr/software/spatsim.html] contains includes a webpage [http://www.ru.ac.za/institutes/iwr/software/spatsimupdate.html] from which SPATSIM-HDSF software updates may be downloaded.

D.1 SPATSIM-HDSF Modelling Framework

See folder AppendixD/SPATSIM-HDSF on the DVD accompanying this report.

D.2 SPATSIM-HDSF External GIS Tools

See folder AppendixD/ExternalGISTools on the DVD accompanying this report.

D.3 SPATSIM-HDSF Data Access Class Library

See folder AppendixD/DataAccessClassLibrary on the DVD accompanying this report.

D.4 SPATSIM-HDSF Data Forms Library

See folder AppendixD/DataFormsLibrary on the DVD accompanying this report.

D.5 SPATSIM-HDSF Time-series Analysis Library

See folder AppendixD/TSAnalysisLibrary on the DVD accompanying this report.