Design and Implementation of the GDManager Geothermal Data Management System

E.B. ANDERSON, G.B. CLARK, G.N.H. USSHER

Geothermal Energy New Zealand Limited, Aucklaiid.

Key words: geothermal database, database structure, data collection, New Zealand

ABSTRACT

The GDManager geothermal database system has proven to be successful in the storage and management of data collected from a geothermal resource, and is now widely used in man) geothermal projects Its strength and flexibility is reflected in the fact that not only is there an ever expanding toolbox of utilities for both raw data entry and data analysis, but in many instances it is possible tu integrate a client's existing software uith the system.

A variety of software construction and management tools have been developed specifically to look after the system to provide a high level of reliability. The structure and operation of the system is being progressively opened to allow greater user flexibility. The structured database provides an open link between standard GDManager operations and any tools that the user wishes io independently apply to their data.

1. INTRODUCTION

As our worldwide geothermal industry continues to grow an increasing number of geothcnnal fields are being explored and developed uith an associated growth in the volume of data collected from these systems. Increasingly, fields are being compared against one another to prioritise development strategies. Development rights lo fields are being actively traded in many areas As a consequence. there is a grouing acknowledgment that a significant portion of the value of a geothermal resource lies in the accumulated knowledge about that resource

Operating as a consultant for more than two decades, Geothermal Energy New Zealand Ltd (GENZL) has been required to perform many resource reviews and feasibility studies for a variety of clients A major part of this uork has been the collation of data from a wide range of sources, reinterpretation of data (often requiring re-evaluation of data at a basic level) and integration into a conceptual model. 'She GENZL scientific team saw the need to have an orderly way of entering and processing this data. Equally necessary was a system for managing the usual collection of interpretation and modelling tools that a scientific operation accumulates

GDManager, an integrated data management system centred around a formalised database, was conceived as a too! for solving these data management needs iur us as a

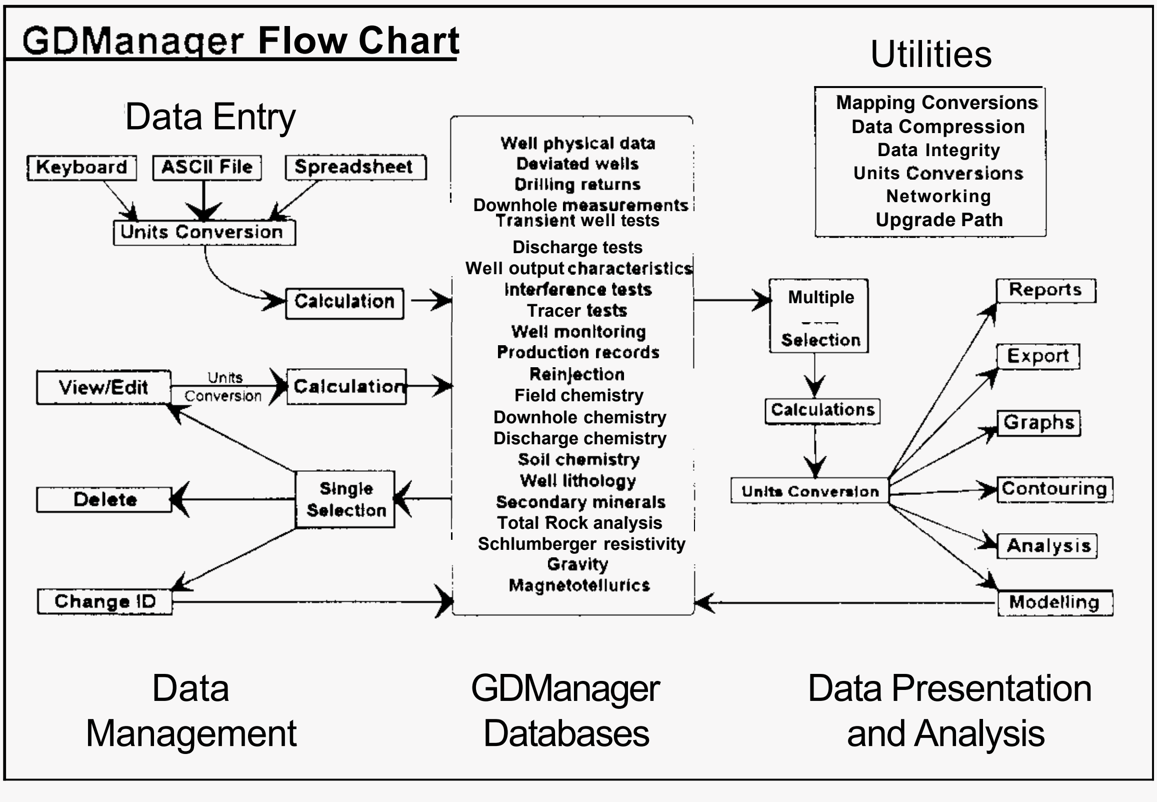

Figure 1 **Flow chart showing the generalised functional** processes **within** GDManager

consultant, but it **was** also clear that most of our clients could certainly benefit from such **a** system From the start of this project in **1986-87**, it was decided to make GDManager commercially available. This has required that the system be well structured and supportable **As a** result, the system has been progressively improvcd **over** the past *8* yrars to where it is now being **used** by **a** wide selection of geothermal operators worldwide.

Increasingly, users are wanting to utilise their data in a greater variety of ways now that GDManager has simplified most of tlie mundane date management tasks.

Recent papers (Anderson and Ussher 1992, GENZL 1994) have detailed the functionality of GDManager which is built utilising the Paradox PC-based database system. The objective of this paper **is** to outline Some of the underlying structure of the system and haw it can be opened to provide greater user flexibility

2. GDMANAGER OVERVIEW

Geothermal data processing requires three main functions

- data collection and storage
- calculations and modelling
- data presentation and visualisation

GDManager delivers all thesc functions but also has the potential to be used as an open system which will allow users to add their own functionality. The basic linking element is the database uhich holds raw, summarised and calculated data.

Figure 1 shows a flow chart of how these main functions are implemented within GDManagcr. At the centre of the system is the database with man) tables which hold data for each of tlie various data types that the system manages. **A** variety of data **entry,** importing and editing routines provide for data collection, quality assurance and management within each module Once data is held within the database, it is available for further calculation or niodclling. Results of such calculations or models can be returned to the database. Finally, raw, calculated, modelled and summarised data must be presented in a visual form to be useful. This visualisation utilises X-Y graphics, contouring and other graphical tools.

3. DATABASE STRUCTURE

GDManager uses the relational model for structuring data. The total database structure **is** a formalised definition of the parameters which we can measure and determine within the geothermal system and, to some extent, how these parameters interact. The structure is a representation of both the physical geothermal system and the measurement process we use to observe this system

A well designed database system strongly contrasts with systems based on **spreadsheets** or stand-alone graphical packages where data is usually stored in separate files for each well or test. With data scattered across many tiles in spreadsheet systems, the data is difficult to find and it is difficult or impossible to correlate or integrate data from the different files In true database systems, however, all data of a particular type is kept in one file (or table)

In GDManager, each data table represents some physical or abstract type of object within the geothermal system. For example - Wells, Sample Sites, Well Deviation Measurements, Well Tests, Temperature-Pressure-Flow Measurements and Interpreted Resource 'Temperature or Pressure

'She relational model allows for defining relationships between tables. For example, *each* Well listed in tlie WELL table may have *many* Well Deviation Measurements in the WELL DEVIATION tablc and *many* Well Tests in the WELL TEST DIARY tahlc, and in turn *each* Well Test may **have** *many* Temp-Pres-Flow Measurements in the WELL MEASUREMENT table Each **tablc** is composed of several fields which specify the attributes of tlie objects (or records) in the tablc **A** Well is a real object and a Temperature Measurement is a real physical parameter hut a Well 'lest is just a convenient grouping of many measurements. This is demonstrated in Figure 2

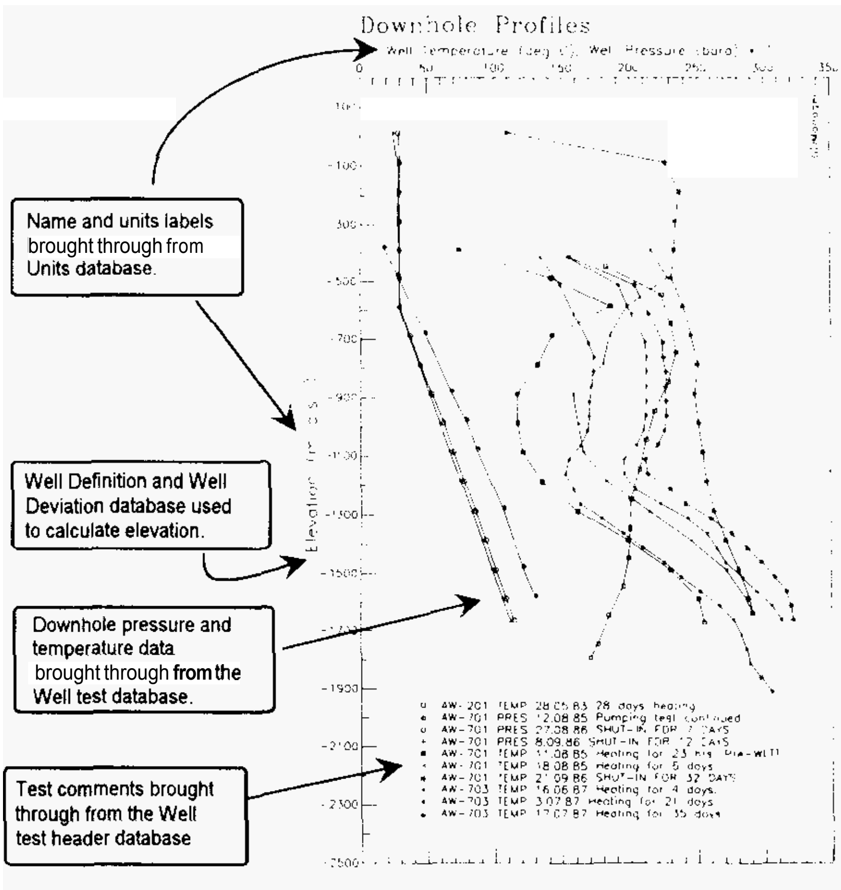

Figure **2** The database structure is **a** representation of the **real** physical objects in a system **as well as** providing for abstract features such **as well** tests.

These data relationships are built into the system in two ways:.

- Paradox tools which provide table links in Forms and Reports,
- standard GDManager routines where the developer specifies relationships between tables for various processes.

The database provides data storage for the whole progression from measured raw data, to observed (or calculated) physical parameters of the resource, to models that summarise aspects of the system.

Observed physical parameters of the system are (arbitrarily) defined as the core data for any module of GDManager. To obtain this data, raw measurements may have been entered and processed using some basic calculations. This data is the end product of the main data entry process. It can then be presented visually or later interpreted in a variety of ways, such as the calculation of more abstract parameters or models.

This can be illustrated by two examples:

1 For a resistivity survey, voltages and currents measured in the field (raw data) are used to calculate apparent resistivities which are the observed physical parameters of interest (core data) for this module. Many apparent resistivities can then be used to produce apparent resistivity maps, but also the data from each sounding can be collectively modelled one dimensionally for an interpretation of resistivity at depth, and each one dimensional model can be combined to produce a cohesive 3D model of the survey area.

2 Transient pressure tests can be used as tools for assessing local reservoir properties. From such tests estimates can be made of parameters like wellbore skin coefficients or permeability thickness values. However, the primary data entry process of the (core) pressure data may require considerable calculation, particularly with conversion of time parameters.

4. Data Collection

The concept of having the core data type (observed parameters) as the end product of the data entry process. allows data to be collected from many sources and integrated into the main table(s) of each module. Expanding the variety of methods for entering data makes the system adaptable to the wide range of measurement techniques employed in different geothermal projects while allowing the application of standard data presentation and analysis systems to the integrated data.

Initially, GDManager included only direct data entry for the observed parameters. It was quickly observed, however, that more assistance for entering actual field data was needed if the system was to streamline the data entry process.

In some modules, there are relatively few raw data options and a one-to-one correspondence exists with the calculated parameters (for example, apparent resistivity values from volts, amps in DC resistivity surveys). In such cases, the core data table can be expanded with extra fields to include raw' measurements in the same table

In some situations, it has not been necessary to store the actual raw data as the calculations required are very simple and the variety of data input is best handled by specific data entry routines In the Reservoir Engineering module for example, decimal hour times (relative or absolute) are converted to the standard absolute hh:mm:ss format upon data entry. and time can even be entered in the abstract format of inches on a Kuster downhole tool chart.

As GDManager is regularly being applied to more genthermial projects, the need for handling a wider range of data sources is being addressed by providing more flexible generic data entry routines as well as building up some 'front-end' databases and calculations.

The need for 'front-end' databases has been seen most clearly in the area of steamfield production data management. Production well flowrate data can come from a wide range of sources such as weirbox levels, lip pressures. orifice plates, chemical tracers. well pressures

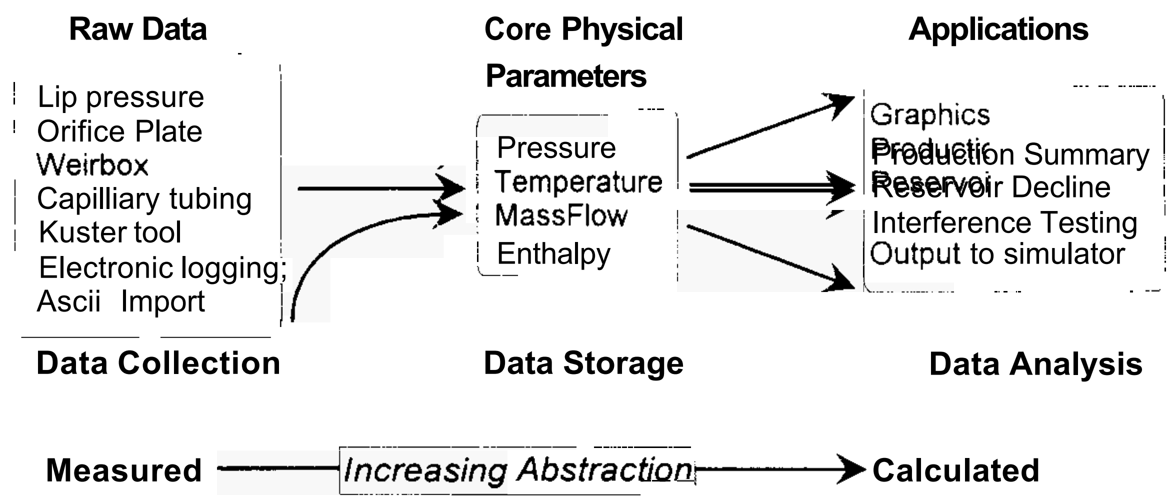

Figure 3 The new Steamfield Production module: An example of how a variety of raw data sources may feed data that is effectively all of a common type into the core data table for that module.

and even assumed enthalpies (Figure 3). Some of the calculations to calculate flowate are not simple. Recent development work has involved providing for this type of data collection and has resulted in the inclusion of more raw data tables in the database.

A flexible generic data importation system has been developed to assist with entering data from other computer files and is allowing direct entry of data from electronic sources such as logging tools. As for all manual data entry, the standard GDManager units conversion system allows for handling importation of data in any mix of measurement units. All imported data has normal data consistency and relationship checking performed.

5. MANAGING AND BUILDING GDMANAGER

Because oi the complexity of the physical geothermal systems which it is encapsulating, GDManager is a complex system with an enormous number of component objects, including:

- 100 core data storage tables containing 580 fields
- 50 forms for data entry aiid 70 reports.
- 1000 menu choices in 180 menus
- 130 graphical presentation formats
- 25 standard data management procedures.
- 230 special calculations which are relevant to their particular nodules.
- 14 tools external to the Paradox systeni designed for enhanced capabilities such as importing and modelling.

Consequently, it was essential to develop systems to effectively manage these components. The database structures are themselves created and administered from within a separate database development system. Three system tables contain table definitions (entities), database fields (attributes), and the relationship between them which defines the database structure.

In addition to this *database* ***table generaror,*** we have developed an *application* ***generaror*** to build the complete menu system for GDManager This system links objects such as data entry forms, reports, graphs, and calculation routines with a code library of standard data management routines as required within each module (Figure 4) The standard code library of common routines provides a high level oi code reusability across all modules of the system Again this allows for ease of dcvelopment, upgrading, aiid a high level of reliability

The power ofthc application generator lies in its ability to track where each object is used within the system making upgrading a reliable and less time consuming process If any standard routine is upgraded, the entire menu system can be rebuilt in minutes.

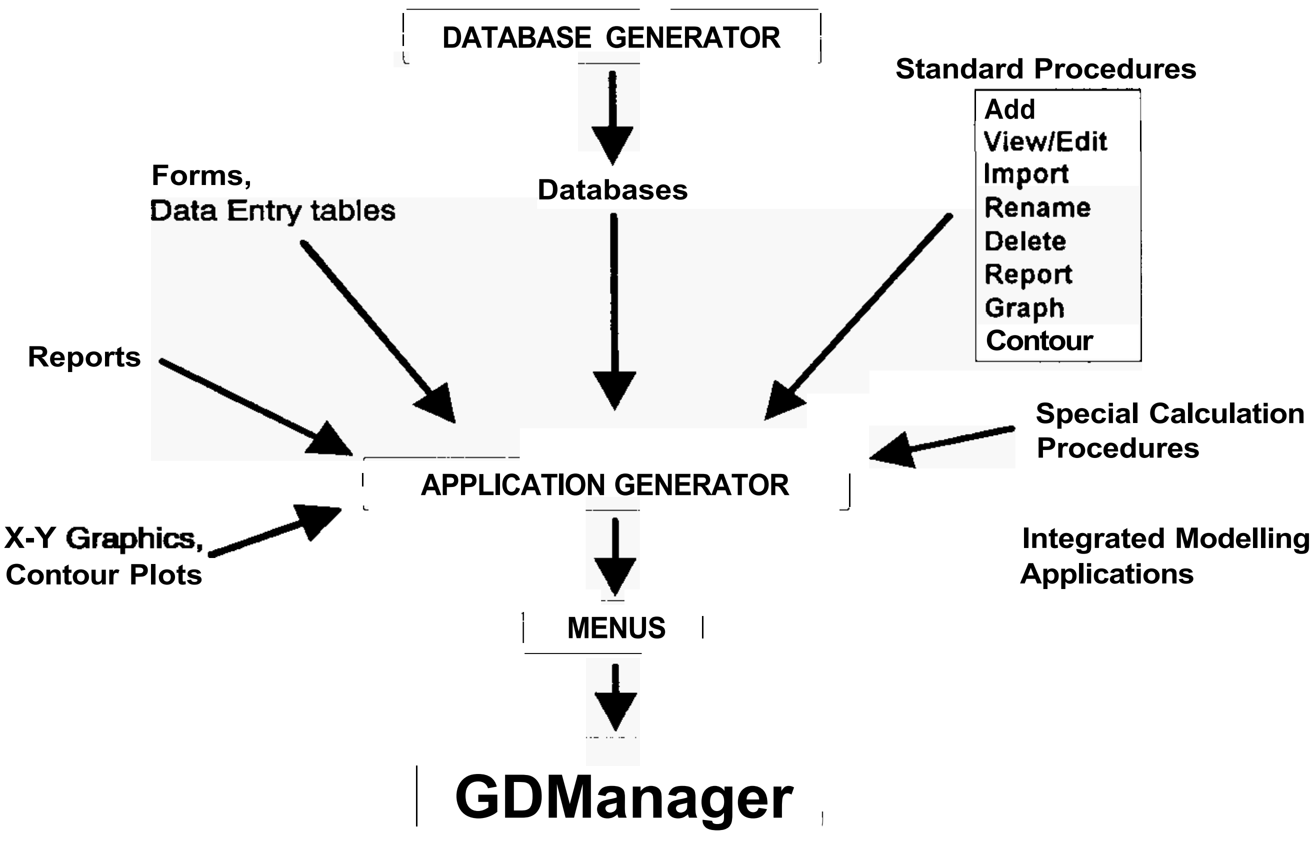

Figure 4 **The custom designed Application Generator links the forms, reports, graphics and special calculations with a standard library of functions. lo build the menu system.**

6. Future Devewpment Options

The present GDManager system has been developed using the MS-DOS version of Paradox as the database tool. As software platforms evolve and users change their computing systems to meet other constraints and needs, it will eventually become necessary to convert GDManager to some other software platform (client-server systems or Microsoft Windows, for example).

Although forms, reports and the standard GDManager library of functions are specific to this version of Paradox, the database itself can already he accessed **by** other tools or could be very simply converted to any other database standard. In any case, the very structured tools that have been built to manage the system would assist conversion of GDManager to other platforms in an orderly fashion.

The database itself will probably remain essentially unchanged in any change of software platform and thereby can provide a mechanism for a gradual migration betueen systems. New data entry or data presentation and analysis tools can be developed using Paradox for Windows, for example, and can access the database simultaneously with the existing GDManager system. This parallels the trend in most modern software systems lo allow data access by **a** range of application tools, allowing the user to choose the tool most suitable for their task.

The objective of GDManager is to provide a broad range of tools for entering, processing and presenting almost all data from a geothermal system. While this enables a geothermal developer or resource manager to quickly implement an orderly corporate-wide data managemmt system, they may need some extra flexibility not provided within the GDManager system. Having a well structured and integrated database allows users to apply their own tools **as** necessary to this data, providing a high level of flexibility. Users can now utilise some standard GDManager library functions in their o w utilities to make data access easier. In the future, new soflware technology such as Dynamic Link Libraries and Object Linking promise exciting options for more direct user flexibility within GDManager.

7. References

Anderson, E.B., Ussher, G.N.H. (1992). GDManager, An Integrated Geothermal Data Management System. Proceedings of 13th NZ Geothermal Workshop, University of Aucklnnd Geotherinel Institule, pp121-126

Geothermal Energy New Zealaiid Ltd. (1994) Geothermal Data Management and Wellbore Simulation Software, GRC Bulletin Computer Programs for Geothermal Developers, Part I, Geothermal Resources Council, p123-127