BIODIVERSITY DATA MANAGEMENT
(Document 3)

GUIDELINES
for
INFORMATION MANAGEMENT

in the context of the
Convention on Biological Diversity

The United Nations Environment Programme

in association with
World Conservation Monitoring Centre

January 1995

A/N 9879

95/1

BIODIVERSITY DATA MANAGEMENT
(Document 3)

GUIDELINES
for
INFORMATION MANAGEMENT

in the context of the
Convention on Biological Diversity

The United Nations Environment Programme

UNEP

in association with
World Conservation Monitoring Centre

WORLD CONSERVATION
MONITORING CENTRE

January 1995

UNEP - United Nations Environment Programme is a secretariat within the United Nations which has been charged with the responsibility of working with governments to promote environmentally sound forms of development, and to co-ordinate global action for development without destruction of the environment.

The **World Conservation Monitoring Centre,** based in Cambridge, UK. is a joint-venture between the three partners in the *World Conservation Strategy* and its successor *Caring For The Earth*: IUCN - The World Conservation Union, UNEP - United Nations Environment Programme, and WWF - World Wide Fund for Nature. The Centre provides information services on the conservation and sustainable use of species and ecosystems and supports others in the development of their own information systems.

For additonal copies of this document
or further information contact:

United Nations Environment Programme
PO Box 30552
Nairobi
Kenya

ACKNOWLEDGMENTS

1. This document is one of a series of four researched and compiled by the World Conservation Monitoring Centre, Cambridge UK with 80% funding from the Global Environment Facility (GEF) through the United Nations Environment Programme (UNEP), Project GF/0301-94-40 (GF/0301-94-06). The need for the development of a package of tools and materials to support national information management for the Convention was identified and the project promulgated by Mark Collins (Director, WCMC) and Robin Pellew (former Director of WCMC).

2. Principal authors were Ian Crain, Gareth Lloyd, and Gwynneth Martin. Contributions and critical review were received from a number of WCMC staff and consultants, including John Busby, Don Gordon, Jeremy Harrison, and Jake Reynolds. The document has benefited, as well, from review and comment from NGOs, UNEP, and experts in a number of countries who participated in a consultation meeting hosted by UNEP in Nairobi in October, 1994. Graphical concepts were developed by Ian Crain and Gareth Lloyd, and executed by Ian Kime of "Constructive Solutions". Document organisation, integration and input was by Laura Battlebury. Ian Crain was the project manager and responsible for overall design and editing.

TABLE OF CONTENTS

1 **INTRODUCTION** 1
1.1 Background 1
1.2 Structure of the Guidelines 3

2 **INFORMATION SYSTEMS DESIGN AND DEVELOPMENT** 5
2.1 The Framework 5
2.2 Custodianship 6
2.3 System Development Methodologies 8
2.4 Structured Development Life Cycle 9
2.4.1 Overview 9
2.4.2 Project Initiation 9
2.4.3 User Requirements 11
2.4.4 System Design 12
2.4.5 Development 13
2.4.6 Implementation 14
2.4.7 Operation 14
2.4.8 Examples 15
2.5 Prototyping Methodologies 16
2.6 Combined Methodologies 18

3 **DATABASE DESIGN AND DEVELOPMENT** 21
3.1 When is a Database Management System Needed? 21
3.2 Database Development Methodology 22
3.3 Logical Database Design 24
3.3.1 Overview 24
3.3.2 Data Modelling 25
3.3.3 Access Analysis 27
3.3.4 Software and Hardware Evaluation and Selection 27
3.3.5 Architecture Selection 27
3.3.6 Making the Final Choice 29
3.4 Physical Database Design 30
3.5 Database Implementation and Installation 31
3.5.1 Overview 31
3.5.2 Populating the Database 32
3.5.3 Relational Data Retrieval 32
3.6 Special Considerations for Biodiversity Databases 33
3.6.1 Synonyms and Equivalent Terms 33
3.6.2 Integration of Text with Conventional DBMS 33
3.6.3 Integration of Spatial Data with Conventional DBMS 34
3.6.4 Handling Hierarchical Taxonomic and Classification Data 35

4 DATA ANALYSIS AND MODELLING 37
4.1 Data and the CBD 37
4.1.1 Overview 37
4.1.2 Specific Categories 38
4.1.3 Data Form and Media 39
4.2 Approaches to Data Analysis 40
4.2.1 Overview 40
4.2.2 Packages 40
4.2.3 Custom Program Design 41
4.2.4 Modelling 42

5 QUALITY MANAGEMENT 45
5.1 Introduction 45
5.2 Institutional Quality Standards 45
5.3 Dataset Quality Audits 47
5.4 Operational and Data Security 49

6 HUMAN RESOURCE ISSUES 51
6.1 Current Information Technology Environment 51
6.2 Scarcity of Expertise 51
6.3 Training and Development 52
6.4 Professional and Vocational Standards 52
6.5 Job Descriptions 53

7 REFERENCES 55

ANNEXES

1: LIST OF ACRONYMS & ABBREVIATIONS 57

1 INTRODUCTION

1.1 Background

The ***Convention on Biological Diversity*** (CBD) was signed at the United Nations Conference on Environment and Development in Rio de Janeiro in June 1992 by 154 nations and subsequently came into force in November 1993. Article 7 of the Convention is concerned with identification and monitoring activities to support Articles 8 to 10 (*in-situ* conservation, *ex-situ* conservation and sustainable use of components of biological diversity). Contracting parties are required to identify components of biological diversity important for its conservation and sustainable use (Article 7a); to identify activities likely to have adverse impacts (Article 7c); and to monitor the status of both components and threats (Articles 7b and 7c). Specifically Article 7d identifies the requirement to:

> *"Maintain and organise, by any mechanism, data derived from identification and monitoring activities".*

Having recognised this clearly identified need for management of data in support of national planning related to biodiversity, the United Nations Environment Program (UNEP), in collaboration with the World Conservation Monitoring Centre (WCMC), designed and submitted to the Global Environment Facility (GEF), a project proposal entitled Biodiversity Data Management Capacitation in Developing Countries and Networking Biodiversity Information (BDM). This proposal was endorsed and subsequently a sub-project was established between UNEP and WCMC for Development of Supporting Materials for Biodiversity Data Management and Exchange.

The sub-project has produced an interlinked package of resource supporting materials to assist in national capacity building. There are four principal components of this package:

Document 1. Data Flow Model
- to identify in a formal structure the relationships between components of biodiversity data, from acquisition through to use in national strategy development, planning, and monitoring for implementation of the CBD.

Document 2. Guidelines for a National Institutional Survey
- to provide guidance to countries in the conduct of a survey and assessment of the capacity of existing national institutions to support biodiversity information management.

Document 3. Guidelines for Information Management
(This Document)

Document 4. Resource Inventory
- the core output of the project; a collection of reference directories, guidelines, and standards relating to biodiversity information management.

These Guidelines are intended to facilitate the development of national capacity for information management and exchange as required under the CBD. They are based on

conventional established principles of information system design, development and management which are applicable to all fields of endeavour. These principles are then elaborated in the context of the nature of biodiversity information and particularly the need to manage this information for the development of national biodiversity strategies and action plans, and for assessing the effectiveness of such actions.

It is also important to note, that while the use of computer systems is likely to be warranted, the same principles apply to instituting manually-based biodiversity information management practices. However, the scale of the problem is such that over the last decade, many institutions holding relevant information have decided to acquired computer technology to manage their data in consistent and efficient ways.

It is important to read these guidelines in the context of the companion document the *Data Flow Model* (Document 1) which provides the framework for conceptual national biodiversity information system. Figures 1.1 and 1.2 which follow are extracted from that document and show the conceptual overview of the CBD process and the first level of a formal dataflow model.

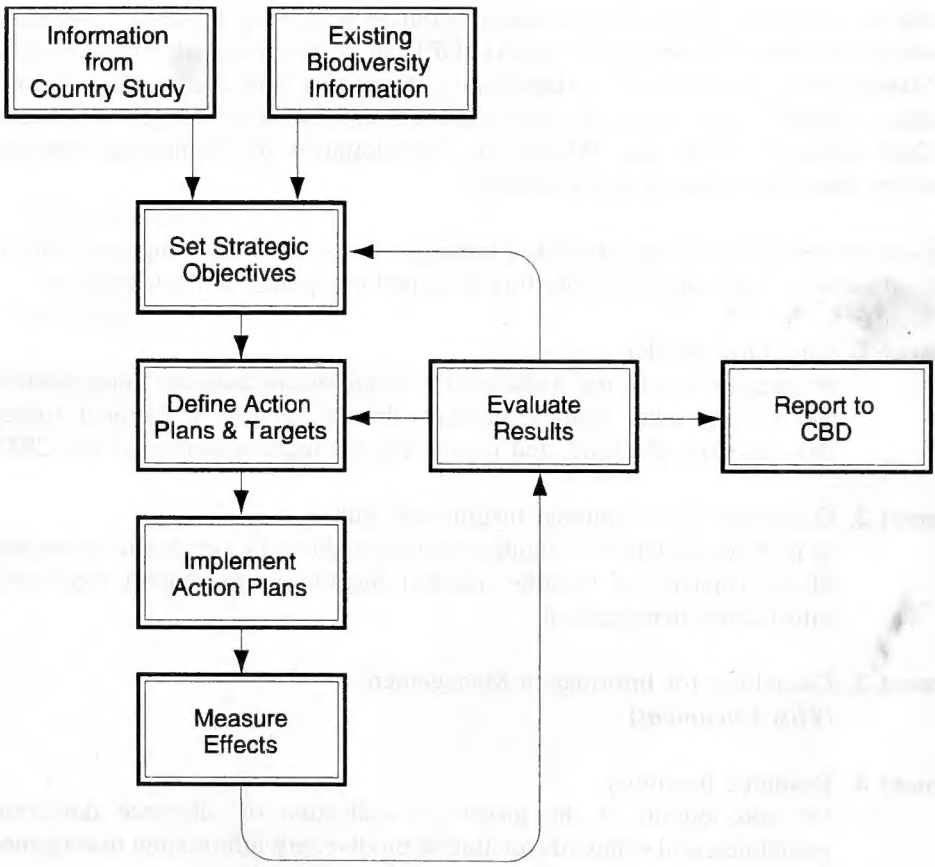

Figure 1.1: Overview of National Activities in Support of the CBD

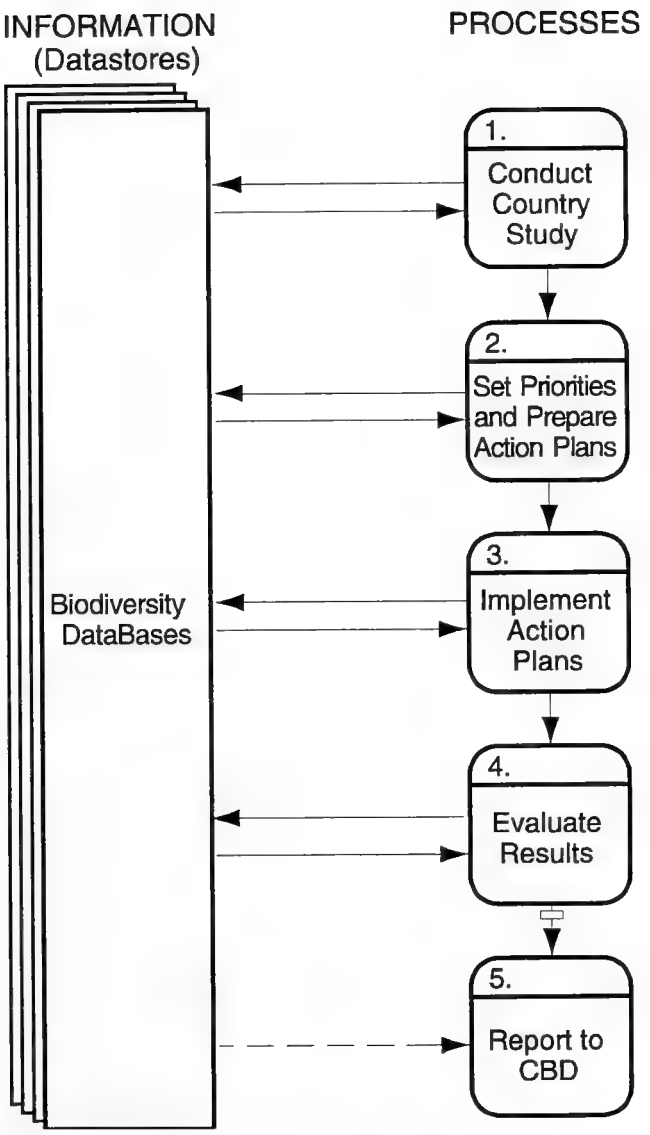

Figure 1.2: CBD Data Flow Diagram, Level 1

1.2 Structure of the Guidelines

The Guidelines are intended to be a practical "toolbox" which will be of benefit to various levels of information system development and at different times in the course of responding to the Convention. Not all sections will be applicable for all situations. There is no one single "correct" way to achieve the desired results, so that alternative methods are indicated, and in all cases the approach must be tailored to the conditions and resources of the country. Additional choices, and more detailed examples can be found through the companion document, *Resource Inventory* (Document 4).

An "information system" consists of a structured set of processes (and associated people and equipment) for converting data into information, **and** for presenting it in forms which are useful for communication and decision making. The modern information system may utilise computers in some of the processes, as well as for data storage, but this is by no means essential. The principles of information management remain the say whether or not computers are used - the need for data to **flow** between system components, the need for well defined **processes** (of analysis and integration), the need to hold and maintain information, and the need to present or **output** the information in useful forms. Some or all of the processes may be manual, requiring specialised knowledge and interpretation (see Document 1).

Biodiversity information systems consist of:

People
This includes the technical staff who operate the computer (or manual) systems, experts in information interpretation (biologists, ecologists, resource economists), specialised technical experts who develop information systems and computer networks, and the users (strategy developers and decision makers at various levels).

Information
The biodiversity information is stored in paper files, on maps, in specimen collections and in computer files and databases.

Hardware and Software
This includes simple and advanced computers, data capture and entry devices, printers and plotters, general purpose software such as spreadsheets and database management systems, and more specialised biodiversity analysis and modelling software.

Procedures (or Processes)
Established, documented processes define what the people do with the hardware an d software to manage, protect and manipulate the information to achieve the outputs required to respond to the Convention.

These Guidelines address considerations of all these components, starting with the very technical issues of overall information systems design and development (Section 2), and the more specific technology of database design (Section 3). Section 4 addresses the nature, availability and selection of biodiversity data analysis and modelling tools, and the final two sections deal with the non-technical issues of quality management and human resources.

Not all parts of the Guidelines will be applicable in all situations. The intent is to provide reference material of wide value from which to select according to national needs.

2 INFORMATION SYSTEMS DESIGN AND DEVELOPMENT

2.1 The Framework

It is extremely important that information systems are developed and implemented to meet the needs of prospective **users**. In the case of biodiversity information management, the users will include scientific specialists and technicians who collect, input, and manage biodiversity data, and interpret them data to give useful information. However, in order to ensure that system outputs are effectively applied to conservation planning, it is important to view officials working for environment-related agencies, and political decision makers, as potential users also. These two major groups occupy different ends of the user spectrum, being closest and farthest from direct information handling respectively. The resulting gap is often filled by a third group of users, the NGOs. These organisations can play a critical role in affecting government policy in favour of conservation, and should be empowered to do this with up to date biodiversity information.

For very simple information systems, the chief users of the system may also be its "developers", ie those implementing the necessary data management procedures, and where needed, the necessary computer programs. However, to develop and implement complex information systems, it is often necessary to employ a team of system developers whose job it is to build the system (not the data) in such a way that it meets user needs efficiently, and can operate smoothly and safely into the future. The development team may consist of in-house staff with experience in information systems, or external consultants assembled for the task period. However, it is common to engage under contract at least some specialised computer staff.

In the following paragraphs, the respective roles of the users and developers is elaborated, noting the importance of good communication and interaction between them at all stages. A commonly used management framework for information system development, suitable for the multidisciplinary approach required for biodiversity, comprises the following essential bodies:

Steering Committee

This is composed of high-level management officials representing the institutions participating in the development and use of the information system. Major data providers, (eg university departments, census bureaux) and users (planning units, NGOs, resource management agencies) are likely to be included. The Steering Committee provides general guidance to the development project and 'signs-off' at each major phase of development. The Committee also has responsibility for allocation of expenditure via an appointed funding manager or sub-committee. The overall purpose of the Steering Committee is to ensure that all participating institutions feel involved and responsible for the project.

Development Team

This may be entirely made up of contracted information systems experts or a combination of internal and external human resources (internal staff should be seconded from current duties to fully participate). The day-to-day activities of the development team should be directed by a project manager responsible in general to the Steering Committee, and in particular to the funding manager. On completion of the information system the Development Team is condensed to a core set of maintenance staff.

User Advisory Group
This group should have representation from all the important user communities (scientific, technical, policy, decision making). The group is responsible for providing the Development Team with the subject matter, specification, and assistance they require, and for communicating information requirements and progress back to users. This extremely important function is designed to make sure that the ultimate users feel involved in the development process, and assume a sense of ownership of the systems. The User Advisory Group will also be very actively involved in system testing, following up with advice to the Steering Committee on acceptance or required modifications at key decision points. After development is completed, the User Advisory Group can take on a liaison role between participating institutions to ensure continued effectiveness into the future.

Guideline 2.1
National biodiversity information system projects should attempt to build an operational framework consisting of a high-level Steering Committee, an expertly staffed Development Team, and a wide-ranging User Advisory Group.

Before discussing methodologies for developing biodiversity information systems, it is worth elaborating on the issue of data flow between agencies managing biodiversity information. In particular, the rights and duties of those agencies within the wider context of a cooperative information network.

2.2 Custodianship

A key to good management of biodiversity data is to make sure that data is always held by the organisation best placed to ensure quality. "Custodianship" provides a framework under which responsibility for a dataset can be assigned to and accepted by this organisation. The responsibilities of custodianship encompass data acquisition, management, and documentation, as well as determining under what conditions a dataset may be accessed and used.

Responsibility for each dataset must be clear and unambiguous. One agency must be the designated custodian for the dataset as a whole, although entities within the dataset could be maintained by others. An example would be a species-site dataset held in a protected areas management agency, where the species authority files within that dataset could be maintained by national collection agencies such as museums and herbaria.

Custodianship needs to be managed at multiple levels within a country. At the national level, responsibility for broad themes should be allocated among the various government departments, eg topographic infrastructure such as national boundaries, topography, roads, rivers, etc, to the central mapping agency, and so on. These agencies should build datasets to support decision making at that level. Datasets to support regional and local activities should be built by agencies, or regional offices of national agencies, at or close to those levels. All these activities need to be coordinated at various levels to ensure standards are adhered to, overlap and duplication are minimised, and local-scale datasets can be smoothly integrated and generalised to support national-level decision making.

The datasets required by biodiversity information systems are complex. The establishment and maintenance of these datasets therefore requires specialist knowledge. This also applies to

documenting the data and advice to clients as to their fitness for various potential uses.

Decision makers and other end users are seldom able to use raw, unprocessed data. They require data relevant to particular issues, often integrated with other data and assessed by experts, before being summarised into information products. The responsibility of custodianship requires agencies to advise users (particularly those without detailed technical knowledge) on what assessment tools are most appropriate for the data concerned.

The most appropriate custodian for a dataset is likely to be the agency which meets one or more of the following criteria:

- has sole statutory responsibility for capture and maintenance of the data
- normally is the first to record changes to the data
- is the most competent to capture and/or maintain those data
- has the confidence of users that it will continue to meet its commitments to data collection and maintenance.

The responsibilities of a custodian include the following:

- to define and maintain quality standards
- to organise the building of information systems
- to keep the dataset up to date
- to ensure the continued integrity of the dataset
- to ensure appropriate access to the dataset
- to maintain documentation on the dataset
- to advise on appropriate uses of the dataset.

The concept of custodianship can be very useful when attempting to build cooperative networks of information systems, whether the linkages between the partners are electronic or informal. An important principle of the scheme is that all datasets on the network are, in theory, accessible by all partners. Designated custodians, however, have responsibility for collection and maintenance of the data and the sole right to up-date and correct it. Varying conditions may be attached to data on a network. For example, data may be used for government decision making, public information or research purposes, but not for any commercial purposes, at least without specific permission.

Custodians are also responsible for management of licensing agreements, which can become extremely complex. Every effort should be made to develop simple generic licences for data access and use within each jurisdiction. 'Memoranda of Understanding', and similar high-level mechanisms to encourage the free flow of information between agencies, should be negotiated. Successful biodiversity information management, after all, requires ready access to many sources of data from a wide variety of institutions. Thus there should be an absolute minimum of administrative, cost and other impediments to the flow of data, consistent with the protection of copyright, intellectual property, and other legitimate custodian rights. Obstacles to the free flow of information will inevitably inhibit responsible decision making and sound management of biodiversity.

Guideline 2.2
Each biodiversity dataset should have an identified custodian responsible for keeping it up to date, ensuring data quality, and controlling access.

Guideline 2.3
Consistent with protection of custodians' legitimate interests in the data, there should be a minimum of administrative, cost, and other impediments to the free flow of data among agencies.

2.3 System Development Methodologies

With increasing use of computers for information management, methodologies for the development of information systems have steadily been evolved. These originally arose to address the problem that development costs (resources and time) often considerably exceeded original estimates. In the late 1960s and early '70s, a classic project life-cycle became accepted to regulate information system projects. Given the level of technology at that time, (eg mainframe architectures, punch card processing, languages such as FORTRAN and COBOL) such projects tended to be in the hands of computer specialists. When eventually delivered, the system was therefore subject to criticisms such as "not what was wanted", "incomplete" and "unworkable". Two factors contributed to this:

- long development periods during which user requirements would change
- difficulties in specifying user requirements in complete and unambiguous ways.

In the early 1970s the first of these was partly addressed through concepts such as structured programming and, a little later, structured analysis. Tools were developed to support these techniques, laying the ground work for many of the tools we use today which relieve much of the burden of programming. Structured programming tools have become increasingly important as the cost of human resources has increased, making development productivity a key factor in overall project costs.

A breed of system development methodologies grew up around the structured programming paradigm. Collectively, these are referred to as the Structured Development Life Cycle approach, in which development is carried out in a series of **structured** phases. Different variants of the life cycle approach are advocated in different countries and organisations, some of which are accepted as "standards" in industry and government.

During the same period, information technology was undergoing rapid change. High power personal computers became available to many users, complete with sophisticated techniques for linking them into networks. Project development became centred around the 'desk-top', with powerful "high-level" languages being developed for accelerated system design. With this revolution came the introduction of **prototyping** tools capable of building working models of end-user systems, (ie the finished product) within a short time frame. This led to the introduction of an alternative class of development methodologies, in which prototype systems are modified on the basis of feedback from prospective users.

The ideas behind the structured development life cycle and prototyping methodologies are described in the following sections. However, in a practical situation the optimum approach

is generally accepted to result from a judicious mix of the two alternatives, the balance between the two being determined on a project-by-project basis (see Section 2.6).

2.4 Structured Development Life Cycle

2.4.1 Overview

A well-established class of methodologies uses the Systems Development Life Cycle approach, in which the development is carried out in a series of structured incremental phases. Although many variants of the life cycle are advocated in different locations, all share the following basic features:

- there are distinct phases (usually between five and eight) moving from introducing the concept of the system to full operation
- specific defined products result from each phase
- the phases are carried out in sequence, building on the products established previously
- a decision as to whether or not the development should proceed is associated with the completion of each phase
- iteration is ideally only to revise and refine products from the preceding phase, ie not to go back more than one phase.

Figure 2.1 shows an example of such a structured systems development life cycle. It has six phases, each of which is described in the following sections with indications of the associated activities, products and decisions, and discussion of aspects of particular relevance to development of biodiversity information management systems. Diagrams such as these lead to the alternative name for the methodology - the "waterfall" approach.

2.4.2 Project Initiation

Description

The need for a system may be raised for a variety of reasons, and at a variety of levels in an organisation. The project initiation phase formalises the identification of the needs, and examines and evaluates the overall feasibility of being able to develop a system to meet those needs. This involves estimation, at a very general level, of the development costs and the time it will take, plus the benefits which may result. These give the basis for an overall cost/benefit or "return-on-investment" decision, ie whether the benefits justify the expenditures. If the outcome is positive then the development project is initiated.

Activities

Both developers and users are involved in this phase. Interaction is needed to jointly describe the needs and benefits of the system, and to quantify the latter. General investigation of available technology is carried out to determine approximate costs, and to arrive at estimates of resources and time required.

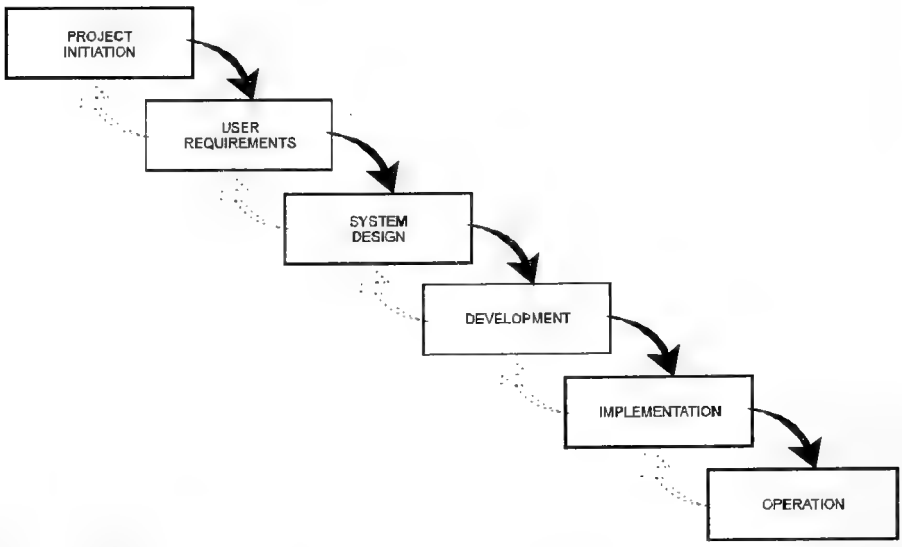

Figure 2.1: Structured Development Life Cycle Methodology

Products

The output of this phase is a project initiation document which will include a project workplan and estimated costs. This varies in size and formality with the size of system proposed and investment required, ie it may range from one page to something much more substantive. As complexity increases, it may be broken down into smaller, more manageable, projects. In many organisations this is the decision point for a long-term commitment of resources, although the actual numbers are by no means exact and will be refined as the project proceeds.

In this phase it is conventional to include a "cost-benefit analysis" of the proposed system. When implementing an industrial information system, the benefits of the system can be estimated in terms of reduced costs, increased revenues, improved delivery times of products etc. This allows a return-on-investment period to be calculated, and thus a decision on whether or not to invest in improved information management. In the case of biodiversity, estimating benefits is very difficult. Section C of the Technical Annex to *The Guidelines for Country Studies on Biological Diversity* (UNEP, 1993) states:

> *"Countries will find that their efforts to measure the value of biological resources and diversity are hampered by tremendous uncertainty. There is uncertainty regarding biological measures of the qualities, quantities, diversity and interactions of biological resources. There is uncertainty of the various goods and services that flow to us from these resources, or that may flow to us in the future. There is also uncertainty about the values members of our society place upon the flows of these goods and services and the values that future generations may place upon them. There is uncertainty about how human actions may impact biological resources and diversity and their associated goods and services, but we face the very real risk that the impacts of our actions may be irreversible. This is clearly the case for extinction of a species due to unsustainable use or disruption of habitat".*

A strict cost-benefit analysis for the development of a national biodiversity information system may therefore not be possible. Section 8.2 of the aforementioned guidelines includes some suggestions for approaches to estimating the economic value of biological resources. Because a concrete calculation of return-on-investment is unlikely to be obtainable, the valuation of the information system must be made on a strategic basis, that is, the decision to proceed must be based on finding the minimum cost solution (most cost-effective) to achieve the identified strategic goals under the Convention.

2.4.3 User Requirements

Description
The process of establishing user requirements is of the greatest importance in information system design. Without a proper user needs survey, time can be wasted collecting the wrong kinds of data, and systems can be designed which fail to address key problems. Very many factors affect how to establish user needs, not least due to the inherent differences between information system projects.

In most circumstances, a user needs survey will result in a "functional specification" of the system following a process of consultation between developers and prospective users (see Document 4, Section 2.2). This document describes the functions and capabilities of the intended system, without specifying exactly how those functions will be realised in terms of hardware and software, equipment connectivity, and so on. The conclusions of the project initiation phase (which were of a general nature) may need to be revised in the light of the final draft of the functional specification, particularly with regard to the stated scope of the system. These revisions are important since, once accepted, the functional specification provides targets for the Development Team and serves as a basis for testing the final product (see Section 2.4.5). It also serves to confirm the costs and benefits defined in phase 1.

Activities
Both users and developers collaborate closely during this phase. Consultation may take the form of interviews and workshops, held regularly as successive layers of detail are discussed and documented. The essential purpose of the phase is to avoid any misunderstandings between developers and users resulting from:

- lack of explanation to users of how the information systems will help
- reluctance on the part of users to adopt new operational practices
- over-concentration on the technological aspects of the system by developers
- lack of participation of users in system development.

Throughout the consultative process, developers should emphasise that the functional specification is open to modification. Indeed, it is very likely that the final document will be considerably better phrased than its early counterparts, due to improvements in the preciseness and practicality of the contributions. Further, it may become apparent that some desired functions are more complex than anticipated, and others simpler, forcing major revision of the specification.

It is vitally important for developers to gain an insight into the operational practices of participating institutions, in order to gauge how best to address their information management needs. Likewise, it is important for prospective users to consider the potential of an information system with open minds, and guide the outcome by active contribution to its design. Richardson (1994) estimated that the assessment of user needs occupied about 80% of time taken to start up a major Australian project, the Environmental Resources Information Network (ERIN). To may sound surprisingly high, but in reality it is sensible given the importance of the decisions made.

Prime topics for discussion during the interviews and workshops are:

- data: national and international sources, format (compatibility), overlap amongst datasets, major gaps requiring filling
- processes: data standards, core validation procedures, field survey priorities and methodologies
- outputs: format/content of reports for differing target audiences, report frequency, graphic outputs, digital files to be generated
- institutional capability: information systems in use, existing data holdings, existing skills level and expertise of staff.

Products
As mentioned above, the product of this phase is referred to as a **functional specification**. It may include:

- clear definition of intended users
- precise information needs of these users
- inventory of key data holdings and information systems
- conceptual or logical data models for new databases (see Section 3.2)
- process and data flow diagrams (similar to those in Document 1)
- survey of capabilities of participating institutions
- description of desired information products.

It should also indicate any areas of uncertainty in user requirements. For instance, a particular type of analysis may need new algorithms to be tested and proven before use.

2.4.4 System Design

Description
In the system design phase, the functional requirements resulting from the previous phase are translated into system specifications. The descriptions of data and processes become the basis for database structures where computer databases are required, and the definition of procedures and programs. The inter-relationships of the modules and the transfer of data between them are specified.

It is at this phase that decisions are taken on the overall system architecture, ie the type of hardware and software to be used. Where this is very specific, brand-names may be specified rather than generic categories. The organisational environment has a large impact on how this is handled. It may be that the system is best implemented on existing hardware and software; at the other extreme there may be nothing in place in which case acquisitions should at least be initiated in this phase; or implementation may be through additions to provide the required capacity and functionality, but the basic architecture is already determined by current practices.

Activities
The major part of the effort in this phase comes from the developers, but continued user involvement and input is essential. The developers are able to put together the system specifications in terms of data structures and program specifications but these should be verified by the users to ensure that they truly reflect their requirements. This may be done by means of discussions and walk-throughs, possibly using the method of prototyping (see Section 2.5). Very specific database design for component sectoral databases will also be undertaken (see Section 3 for expansion of this important activity).

This phase is also an opportunity to begin user training in establishing an understanding of the basic design premises and approach. If there are considerable new acquisitions of hardware and software involved, effort is needed to verify true functionality against vendor claims, and to develop tight specifications for contractual purposes.

Products
The output of this phase is a design specification which defines the development tasks to be undertaken with the way in which they fit together, and also provides the specification for hardware and software to be acquired. Estimation of costs now becomes much tighter as it can be done by totalling the development time and resources required for the low-level tasks specified. Similarly, acquisition costs can be confirmed by prices supplied by vendors. The decision to accept the specifications may therefore involve further analysis of cost-benefit, and review of budgetary commitments.

In terms of hardware, most biodiversity information systems will require only rather conventional equipment: PC level computers possibly networked, or somewhat larger machines such as Sun Workstations. Common peripheral equipment includes digitisers, scanners, and plotters to deal with map-based data, and CD-ROM drives to input large data sources.

Software requirements will include database management systems, statistical packages, graphics packages, word processing, desk-top mapping, GIS and specialised biodiversity modelling and analysis software (see Document 4, Section 3.2).

2.4.5 Development

Description
In the development phase, programs and procedures are developed and tested to meet the system specifications resulting from the previous phase. Database and file structures are

established in detail (see Section 3.4) and populated with test data to verify their operation.

Activities
Again the major part of the effort in this phase is from the developers who are coding, testing and documenting the system. This latter will produce both system and user level documentation and include such things as security measures and installation procedures. User involvement should be maintained through demonstrations of the various system functions as they are developed. This assists with testing and also prepares users for delivery of the system in the next phase. An implementation plan is also determined in close cooperation with users.

Products
The output of this phase is a functioning system which has been tested during development. It should meet the system specifications as decided earlier, since the decision to proceed is based on verifying that these specifications have been achieved.

2.4.6 Implementation

Description
Although testing has been done in the previous phase, that was primarily carried out by the system developers as the system was being built. The purpose of this phase is to establish operational procedures and for the users to fully test the system in an operational environment, and determine whether or not it is acceptable. Basically, the total functionality of the system is checked against the original user requirements. The test plan produced in the previous phase is the blue-print for how the testing is done to ensure, as far as possible, that all system capabilities are exercised. The testing will reveal problems at various levels - functionality may be missing, some may appear to be there but not work, some may appear to work but with incorrect results, and some may be functionally correct but their appearance is not as the user wants. All problems encountered need to be noted for correction and, when modifications and corrections are made, the affected parts of the system must be re-tested.

Activities
Both users and developers are involved heavily in this phase. The former are organising and carrying out the testing, and the developers are correcting, modifying and fine-tuning the system (and its documentation) to more fully meet user requirements.

Products
The decision that the system meets the specification and is acceptable to users draws the phase to a close. The output is a fully functioning system ready for operation.

2.4.7 Operation

Description
The system moves into the operational phase which will continue for its lifetime. It becomes part of the regular functions of the organisation. In this phase, there is also a maintenance requirement, as users will inevitably need changes to be made to the system. These could result from problems not discovered during the testing phase, or because a new requirement arises which can be met with a small enhancement to the system. As long as these needs are

"small" they can be met under the maintenance arrangements. However, a major new requirement would require the full development process to be initiated once again.

Activities
The users are now the major players with developers available to respond to maintenance requests.

Products
The outputs of this phase are those which come from the system itself, that is the benefits originally sought when initiating the project.

2.4.8 Examples

BG-BASE
An illustrative example of a computerised biodiversity information system is *BG-BASE* (for comparison with other systems see Document 4, Section 3.2.7), which was implemented following a request from IUCN to create a microcomputer database application for botanical gardens, both large and small, based on the International Transfer Format (ITF) for plant data (see Document 4, Section 5.8). A full account of the implementation process is given in Walter (1989), an excerpt of which is included below:

> *"From the beginning the design of* ***BG-BASE*** *has been a group effort; it has now involved more than 100 people from over 35 institutions.... For approximately two years, a group of five to eight of us (specialists) met over lunch nearly every week to plan and to discuss the design, and eventually to test and criticise the implementation. Ideas for new data fields, new files, and new reports were presented regularly for general discussion, resulting in some fairly heated debates. The heart of the system was always understood to be based on the International Transfer Format, but since this format specified only 36 fields, we had a great deal of fleshing out to do. As it currently stands,* ***BG-BASE*** *comprises 564 fields spread over 12 major files. In addition to these major files, there are another ten index files that allow the user to look up information in a wide variety of ways".*

The use of *BG-BASE* to manage plant conservation data at WCMC illustrates the importance of a flexible design. Although originally designed as a specimen based system, to manage botanic gardens' living collections, *BG-BASE* has proved a suitable application for use elsewhere. Due to the built-in full taxonomic hierarchy *BG-BASE* functions equally well as either a specimen, or taxon based system. Since this excerpt was written in 1989, the application has continued to expand, with a current total of 3,500 fields spread across c.150 files in the 53 installations at which it now runs - although no one installation maintains all these fields.

Biodiversity Data Bank (BDB)
As a second example, the "Biodiversity Data Bank" information system was established at Makerere University, Kampala, Uganda, in early 1993 (see Document 4, Section 3.2.7), although the task of collating Uganda's biodiversity information began long before using

manual techniques (even today the manual system is maintained for back-up and other purposes).

The specification of BDB was conceived by a small development team with extensive knowledge of the information requirements of the biodiversity sector in Uganda. Many key organisations were consulted including the Botany and Zoology Departments at Makerere University, the University Herbarium and Zoology Museum, Uganda National Parks, Forest Department, and several NGOs, such as IUCN and WWF.

The scope of the system is such that it can handle a wide variety of biodiversity data. This was considered important by users who requested a single system to manage their data, rather than a series of separate databases. The major data holdings include taxonomic names, species distribution records, protected area profiles, geo-political structures, geographic gazetteer, environmental bibliography, and useful contacts.

BDB was originally conceived as a means of organising the large number of data relating to Ugandan biodiversity, located inside and outside the country. From the outset an aim of the system was species mapping, and thus facilities were built into the system from the start to download distribution data in a form acceptable for desktop mapping programs. However, due to an urgent requirement to review the country's protected area system, pre-defined reports were also developed to list, and in some cases estimate population density, of species in protected areas. More complex analyses were also developed to predict species distributions on the basis of habitat usage.

With many of the procedures for managing biodiversity data continuously being revised, the design of BDB had to be flexible. Thus many of the "fixed choice" fields available during data entry can be populated with custom values. Examples include the set of designations for protected areas, or the set of lifeforms for flowering plants. The system is written in the FoxPro for Windows RBDMS, and therefore integrates well with other Windows-based programs, such as the mapping program, MapInfo.

2.5 Prototyping Methodologies

The classic structured development life cycle methodology described in detail in Section 2.4, while widely used and accepted, has some disadvantages. The process requires a great deal of interaction with users at the early phases to define the requirements, followed by a (potentially long) period where the developers convert the specification to reality, after which the users are involved once more to test the product. A gap in participation at any stage can erode confidence in the development team. Furthermore, user needs tend to evolve throughout the development period, making it essential to maintain dialogue on a regular basis. In industrial and administrative information systems, it is often relatively easy to **exactly** specify both the data requirements and the processes which are required to transform data into the required information outputs. However, with biodiversity information systems (and other scientific applications) the "process" part of the specification is more difficult, ie determining what types of analyses and models should be applied, and how to summarise the information in ways that are suitable to policy decision makers. This increases the risk that early decisions in the User Requirements phase may have to be revised.

These concerns have led to alternative, more interactive approaches to development which involve "prototyping". The principles of the approach are:

- to create a common ground between users and developers
- to have all parties understand the complexity of the processes being automated
- to build a small version of the system quickly (and inexpensively) so that users can discuss their needs, and so that developers can educate users on potential capabilities
- to permit a hands-on and iterative way of determining system requirements
- to allow changes to be incorporated easily during the development process
- to provide for continuous interaction between users and developers throughout the development process.

The principal advantages are that the developers can quickly verify that their understanding of the requirements is correct, and that problems are identified and corrected early in the process. Within this general framework, computer-aided prototyping tools are of basically two types:

The "Throw-away" Prototype
With this approach a simple mock-up or demonstration of the system or one of its parts in built, showing users how it might look like in practice. For example, how the input and output screens would be structured, the format of reports, and so on. However, all of these do not necessarily use real data, nor are real-world analyses normally tackled. The prototype is rather like an artist's sketch of a new building. (See Fig 2.2). The prototype is then modified, usually several times, until users are completely satisfied. It is then **discarded**, and the real system is built on the basis of the mock-up.

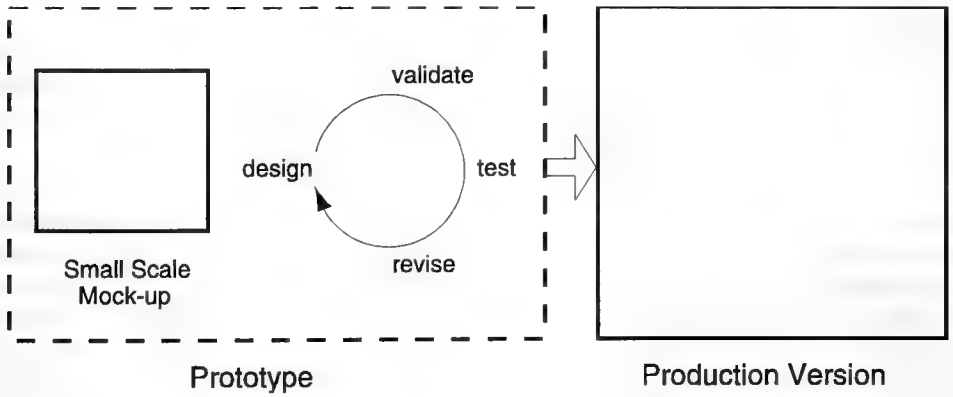

Figure 2.2: The Throw-away Prototype

The Evolutionary Prototype

The evolutionary prototype starts with a small part of the system, (eg one process) and takes it through the structured development life cycle. When this is tested, additional functions are added, each time increasing the core capabilities until the prototype is more complete, and finally evolves into the production system. The usual result of the evolutionary approach is a system which can more easily be adapted to future changes. (See Fig. 2.3)

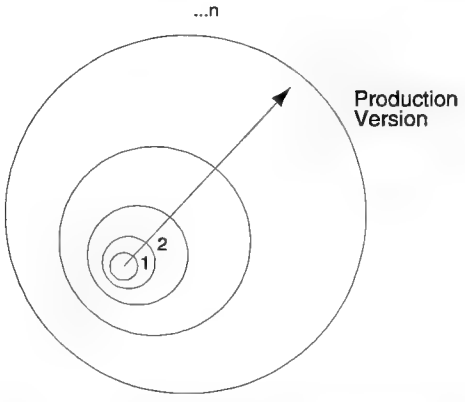

Figure 2.3: The Evolutionary Prototype

2.6 Combined Methodologies

The features of the two basic approaches may be combined in several ways. For instance, prototyping may be used at the system design phase of the life cycle methodology (Figure 2.4), or integrated as an additional phase in its own right (Figure 2.5). In this approach, changes to the system become an integral part of the development process and the system is modified as new needs arise during actual use of the system.

In practice it is found that the traditional "waterfall" approach is best for complex projects which can be defined well, (ie low uncertainty of user requirements), and that prototyping is best for simple projects of greater uncertainty. This would be typical of a biodiversity information system for a small specialised institution. A combination of the two is recommended where the system development project is both complex, and has high uncertainty, for instance in the design of a multi-sectoral system involving many institutions.

It should be stressed that the choice of a development methodology and related software tools is usually made by the Development Team, particularly the expert contractors who take responsible for product delivery. However, users should be aware of the options and principles to help select the contractors, and participate in the development process.

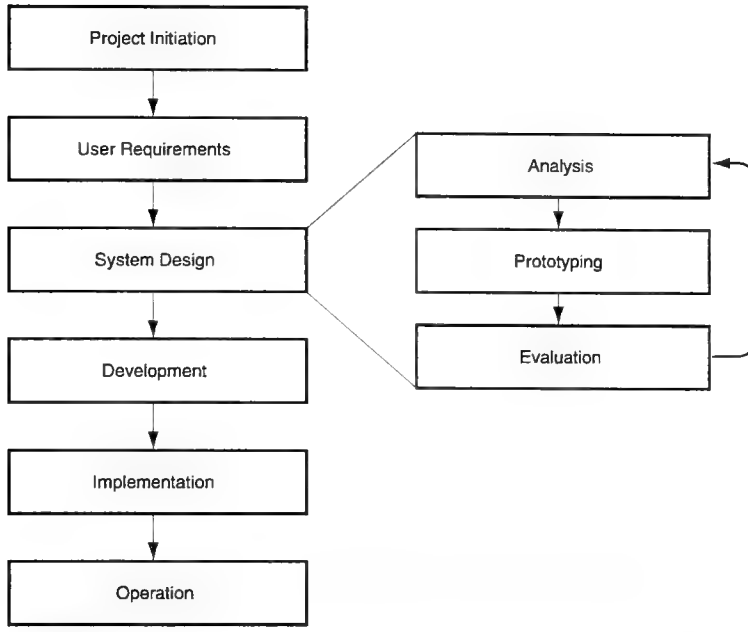

Figure 2.4: Prototyping in Design Phase

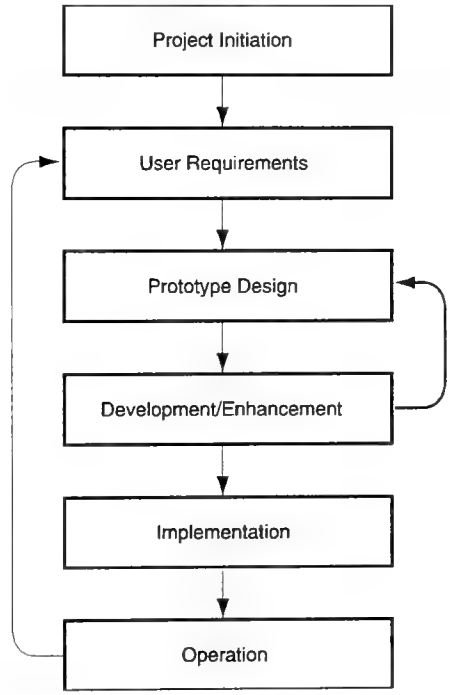

Figure 2.5: Integrated Prototyping

Guideline 2.4

In developing biodiversity information systems, consideration should be given to the principles of a structured development life cycle. Advantage should also be taken of prototyping tools, if available, where user needs are difficult to specify.

3 DATABASE DESIGN AND DEVELOPMENT

It is anticipated that national biodiversity information systems will include a variety of "datastores" relating to different sectors and institutions in the biodiversity field (see Document 1). Although it need not be automatically assumed, the volume of data contained in these datastores may eventually grow so large that computer database management systems (DBMS) will be necessary for effective management of the data. This section provides details of the design and development techniques necessary to build the database components of an information system. Such activities normally fall into "System Design" and "Development" phases as outlined in the previous section.

3.1 When is a Database Management System Needed?

The quantity of information available and accessible in the world is growing at an exponential rate. The subset of this information relevant to biodiversity is probably increasing more than any other thematic group. As a result it is often the case that the biodiversity information management needs of a particular country will be of sufficient size and complexity to justify the development of one or more computerised databases. To decide whether the more advanced technology of DBMS is justified, some of the questions which should be asked are:

- do the data contain relationships too complex for the capabilities of a spreadsheet or word processor?
- is the quantity of data too much for manual methods or word processing to efficiently handle?
- is it important that data from heterogenous sources can be integrated into a combined output?
- does the work follow predictable patterns (on going inventory, repeated queries, and so on)?
- is there a need for the data to be shared amongst more than one user in a single institution, or with other institutions?
- do the data require extensive searching and sorting?
- is extensive reporting of the data required?

If the answer is **no** to all these questions, then the use of a DBMS is not indicated. If the answer is **yes** to more than four questions, then the use of a DBMS should be considered. If the answer is **yes** to six or more questions, a DBMS is definitely needed.

Guideline 3.1
Methodically evaluate whether a computer database management system is needed before proceeding.

Databases require considerable effort to implement and use, but do confer numerous advantages as follows:

- the use of databases enforces a consistent means of entering and verifying information, removing much of the inaccuracy of manual record-keeping; a database can be designed so that entry mistakes are minimised by incorporation of automatic validation procedures

- the use of databases contributes to data quality assurance

- databases can handle large volumes of data

- databases can integrate and provide facilities for sharing of data from varied sources whilst still allowing each group of users to access and control their own data

- data can be accumulated in a database over a long period of time, longer than the tenure of an individual worker

- a database **systematically** stores information for retrieval, facilitating analysis which would be much more difficult with manual tabulation

- many reports can be produced from the same data

- databases can be used to make visible trends in the data.

3.2 Database Development Methodology

After a country has analysed its overall needs for biodiversity information management and a national biodiversity information system, it is quite likely that it will need a suite of sectoral **databases** for the various sectors. For example, the recording of natural habitats data will require different data structures to those needed for recording animal species data (see Document 1).

Once the sectoral needs are identified, it may be that some needs can be supported by database structures and procedures that have already been developed by another organisation or country. If such a database "shell" is available, then its suitability should be investigated.

For example, the BG-BASE application that WCMC uses to manage species information, was originally designed at the request of the Arnold Arboretum of Harvard University and IUCN to manage the living collections of botanical gardens, both large and small. Care was taken in the original design to keep the application as generic as possible in order to meet the needs of many gardens. The heart of the system was based, as requested by IUCN, on the International Transfer Format (ITF) for Botanic Gardens, a protocol created for exchanging information (see Document 4, Section 5.8). The foresight shown in placing emphasis on the ITF and in the need to keep the application generic has since been confirmed as this application (*BG-BASE*) has now been adopted by over 50 institutes world-wide, to manage living collections, conservation information, herbarium specimens, and as a teaching tool.

These institutions comprise botanic gardens, arboreta, horticultural societies, museums, universities and conservation monitoring centres.

Guideline 3.2

Consider adapting an existing database application to meet the needs where feasible.

In some cases, it will be necessary to design a completely new database, or at minimum add to or modify an existing application. Guidelines on the development of new databases are therefore provided.

There are many different approaches to designing databases. The version summarised here captures the essence of most database development methodologies. The process of identifying the project definition and user requirements is assumed to have already been carried out in the User Requirements phase of the structured information system development life cycle of Section 2. This section describes the processes which follow on from this work leading eventually to database implementation. The terminology for the following processes follows Daniels and Tate (1984).

Under this methodology, the design of a database is partitioned into two essential phases: the **logical** design phase which is independent of the computer hardware and software; and the **physical** design phase which indicates how the logical design will be implemented with the chosen hardware and software. These phases and their underlying activities are illustrated in Figure 3.1.

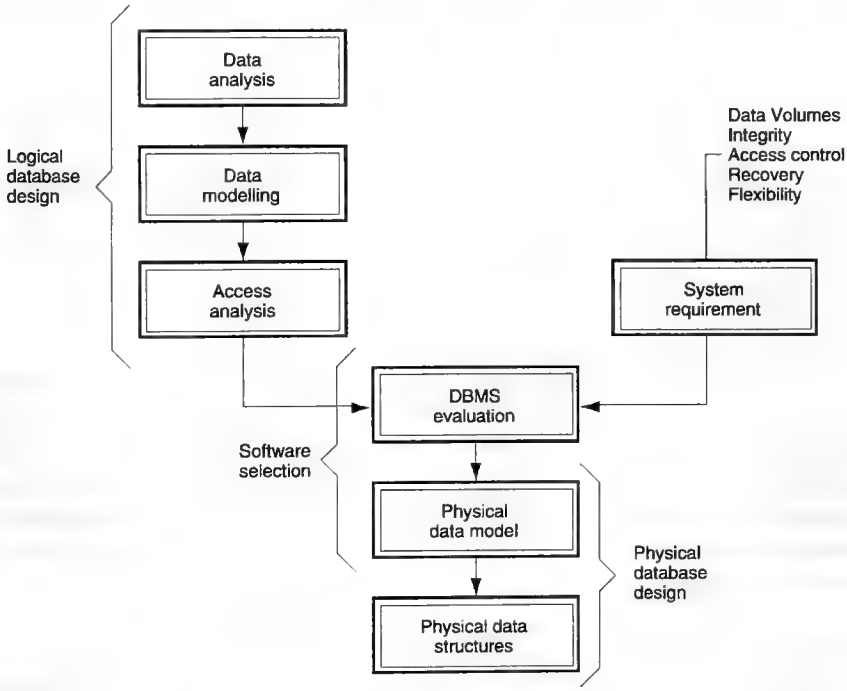

Figure 3.1: Database Design

It is important to note that when designing a new database, the earlier errors are identified the easier (and therefore cheaper) it is to correct them. Correcting an error in the logical design is simple. Correcting the error after the logical design has been transformed into the physical design is more difficult, but tolerable (this may require equipment/software modification). Correcting an error once the database has been implemented is much harder.

Likewise, correcting an error after the application has been constructed is more expensive still; and correcting the error after the users have begun to employ the database for their work (and entered data) can be exceedingly expensive (this may require changes in hardware, software, systems redesign, and human resource training). Thus, it is important to make every effort to have a complete and correct definition of the database at the logical design and data modelling stage shown schematically in Figure 3.2.

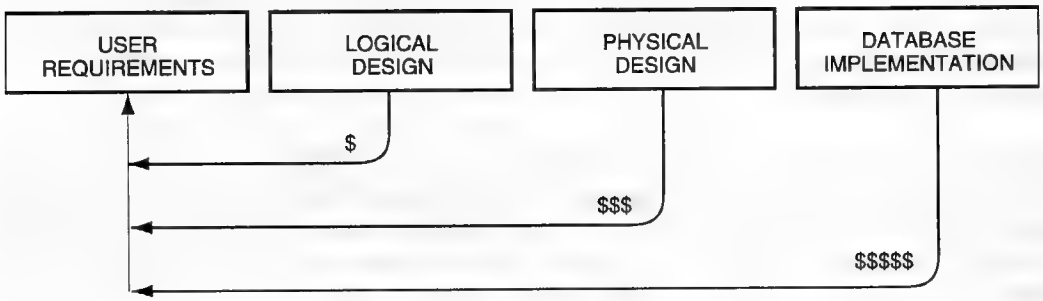

Figure 3.2: Relative Cost of Changes at Database Development Phases

Guideline 3.3
Employ a systematic database design methodology and focus particular effort on the earliest stages of the design process.

3.3 Logical Database Design

3.3.1 Overview

Production of the logical design involves a detailed analysis of the data requirements of the proposed system. This process involves analysis of the data requirements and the ways in which data need to be accessed, on the basis of feedback from prospective users. The resulting design should be completely independent of both hardware and software, and should not assume any particular method of physical data organisation. In practice, the hardware and software platforms available (perhaps constrained by budgetary limitations) may mean that performance expectations limit the logical data design.

The advantages of approaching the design of the database in this way are:

- it provides a stable base from which to set standards and coordinate the development of the system

- it provides a "data model" that is completely free of any implementation considerations, and which can be used as a point of reference when adding to or modifying the functionality, or changing components of the hardware or software

- it provides a specification which can be used in the evaluation of alternative database management software

- it provides a base line from which an optimum physical data organisation can be produced.

3.3.2 Data Modelling

It is important to understand and document the data included within the intended database, ie the data needed to meet user requirements. This procedure is separate from designing the **processes** required within the system, although the two are closely linked. Following this a **data model** may be defined to identify the inter-relationships between the selected datasets.

A data model is a representational tool consisting of language and diagramming standards representing the overall structure of and inter-relationships between a group of datasets. To build a data model, the development team must fully understand the processes involved in creation of the various datasets, how they overlap, and how they depend on one another. This understanding is best achieved by interviewing prospective users of the system, plus experts in the biodiversity domain.

During the interviews it should be kept in mind that users interact with database **applications**, not directly with the database which refers to the actual collection of datasets being managed. Thus users "see" the database data through data entry screens, menus, and reports. When users express requirements, they normally refer to design features of the application interface. Translating **application** requirements into **database** requirements and thus into the design of the **data model** is the responsibility of the development team.

The first step in the development of a data model is to study the functional specification that was produced during the User Requirements phase (see Section 2.4.2). Consideration of this document, together with discussions with both users and experts, permits determination of the basic "items of interest" and hence the initial **entities** of the data model.

The next step is to determine what relationships exist between the entities that have been identified. It is important at this stage to concentrate on the "natural" relationships which exist, rather than just those which it is thought may be computerised. The finalised version of the model will only emerge following the completion of the access analysis phase (see Section 3.3.2).

Data models are often represented in a formal manner. The most popular representation is the **entity-relationship** (E-R model), first described by Peter Chen in 1976. This model provides a very clear diagrammatic representation of the top-level objects to be modelled in a domain. In the original paper, Chen set out the foundation of the model; it has since been extended and modified by Chen and many others. In addition, the E-R model has been made part of

a number of Computer Aided Software Engineering (CASE) Tools (see Document 4, Section 2.3), which have also contributed enhancements. Today, there is no single standard E-R model, although most share the features outlined below.

Entities
Data items (concrete or abstract) that the database will contain; in a relational database, entities are represented as **tables**

Attributes
Characteristics that describe the entities, (eg "IUCN Category" is one attribute that describes the entity "Protected Area"); attributes are represented as columns or "fields" in database tables, such that all instances of a given entity have the same attributes

Relationships
Descriptions of how two entities relate to one another (eg "species" may be related to "genera" by a "belongs to" relationship). Figure 3.3 illustrates this.

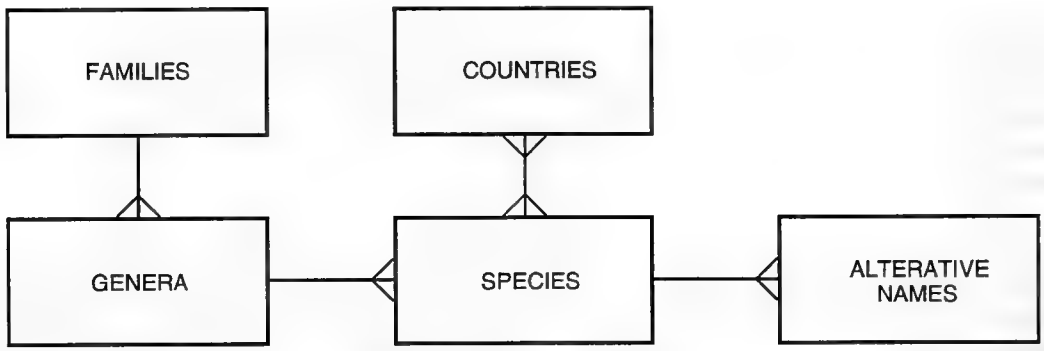

Figure 3.3: E-R Model

Alternative symbolisms are used to construct entity-relationship diagrams. The notation adopted in this document follows that of Ashworth and Goodland (1990). Connecting lines between entities are single or forked depending on their relationship, forked lines indicating the "many" side of a one-to-many or many-to-many relationship (see Document 1). For further description see Kroenke, 1992, or Document 4, Section 2.3.

The advantages of undertaking a data modelling exercise are:

- improved dialogue between users, and consequent development of data structures
- identification of redundant data
- improved capacity to identify data validation criteria
- a formal, possibly automated method for implementing the physical database.

Guideline 3.3
Prepare Entity-Relationship diagrams to explore data relationships and record the data model.

3.3.3 Access Analysis

Following the data modelling phase, the next step is to study how the data are to be used. For each data retrieval process the access requirements are recorded in terms of:

- the entities, attributes and relationships used
- the types of access required to support the various processes
- the access keys required
- the frequency of access
- the response time required.

A composite picture of the access requirement can then be produced by aggregating the estimated usage statistics. This information can be used to determine which relationships are required to support the system, and how to optimise data structures to best serve data retrieval.

3.3.4 Software and Hardware Evaluation and Selection

By themselves, the logical data model and access requirement do not provide all that is required to specify the database. As indicated in Figure 3.1, there must, for example, be a careful investigation of:

- the data volumes associated with each entity
- the security requirements in terms of integrity, recovery and access control
- the degree of flexibility required.

If the implementing agency is not constrained to use a particular brand of data management software, and there is a choice to be made, the data model has an important role to play in the selection process. It is important to ensure that the software chosen will allow an effective transformation of the logical design into the physical design and through to the database implementation.

The most commonly used form of database management system on the market today is the **relational** database management system (RDBMS). These offer good flexibility and performance at modest cost. It should be noted however, that the RDBMS does not deal easily with time series data and textual information.

3.3.5 Architecture Selection

What makes a particular DBMS package suitable has more to do with the needs of the application than with the strengths and weaknesses of the software. What is good for a mathematical or statistical expert is not necessarily the best tool for a botanist or forest ecologist. When selecting the software, the following basic questions about the application should be answered:

- How big is the application? How many individual entities will be included? How many cases (instances) of each entity are there?
- How many people need to access the database? Will they be sharing a single computer or using a network? Are they all in the same institution or physical location?
- Are any special data types needed, such as spatial data, large volumes of text, images, sounds, or video? Will document storing and searching be necessary?
- How much computer experience does the implementing agency have? How much time is there to learn new software?
- How much money is available to spend on hardware and software?
- What are the long-term plans for the application? Will the scope or the number of users grow?

There are a large number of evaluation criteria which can be used in the selection of a DBMS. The ones with the highest priority are the following:

Maximum sizes:
- Field size
- Fields per record
- Fields per database
- Record size
- Records per table
- Indexes per table

Field (attribute) types supported:
- Text
- Integer
- Floating point
- Date
- Logical
- Object (image/sound/video/ document)
- Spatial

Data Dictionary:
- Automatic
- User defined

Data validation functions:
- Lookup tables
- Valid data ranges

Search parameters:
- First occurrences
- All occurrences
- Index fields only
- Multiple fields
- Value or character range
- Soundex[1]

Performance:
- Search speed
- Reliability

Technical support:
- Available locally
- Available on-line
- Good manuals
- Other nearby users

[1] Soundex is a commonly used algorithm that enables searches to be conducted on the *sound* of a word and enables retrieval of words in text strings that are mismatched on spelling.

The criteria above should be evaluated against the requirements specified in the data model and access analysis. However, not all evaluation criteria are equal, and thus counting the number of check marks in each column is a poor way to compare products. For example, features such as reliability and speed may overshadow lesser capabilities.

If performance is a critical consideration, it may be necessary for the development team to test the DBMS themselves under realistic local conditions. Published software "benchmarks" are often optimistic and may not reflect the demands of the destined application. Many important characteristics of a DBMS are often subjective. These include ease of use, consistency of the user interface, and expressiveness of a programming language. Selecting a DBMS package purely from a list of features is unlikely to be satisfactory; nothing can substitute for examining a live system.

Finally, it is not unknown for application developers and end users to taint evaluation criteria with personal and professional biases, knowledge, and experience. Thus unless you will be the sole user of the DBMS, you must account for the skills and needs of **all** users when you apply the criteria.

Fortunately, reputable computer magazines often contain advertisements and wide-ranging reviews of DBMS packages, although these too can be biased (software reviewers sometimes have connections with vendors whose products are under review). If you rely on published reviews, temper the prejudices of any one reviewer by using several sources.

Computer bulletin boards are another source of outside expert advice. The vendors of very popular DBMS packages usually maintain bulletin boards which may be accessed via services such as Internet newsgroups and CompuServe Forums. Bulletin boards not only store objective assessments of software, but can also provide solutions to technical problems via a network of remotely connected users. Knowledge can often be gained simply by observing the debates and comments of other users.

3.3.6 Making the Final Choice

Be sure that you can answer the following questions in the affirmative before you commit to purchasing DBMS software:

- Is it powerful enough to fulfil the application requirements?
  (consider speed, file, and record limitations)
- Will it meet end-user requirements?
- Is the user interface intuitive or at least familiar?
- Does the application contain good facilities for application development?
- Will designing the file structure, forms and reports be easy?
  (remember that the amount of money spent on application development usually exceeds the DBMS software cost, so faster development time can result in significant savings)

- Will the application be easy to maintain?

- Does the software have a future? Is it an industry standard that is sure to enjoy continued support and enhancement? (it can be beneficial to forsake the latest software technology, best performance, and slickest user interface, in exchange for the stability and support of a large well established software company)

- Is it affordable?

Guideline 3.5
When selecting DBMS software consider the criteria that are of most importance to your DBMS application, prioritise them, and assess how well they are met.

As well as choosing the DBMS (probably a relational DBMS), the development team should also design the computer **architecture** on which it will operate. There are various options, including:

- stand-alone PCs
- networked PCs with the database software residing on a file server machine
- client-server architecture.

The third option, **client-server architecture** is an increasingly popular solution to the data processing needs of medium to large-sized organisations. This architecture is a hybrid of the stand-alone and the traditional network options and integrates the best characteristics of personal computers (friendly software and quick response) with the best traits of powerful centralised servers (high storage capacity, data sharing with strong security). Client-server architecture is a combination of hardware and software that enables users to divide computing tasks between the user's workstation (usually a microcomputer) and server computers (micros, minis or mainframes, depending on the volume of data and the speed required). The user interface runs on the workstation, the database software runs on the server, and a communication path links the two.

The key benefit of a client-server system is that it is "pooled" between different applications. For example, a graphics program could be used to create a chart based on information stored on a central DBMS.

3.4 Physical Database Design

The physical database design involves adapting the logical database design to the requirements of the actual hardware and software used for implementation; the usage statistics resulting from the access analysis phase play a vital role in this process.

In general, an entity in the E-R model will become a table in the physical model, and attributes become table fields. However, the way in which relationships between entities are dealt with depends on which DBMS is used, and may therefore be different in each case. With some types of DBMS, relationships are established explicitly by means of pointers stored with the data. With RDBMS systems, data is held in tables and the data model is implemented indirectly via indexes.

If the chosen package does not support some types of relationship, this has to be resolved by altering the logical database design. Two ways in which this may occur are:

- implementation of many-to-many relationships can be achieved by creating an artificial "intersection" entity to establish two one-to-many relationships

- two entities linked by two relationships may be simplified by adding an extra entity and relationship, so that only one relationship exists between the entities.

During the transformation of the logical design to the physical design, CASE tools (see Document 4, Section 2.3) can prove very useful. These allow E-R diagrams to be drawn up, and used to validate and maintain the logical database design. Some CASE tools are also able to output the E-R diagrams directly into a **Data Definition Language (DDL)** that prescribes the physical database design.

3.5 Database Implementation and Installation

3.5.1 Overview

These guidelines emphasise the **relational** model as the basis for database design. This model allows direct movement from physical design to implementation, without the need for alterations.

Depending on the software used, there are a number of different means to define data structures. With some products the structure is described in a text file (or from a command line) using some form of DDL (for example the dBase command set). These commands detail the tables, columns, indexes, and other structures such as quality constraints and security restrictions. Databases with a full SQL (Structured Query Language) command set can use SQL to fulfil the role of the DDL.

One common alternative to a text or command based DDL is to provide a graphical means for defining data structures. With Paradox (see Document 4, Section 3.2.1), for example, the developer is shown a graphical list structure and asked to fill in the table and column names in the appropriate places. In general, graphical definition facilities are easier to use than textual DDL.

Regardless of the means by which the data structures are defined, the development team must name each table, define the columns for that table, and describe the physical format of each column. These definitions form an essential part of the **data dictionary**, which is a repository of information about the structure, format, and use of data.

The business world is highly heterogenous and a database for one company is unlikely to use the same data dictionary as that of another. In contrast, it is likely that countries and organisations monitoring biodiversity information will be recording and tracking many of the same parameters. Thus in the interests of data exchange and cooperation with external partners, notice should be taken of existing standards and common practices (see Document 4, Sections 4 and 5).

Guideline 3.6
Develop data dictionaries for database tables using standard biodiversity terminology and thesauri where available.

3.5.2 Populating the Database

Once the database design has been implemented, then it can be *populated* with data. The means by which this is done depends on the application requirements and features of the DBMS product. In the simplest case, all data are available on magnetic media in a format suitable for direct importation into the DBMS. In the worst case, all data must be entered manually via the keyboard using custom application programs created by the development team. Most data conversions lie between these two extremes.

Where data are entered via the keyboard, validation checks should begin with rigorous examination of the raw, normally hard-copy, data sources. This can be a labour-intensive and tedious task, but is very important for maintenance of data quality (see Section 5). Where data are not entered directly, but are imported in bulk from magnetic media, validation checks should be performed following the import process.

As an example of the possible types of validation, Richardson's (1994) account of the checks applied to species distribution records prior to entering the ERIN system in Australia is presented below:

- records are checked to see that all required data fields are present
- scientific names are checked for validity
- grid references are checked for being over land, not sea
- the presence of a species in a certain location is tested against a prediction based on bio-climatic factors, and outliers selected out for study

For large applications, it is a good idea to write special-purpose validation routines, or take advantage of automated procedures offered by the DBMS package. Such routines perform "reasonableness" checks on field values, such as ranges for numeric fields, or string-length for character fields. It may also be possible to enforce consistency checks such as capitalisation and hyphenation. Finally, many DBMS packages permit the user to select field values from a set of predefined choices. This effectively eliminates the possibility of typographic errors, and can speed up data entry considerably.

3.5.3 Relational Data Retrieval

Structured Query Language, or SQL, is the most widespread relational data query language in use today. It has been endorsed by the American National Standards Institute (ANSI) as the language of choice for querying databases, and it is the data access language used by many commercial DBMS products. Since a version of SQL will run on virtually any computer and operating system, computer systems may exchange data by passing SQL requests and responses to each other. Increasingly, biodiversity monitoring organisations may decide to interchange data across networks using SQL.

Guideline 3.7
Where possible choose a DBMS which supports data access through SQL.

3.6 Special Considerations for Biodiversity Databases

3.6.1 Synonyms and Equivalent Terms

In a typical RDBMS, information can only be retrieved by means of precise requests. Thus, if the user wants to find information on protected areas by providing the search string "protected area", the search will fail to retrieve records marked "park", or "reserve" or "sanctuary", despite the semantic similarity. The problem of synonyms and equivalent terms is particularly prevalent in the environmental domain due to its heterogeneous make-up.

This difficulty can be overcome by developing custom search routines and offering them to the user as menu or push-button options. An on-line thesaurus can also assist the user by providing a series of alternative search terms. This can be done in a passive mode by suggesting the terms to the user on request, or in active mode where the thesaurus is automatically consulted during the search process to identify synonyms and semantic matches.

There are currently several international projects to assemble environmental thesauri (see Document 4, Section 4.4). These are being developed in multi-lingual versions (primarily European languages at this stage). The most mature of these thesauri is the *INFOTERRA Thesaurus of Environmental Terms* (UNEP, 1990), which currently contains around 1,600 terms. This number is not sufficient to cover many local terms, and must therefore be augmented in such situations.

3.6.2 Integration of Text with Conventional DBMS

Biodiversity information is frequently exchanged in textual form (see Section 4.1.2), which is not easily processed by many DBMS packages. Although small amounts of text (no more than a few lines) can be stored in character fields, and moderate to large amounts in memo (free text) fields, the text remains unformatted without font changes, *italics* or **bold**.

This difficulty is now receding as software manufacturers agree on software connectivity protocols. For instance, very modern DBMS packages permit whole text documents in various formats to be embedded as "objects" in special kinds of fields. Less sophisticated solutions include the establishment of fixed links to external word processing packages via **pointers** stored in database fields, and the use of internal word processors within the DBMS.

An example of DBMS linked to a word processing package is the protected areas database of WCMC. This is a Foxpro application linked to WordPerfect text files. Every record for a protected area contains a field holding an alphanumeric code pointing to a document address. A menu option in the application applies a macro to that field, picking up the code as a filename, and retrieving the corresponding WordPerfect file. The user of the application can then browse and edit the protected area's "site sheet" in WordPerfect.

This is not an ideal approach as it requires manual intervention and document tracking, and does not ensure automatic updating. For instance, the deletion of a protected area from the database will not force a corresponding deletion of the associated text file, nor will a change in name of the protected area cause a change in the corresponding text. In addition, the content of the text files cannot be queried or searched by the DBMS.

The task of integrating text and tabular data can also be approached from the opposite direction. In this case, data held in a DBMS are referred to **from** a special type of document known as a **hypertext** document. These have "hyperlinks" embedded in their text which permit the reader to "jump" off to other sections of document that concern a particular subject of interest. The reader may return to their original position simply by reversing the jump, a process which is described as moving forwards and backwards through hypertext.

Hyperlinks can also be used to make searches on connected databases. For instance, a hypertext document describing a species database could be provided with a **form** to be filled out by users. On the form would be entered the names of the species to search for, plus the name of the desired database file. Upon request, the chosen database can then be searched and the resulting data sent back to the hypertext document as a new text page. To be most effective, the hypertext pages should be laid out in a highly consistent manner, using a tightly controlled vocabulary.

Guideline 3.7
Where large text files are to be linked to a conventional DBMS, text should be consistently structured and consideration given to hypertext approaches.

3.6.3 Integration of Spatial Data with Conventional DBMS

Biodiversity information systems often store data that is ultimately related to a geographic location. For instance, protected areas have geographic boundaries, species have distributions, human and natural forces, (eg rainfall) have geographic zones of influence. Conventional DBMS can store geographic attributes within biodiversity databases, for instance by keeping the coordinates of a polygon as a large string of characters in a character field. This is rather awkward in a relational database since the number of coordinates cannot be specified in advance. Additionally, the conventional DBMS has no facility to respond to spatial enquiries, such as "is this site within this region?", or "how many hectares of this vegetation type occur at an altitude of less than 200 meters?".

Spatial analysis is better achieved through the use of specialised Geographic Information Systems (GIS) and desk-top mapping packages. These treat attribute data, (eg vegetation type, protected area designation) separately from spatial data components. In summary, effective analysis of biodiversity information may often require non-spatial attributes in a conventional DBMS to be linked (joined) to corresponding attributes in a GIS.

This can be achieved by setting up a one-to-one relationship between the DBMS table and GIS coverage, using a common key or unique polygon identifier (examples of such relations are given in Document 1). Some GIS and mapping packages provide simple linkage facilities, (eg ARC/INFO, MapInfo), but as with text, some custom programming may be needed (see Document 4, Section 3.2.5). For example, WCMC manages its Biodiversity Map Library using the ARC/INFO system. Here, the polygon coordinates are held in binary files and linked to an internal database of attributes. The linkage to an external DBMS is made through the ARC/INFO Polygon Attribute Table (PAT) and a standardised set of polygon identifiers. Adjustments are made to the DBMS manually when changes are made to the PAT. This can be time consuming when complex geographic processing has taken place.

Guideline 3.8

Integration of spatial data with a conventional DBMS can be achieved by establishing a link between the DBMS table and map attribute file, by means of a common field such as a polygon number.

3.6.4 Handling Hierarchical Taxonomic and Classification Data

Alongside textual and spatial data, **relational** database management systems (RDBMS) do not easily deal with some other varieties, notably time series data and hierarchically organised information, such as taxonomy.

In a recent study from Australia, Richardson (1994) highlighted the problems encountered when establishing a taxonomic database structure, and the need for these to be tackled during the system design phase. Firstly, systems had to be designed to integrate differing standards between disciplines, (eg botany and zoology), and between institutions. This is especially common at the generic level where different practices can result in the "splitting" or "lumping" of genera. Secondly, taxonomic standards change with time, as knowledge of the phylogenetic relationships between species improves. Thus data supplied by different sources may use differing names for the same species, and the database structure must be able to integrate these synonyms. This situation may also arise when it is discovered that taxa previously thought of as one species consist of two or more, and as a result a part of the data for a species is included under the wrong name. Richardson suggested that taxonomic database structures should take into account the following:

1. Formal Categories. The family, genus, species, sub-species, other infra-specific categories, and corresponding authorities of the taxa (family name is included as the same name may be used for genera of plants and animals).

2. Applied Categories. Users may need to associate other names with the formal categories such as synonyms and common names. Applied categories should be fully referenced in terms of authority, date, and published source.

Including hierarchical entities in a relational database is feasible, but requires some additional tables and linkages which are somewhat inelegant. The hierarchy is achieved by having a field in the "child" entity that defines which "parent" entity it belongs to. That means that there must be at least one entity for each level in the hierarchy, even if there are no attributes associated with the levels.

For example, in the *BG-BASE* database used to store information on threatened plants and plant collections, the records in the *Names* table contain information on plant species. One field of each record contains the **Genera**, providing a link to the *Genera* table. Thus, the plant name *Acer palmatum* would have "Acer" in its **Genera** field. To truly complete the taxonomic hierarchy requires an additional entity for the family, sub-species etc, and linked tables of synonyms. Many of these tables will only have two fields (or columns) with unfortunate extra overhead in processing and storage, but necessary to respond to questions such as "to what family does *Acer palmatum* belong?".

Hierarchies are often required to manage other forms of biodiversity data, such as geographic

hierarchies where region *x* is located in country *y*, in continent *z*. A further example occurs with multi-level classification systems, used commonly for description of land-use, vegetation, and other ecological classifications. Seemingly simple attributes such as "vegetation type" must often be broken down into multiple entities and attributes as in the case of taxonomy.

Care during the User Requirements and Design phases, including judicious choices of classification and nomenclature standards, can reduce the number of hierarchical levels needed, and thus the complexity and performance loss of the system.

Guideline 3.9
Special attention must be given to the design phase to establish methods of handling taxonomic and other hierarchical data.

4 DATA ANALYSIS AND MODELLING

4.1 Data and the CBD

4.1.1 Overview

The broad scope of content and format of biodiversity data has implications for the definition of analysis strategy. Key considerations include quality control, integration of spatial data with other sources, processing very large volumes of data, and development of models and analysis techniques which produce information in formats suitable for decision making. Before examining analysis strategies in detail, it is useful to consider which categories of biodiversity information are most required by the CBD. The latter identifies three broad categories of information for reporting purposes:

- ecosystems and habitats
- species and communities
- described genomes and genes of social, scientific or economic importance.

To this basic list one must add:

- the scientific and technical information required to measure, assess and take decisions on appropriate action
- bio-technology, its value and risks
- local knowledge of traditional uses and values of biological resources
- interrelationships between biodiversity, human actions, laws and conditions, economics and development.

To be effective, this list of information requirements (see Document 1) should reflect the way in which national and international agencies are organised to manage biodiversity information. The eight-point classification below consequently has some pragmatic appeal:

Conservation
Encompassing information on species, habitats, protected areas, biodiversity indicators, wildlife, etc.

Genetic Resources
Encompassing agriculture, agricultural research, gene banks, use of genetic resources for benefit of mankind, traditional uses, genetic threats, etc.

Technology
Encompassing information on the technology of biodiversity monitoring and assessment, such as data collection technology, computer systems and telecommunications, remote sensing, geographic information systems, database techniques and standards.

Biotechnology
Encompassing a forum for interchange of information on research and application of biotechnology.

Environmental Statistics/Economics
Encompassing resource utilisation, value of biodiversity, land use, industrial outputs, equitable sharing of benefits, natural resource utilisation, trade, economics etc.

Policy
Encompassing policy development, modelling, decision support systems and technology, empowerment and public consultation techniques, etc.

Human Factors
Encompassing population, human health, social conditions, and their relationships to biodiversity.

Environmental Law
Encompassing environmental legislation, conventions, protocols, regulation, standards etc.

4.1.2 Specific Categories

With limited resources available, the setting of priorities for the types of information to be collected and analysed is critical. Furthermore, it is essential to review what data already exist, and who are the custodians. This effort serves not only to minimise duplication of effort, but also to foster cooperation between related organisations as described in *Guidelines for a National Institutional Survey* (Document 2).

Specific categories of information which are most relevant to the CBD include:

- baseline distribution data for species and habitats, for assessment of status and planning
- socio-economic value of local and national biodiversity, and of protected areas
- functions and benefits of biodiversity, particularly service functions of ecosystems and protected areas
- policy and conservation programmes, legislative framework, and other institution-related matters
- technology useful in monitoring, assessing, and improving the sustainable utilisation of biological resources
- genetic resources, including medicinal plants, landraces and wild ancestors of domestic breeds and cultivars
- species that can serve as *indicators* of ecosystem health, and *flagship* species, the conservation of which protects other species and habitats

- alien or exotic species, the spread of which could threaten indigenous biodiversity
- threats to biodiversity, and biodiversity known to be threatened
- changes in species and habitat distribution over time.

4.1.3 Data Form and Media

Within each of these categories, data exist in a variety of forms and media as described below.

Numeric Data

Numeric data are derived directly from many types of survey ranging from counts of species in particular locations, to measurements of rainfall, tree growth or the length of a bird's primary feathers (which might be used in identification and taxonomic work). Numeric data can also be generated automatically from climatic recording machines, or derived from remotely-sensed images.

Numeric data lends itself to computer-aided analysis, and the derivation of further datasets based on such analyses. For example, the absolute altitudinal range of a protected area can be derived from subtracting the lower altitude from the upper. It is also extensively used in modelling. For example, information on the temperature, rainfall and altitude of a particular site (all numeric data) can be used to predict the Holdridge life zone within which it lies. It is possible to structure numeric data very strictly in database tables, and exercise stringent validation procedures during data entry.

Categoric Data

Biological information commonly includes classified or coded non-numeric data, such as descriptions of soil type, land cover, forest type, life form, protected area designation, and so on. The data are usually structured through a thesaurus or data dictionary, and can be restricted to allowed values. Although statistical analysis may not be appropriate, categoric data are frequently used for database searches. For instance, if a life form category was given to every plant distribution record in a database, it would be simple to list all the "tree" records assuming "tree" was a life form category.

Textual Data

Text is an extremely common form of biodiversity information, including descriptions of protected areas, descriptions of species, descriptions of threats, ecosystem status reports, "State of the Environment" reports, legislation, regulation, strategies and plans. By comparison with numeric or categoric data, it is much less structured, often subjective, and difficult to search and retrieve unless stored in a regular format. However, when combined with other forms of data, text can provide valuable supporting information regarding data quality and sources. For example, text can be used to support the identification of a rare species, or clarify a difficult ecosystem classification.

Spatial Data

A spatial component is present in most biodiversity data, since they are drawn from the environment around us. Spatial data include point location records for species, species ranges, and protected area boundaries. Also included are descriptions of basic biogeographic

phenomena such as climate, topography, vegetation, and land use. The latter are often presented on paper maps, or held in remotely-sensed digital format, or in computer-based geographic information systems.

Data on Other Media
Biodiversity information may also include non-digital images (photographs, drawings) of landscapes, specimens, and technology (instruments, methodological flow diagrams and so on). Future consideration should also be given to moving images (such as video sequences recording wildlife behaviour), and sound recordings.

4.2 Approaches to Data Analysis

4.2.1 Overview
The subject of analysis, modelling, and interpretation of biodiversity data is central to the synthesis of **information** suitable for developing national strategies, action plans, and for monitoring and reporting progress under the CBD (see Figure 1.2 and Document 1, Section 1).

Elementary data analysis procedures, such as summation and averaging, are standard features of most database and spreadsheet packages. Given suitable data, they enable simple calculations to be performed such as the total number of species recorded in a protected area, or the average abundance level of a particular species nationwide. Results can often be summarised in the form of lists, tables and charts. Designers of biodiversity information systems can save valuable development time by making use of these built-in features.

However, in some situations it is necessary to apply more complex, possibly spatial, analyses to biodiversity data in order to obtain the desired information. Some examples of situations demanding more complex procedures are:

- assessment of population trends (time-series analysis)
- modelling of species-habitat relationship (canonical analysis, predictive models)
- assessment of protected area complementarity (clustering techniques)
- determination of landuse or vegetation types from remotely-sensed imagery (image processing)
- monitoring resource depletion in buffer zones (buffer zone analysis).

There are three basic approaches to complex data analysis: packages, custom program design, and modelling.

4.2.2 Packages
The simplest approach to complex data analysis involves the use of the capabilities of commercial (or academic) analysis packages. These can be divided into the following groups:

- statistics packages
- GIS and desk-top mapping packages
- image analysis packages.

The use of packaged software can greatly reduce implementation time and costs compared to custom programming, and more importantly improve compatibility with other institutions for information exchange. Indeed, it is worth establishing an informal policy to limit the number of different packages in use by cooperating institutions to a short list, to take full advantage of the built-in compatibility and shared support which this will bring. This type of agreement could be done through the User Advisory Group identified in Section 2.1.

A disadvantage of packaged software is that the user is constrained to the functions included in the software, that is, it is usually not possible to add an additional function or operation at a later date. However, this need not be a problem if a good range of options are provided, and thus it is important to select the software carefully. Useful criteria to consider include:

- compatibility with existing configuration (hardware and operating system)
- data transfer compatibility with existing application software and DBMS
- richness of functions of the package
- well known and respected supplier
- customer support available locally
- commonly used by other institutions for similar biodiversity applications.

Information on some commonly used packages is provided in Document 4, Section 3.2.

Where an external package is adopted, it is important to continue to view data analysis as an integral component of the overall information system. Thus it is necessary to consider ways in which the facilities of the package can be integrated with other information system components. For instance, data are commonly entered directly into database tables or spreadsheets. In the case of species distribution records, it would be convenient if they could be mapped *on-line*, without having to exit the data entry process. This would allow errant records (such as terrestrial records apparently occurring over open water) to be rejected at the earliest opportunity.

This facility is now available following the popularisation of "multitasking" computing environments in recent years. Graphical user interfaces (GUI) such as Microsoft Windows, OS/2, Macintosh, X-Windows, and Motif, all permit several packages to be run concurrently. A solution to the case described would therefore involve running the mapping package and data entry system at the same time, and switching between applications to check the validity of records.

Guideline 4.1
Packaged analysis software should be acquired as needed to assist in information processing and interpretation. The choice of package should particularly consider compatibility with existing hardware and software, and with other similar biodiversity institutions.

4.2.3 Custom Program Design

Analysts who possess a good knowledge of statistical theory and programming concepts can write "custom" analysis routines using a computer programming language. Options include the macro language of the RBDMS or spreadsheet package managing the data, or a high-level

language such as BASIC, FORTRAN, C, or PASCAL. In some cases the task may be simplified by drawing on third party "libraries" of commonly used statistical routines, or alternatively, implementing published program listings or "numeric recipes" directly. An example of this approach might be the calculation of biodiversity indices for a series of protected areas, which might involve counting species in each area, and weighting according to distribution and so on. Relatively complex routines can be written in this way and integrated into analysis processing in the information system in a way similar to packages.

4.2.4 Modelling

The above discussions have concentrated on providing solutions to specific analytic problems. However, the same techniques can be applied to **modelling** ecological phenomena, albeit at a higher level of technical and conceptual complexity.

As with more traditional data analysis techniques, custom programming and package options are available. However, in the latter case, packages are not normally commercial in nature; they are generally designed by academics, government researchers, conservationists, and others directly involved in biodiversity research. Examples of problems requiring modelling include unravelling the relationship between species and habitats, assessing the impact of climate change on natural and managed ecosystems, the effect of forest management practices on tree regeneration, and the effect of population growth on ecosystem health.

It is obvious that many of the areas requiring modelling work are at the cutting edge of current knowledge, and are thus evolving rapidly. For this reason most available models apply only to specific local situations and may only be valid elsewhere under limited circumstances. More general models are better developed in more mature disciplines such as agriculture, forestry, and climate than in biodiversity assessment. Models for national biodiversity sustainability and economic valuation are at very early stages of research.

While it is preferable to make use of existing modelling software where possible, it should be recognised that such models will nearly always require modification to suit local or national conditions. Useful criteria when selecting modelling software are as follows:

- adaptability to local conditions
- scientific peer acceptance
- compatibility with existing software and applications (eg with DBMS and GIS)
- good documentation and future support (eg newsletter).

Information on various modelling software is given in Document 4, Section 3.2.7. A good example is the BIMS software, developed by Asian Bureau for Conservation, Hong Kong. BIMS uses a simple, but effective, model to relate species occurrence to recorded habitat. For each species of concern, the following parameters are used as determined from field records: recorded altitude range; suitable vegetation types (following an accepted convention); known bio-geographic zones (following an accepted convention).

In theory, these parameters can be deduced from a small number of records. However, larger numbers of records help to improve the quality of the model, especially where these are derived from structured survey operations. Having deduced the parameters, it is possible to

search the region of study (protected area, country, region, or world) for matching habitat. This permits the potential distribution of the species to be determined, and consequently its status in terms of distribution extent and protection level.

Guideline 4.2

When implementing computer-based biodiversity models, attention should be paid to their suitability and adaptability to national conditions.

5 QUALITY MANAGEMENT

5.1 Introduction

Quality management refers to the overall process which governs the quality of a product from beginning to end. In the case of information the process begins with data collection and ends with the user application. Quality control checks and quality assurance methods should be applied through all stages of this process.

There can be no absolute measures of the quality of a dataset. What may be "high quality" data for regional planning may be poor or useless for local decision making because of factors such as scale, detail, and error. Particularly with biodiversity data, datasets can rarely be made error-free or "100% accurate", as the data is often based on subjective observation (such as deciding the boundary of a habitat), incomplete sampling, (eg soil sample collection, wildlife observation), or indirect measurement, (eg remote sensing). Even if it were theoretically possible to collect complete and accurate data, time and cost considerations often make this impossible from a practical standpoint. Therefore, with rare exceptions, it must be assumed that *all* biodiversity datasets will contain error and uncertainty. "Quality" must be considered a measure of "fitness for use" and is therefore relative to the proposed or intended use. This is a very important consideration when data are being integrated and used for applications beyond the original purpose of data collection.

Quality management in these circumstances requires attention to quality assurance, integrity protection, *and* to the complete documentation of the dataset in terms of its quality, uncertainty, limitations, origin, and intended purposes. Such descriptions of datasets are often referred to as "metadata" or "co-data".

Given that quality must be judged and assured in order to satisfy user requirements, an "end-user" approach to quality is recommended.

Guideline 5.1
Data quality standards should be developed and documented for each dataset in relation to the intended uses of the data.

5.2 Institutional Quality Standards

The establishment of institutional-wide quality standards is best exemplified in the series of Quality Management Standards of the International Organisation for Standardisation, identified as "ISO-9000". These standards are unusual in that they are generic and process-oriented; that is they do not specify any specific levels of quality for products, but instead insist on a continuous **documented** process which includes an **active** feedback mechanism to ensure that quality deficiencies, and quality management deficiencies are diagnosed and treated.

No other international standard currently takes this approach, although others are now being developed in the area of environmental management, (eg ISO Technical Committee 207). Much simplified, ISO-9000 requires a biodiversity-related institution to provide:

- a **Quality Policy** that everyone in the institution should understand

- a **method of measuring the quality** of information outputs that is applied consistently
- a **method of determining external user satisfaction** with information outputs that is applied consistently
- a **feedback mechanism** (see Figure 5.1) which ensures that internal and external measurements are actually used to ensure or improve the quality of the information service, as specified in the Quality Policy.

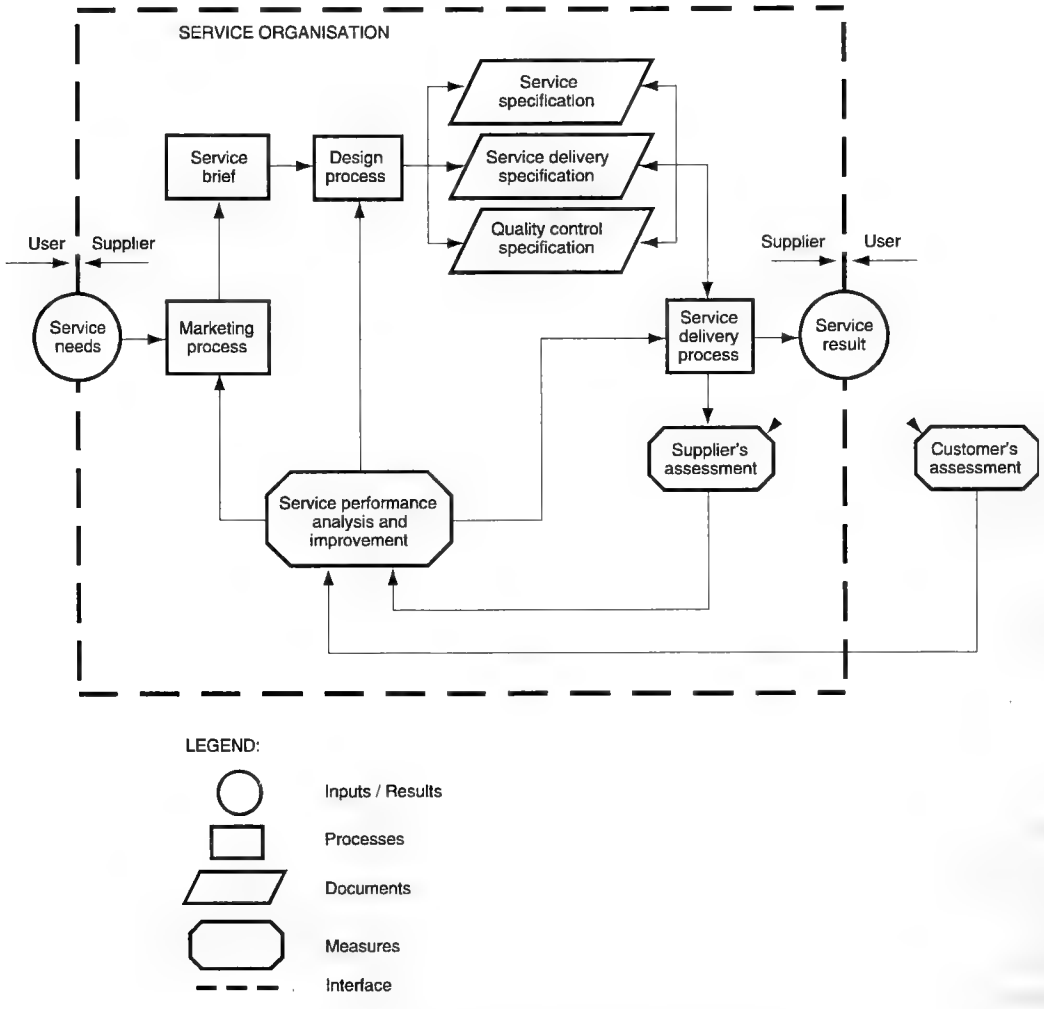

Figure 5.1: The Quality Management Loop

The overall emphasis is on the **end user**, and on quality considerations across all aspects of the operation (hence the term "Total Quality Management" is frequently applied).

The organisation is totally free to establish its own Quality Policy, specific quality measures and targets, measurement methods and feedback mechanisms which are appropriate to the national needs and the nature of the issues being addressed.

Certification is by an **independent** third party (according to the ISO-10000 Standard) which audits and certifies that the various specified processes are in place, and are followed.

The ISO-9000 series of standards leads to the "certification" of **entire organisations** which have implemented processes which meet the conditions of the standards with regard to *all* their functions and products. Some institutions managing biodiversity information may wish to seek this overall certification. This can be a major undertaking, and take a long time to achieve. It is therefore recommended that institutions implement a quality management process which follows the spirit of ISO-9000.

Guideline 5.2
Institutions managing biodiversity information should implement a data quality management process which is compatible with ISO-9000, especially in consideration of the ***Quality Management Loop*** *of ISO-9004.*

5.3 Dataset Quality Audits

To establish whether a dataset meets quality objectives, a quality audit should be performed on a regular basis. Where possible, this audit should be conducted by a party who is independent of the collector and custodian of the data. In conducting a quality audit of a biodiversity dataset, the fundamental principle must be "truth-in-labelling"; that is the dataset must be **as described** and of a quality which is suitable for the identified (and implied) **uses**. In the past, agencies rarely devoted much attention to comprehensive dataset documentation. This was because datasets were usually built for one specific project by people who well understood the nature of the data and any deficiencies and caveats. At the end of the project, each dataset was usually archived, filed, or neglected. Dataset documentation has always been regarded as desirable, but has seldom been accorded a high priority because no one believed it would be of much real value.

Because datasets can and must be used for multiple purposes as part of an overall national biodiversity information system, comprehensive documentation of datasets is increasingly being recognised as an essential obligation of data custodianship, and in addition, a strategic corporate asset. However, dataset documentation must be planned thoroughly with suitable allocation of resources. Major areas which should be included in the documentation are reviewed below.

Dataset Description
This part of the dataset should contain the information needed to correctly interpret and use the data. Elements of this include:

- custodian (institution name, contact details, responsible person)
- technical format and data structure information

- data collection methods
  - measurement techniques
  - instrumentation
- availability
  - evaluate transfer media
  - charges
  - restrictions
- data sources (if secondary)
- data processing and reduction techniques applied
- data dictionary
  - exact meaning of all attributes
  - coding schemes
  - references to "standard" classification schemes
- reason for data collection and intended uses
- quality information
  - quality control processes applied
  - standards used
  - quantitative quality estimates
  - qualitative quality statements and comments on known limitations (may vary within parts of the dataset)
- full geo-referencing information (for spatial data)
  - projection
  - parameters
  - ellipsoid and datum
  - origin and offsets.

Spatial Quality (GIS datasets)
Three major questions are:

- are the data a correct reproduction of the original source? (if digitised from paper or copied/transformed from an earlier digital source)
- to what extent is this an accurate representation of the spatial phenomena? (ie how does it match reality on the ground, with associated questions of resolution or effective "scale")
- are the data internally consistent?

This last question seeks to ensure that data elements are consistent with both themselves and the stated topological and structural constraints. Basic tests for this include:

- all polygons closed
- networks correctly joined
- left and right pointers correct
- one-to-one relationship of spatial objects to attributes
- edge matching of "tiles" correct (both attribute and spatial)

- natural phenomena represented consistently, for instance:
  - streams flow down hill
  - rivers are in valleys
  - identified peaks are at the top of hills
  - cultural features consistent with land cover
- data missing in some layers but not others.

Each of the above quality areas would need defined minimum standards (at least of the "must be present" variety). These should be documented and made available to users of the service (similar to the *Quality Manual* of ISO-9000).

Guideline 5.3
Periodic independent quality audits should be performed of important biodiversity datasets with particular attention to the completeness and accuracy of "co-data".

5.4 Operational and Data Security

Approaches to ensuring that data entered into the information system are of the requisite quality were identified in Sections 5.2 and 5.3. However, a range of operational procedures are also required to guarantee the **continued** integrity of the data. In particular, protection from accidental or intentional corruption should be afforded. The three principal sources of corruption of electronic databases are:

- mechanical failure of disk drives and logical faults caused by power failures and fluctuations occurring during database transactions
- human errors in copying files, updating records, reorganising databases, and other operational procedures
- destructive effects of computer "viruses".

Threats to data security tend to be greater in countries where the physical environment may be less than ideal (eg the tropics), where technical expertise is in short supply, or where informal computer networks are the primary means of data sharing. The principal enemies of removable media (eg floppy disks) and computers are dampness, dust, extremes of temperature, and uneven power supply. Protective measures include:

- well documented rules of operation in the form of User Manuals, Standard Operational Procedures, Measurement Protocols, and Data Security Policies
- regular back-ups of all magnetic media
- regular virus checking with up-to-date software, and rigorous avoidance of un-licensed or "borrowed" software, computer games, and other personal-use software.

Guideline 5.4

Document and make widely known operational procedures for the information system, including procedures for regular data back-up, virus protection, exchange control, and other means of preventing accidental or intentional corruption of data.

A full discussion of operational procedures, which are a basic requirement of professional information systems, is beyond the scope of these guidelines. However, key references relating to data quality are provided in Document 4, Section 4.2.

Insight into the application of such procedures in the field of biodiversity information management can be obtained by reviewing the procedures of experienced organisations like the Environmental Change Network (UK). This organisation has a long-term monitoring programme at a large number of sites in the UK, and maintains a data structure using the Oracle RDBMS which explicitly holds dataset descriptions, including quality criteria, and quality codes. Detailed "Measurement Protocols " are provided to responsible data gatherers at each site, helping to ensure that data are collected in a consistent way, and that factors which may influence the quality of a particular measurement are recorded. Overall quality policies and objectives are currently being defined in the spirit of the ISO-9000 standard.

As a further example, The Nature Conservancy (US) has developed both standard operational procedures and quality guidelines for its Biological and Conservation Data System (BCD) which has been implemented in many countries. More information on these organisations and their information management practices can be found in Document 4, Section 8.1.

6 HUMAN RESOURCE ISSUES

6.1 Current Information Technology Environment

With computer systems being adopted at a rapid pace throughout the world the demand for technical support and managerial resources greatly exceeds supply. Even mature computer specialists require continual upgrade and renewal of their skills to deal with new developments in technology. Despite being marketed as time-saving, personal solutions to a wide range of technical and secretarial tasks, a network of personal computers offering varied and sophisticated services requires a high level of technical support. A suitable ratio of technical support staff to users has been estimated to be in the range of 1:25 to 1:100. However, since it is frequently not possible to share one support person between institutions having less than 25 users, the effective ratio is lower. Alternatives, such as remote technical support services, may alleviate the situation somewhat.

Human resource issues which arise in this context include:

- scarcity of expertise
- training and development of both users and support staff
- infancy of professional and vocational standards
- lack of standard job descriptions and skill sets.

6.2 Scarcity of Expertise

New technology requires new support skills. In the past ten years microcomputer-based networks have grown in scope, capacity, and complexity to the extent that they can now support most business and scientific applications. These networks have become pervasive in supporting corporate information systems. Database management systems have also become increasingly larger in scope and complexity, (eg distributed databases with update privileges). Technical support staff are therefore in great demand, as can be seen in career advertisements. This shortage of supply is likely to prevail for the foreseeable future.

The technical support skills required can be categorised as following:

- network operating systems, (eg Novell Netware, Banyan Vines)
- proprietary general purpose packages, (eg e-mail, word processing, graphics)
- proprietary applications development tools, (eg Oracle, FoxPro, Visual Basic)
- communications software, (eg router, bridge software)
- specific application packages, (eg GIS, biodiversity modelling).

Some of these skills are provided by equipment suppliers in the form of technical representatives, whose services form part of an acquisition contract; some may be obtained under contract from specialist consulting firms, (eg Oracle specialists); others may be provided by in-house staff, (eg e-mail specialists). Some vendors provide certification courses for their own products for a fee. However, since people with these skills are in short supply, the competition for them is strong and salaries are high. As a result, the provision of competent technical specialist support at small and/or remote sites is problematic. These specialists are in great demand in larger communities and enterprises which offer advantages of peer interaction, availability of training, and career advancement.

The appropriate approach to human resource recruitment will depend on such factors as: stability and duration of tasks to be undertaken; local availability of skill sets; obligations of suppliers to provide support services; institutional staffing budgets; and staff retraining opportunities.

Guideline 6.1
Consider various approaches to obtaining specialised technical support, including external contracting and sharing expertise with other institutions, as well as conventional in-house training.

6.3 Training and Development

Each new piece of computing equipment imposes an additional training requirement, sometimes for specialists only and sometimes for numerous users within an institution. This implies both direct (fees) and indirect (loss of work time) costs. It is common practice to send both small and large numbers of technical support staff to suppliers' premises for training.

Various approaches to training large numbers of users are available. For instance, contracting a trainer to conduct sessions *in-situ* at the institution; training a core group of staff at the suppliers' premises and tasking them with training others upon their return; or using in-house technical support staff or training unit to formally train users.

Informal training can also be obtained through interaction with telephone "help desks" and "hot lines", or via computer bulletin boards available over the Internet and other large electronic networks.

Guideline 6.2
Develop training strategies and plans for different categories of staff, eg users, technical support, and operations managers, which take into account the variety of training modes available.

A range of training and education options for biodiversity information management are provided in Document 4, Section 6.

6.4 Professional and Vocational Standards

Many graduates of universities and technical colleges in scientific fields, acquire a high level of competence in information systems and feel comfortable with computers. However, the extent and complexity of the tools and services used, may mean that they have never held responsibility for diagnosing and rectifying problems themselves. Consequently, they may find themselves unable to operate and maintain the less sophisticated computing equipment often found in more applied environments. This has major implications for both human resource selection and training program design.

It may be a long time before the major training institutions (universities and so on) begin to provide graduates suitable for applied biodiversity information management tasks. Indeed, from whatever background users emerge it will be a challenge to tune their skills towards those required.

No vocational standards currently exist for the "biodiversity information management support technician". Requirements will vary with the nature of the institution and the level of technology in use. Document 4, Section 6, provides some guidance on necessary skill sets and sources of information on vocational standards which may be of use in selecting staff and developing training programmes.

6.5 Job Descriptions

In government organisations, the preparation of job descriptions for positions requiring unfamiliar skill sets is often slow and difficult. Although many organisations now have technical support personnel working in information management, there is a dearth of appropriate job descriptions and associated statements of qualifications. Those working in the field are often seconded informally from other jobs and fall uneasily between customary "scientific" and "computer" positions. Consequently, institutions should consider the development of model job descriptions which reflect this new reality. Significant economies would accrue from using a set of common job descriptions between similar institutions (see Document 4, Section 6).

Guideline 6.4

Develop model job descriptions for biodiversity information management technical support personnel, in conjunction with other institutions.

7 REFERENCES

Ashworth, C. and Goodland, M. 1990. *SSADM: A Practical Approach.* McGraw Hill.

Chen, P. 1976. The Entity-Relationship Model - Toward a Unified View of Data. *ACM Transactions on Database Systems 1.* pp. 9-36.

Daniels, A. and Tate, D. 1984. *Practical Systems Design.* Pitman.

Kroenke, D.M. 1992. *Database Processing.* Maxwell Macmillan.

Richardson, B.J. 1994. The Industrialization of Scientific Information, Systematics and Conservation Education (ed. P.L Foley, C.J Humphries, and R.L Vain-Wright), *Systematics Association Special.* **50**:123-31. Clarendon Press, Oxford.

UNEP 1990. *INFOTERRA Thesaurus of Environmental Terms (3rd ed.)*, United Nations Environment Programme, Nairobi.

UNEP 1993. *The Guidelines for Country Studies on Biological Diversity.* United Nations Environment Programme, Nairobi, Kenya.

Walter, K.S. 1989. Designing a Computer Software Application to Meet the Plant-Record Needs of the Arnold Arboretum. *Arnoldia* **49**(1):42-53.

ANNEX 1: LIST OF ACRONYMS & ABBREVIATIONS

ANSI	American National Standards Institute
BCD	Biological and Conservation Data
BDB	Biodiversity Data Bank
BDM	Biodiversity Data Management
CASE	Computer Aided Software Engineering
CBD	Convention on Biological Diversity
DBMS	Database Management System
DDL	Data Definition Language
E-R	Entity Relationship
ERIN	Environmental Resources Information Network
GEF	Global Environment Facility
GIS	Geographical Information Systems
GUI	Graphical User Interface
ITF	International Transfer Format
IUCN	International Union for the Conservation of Nature and Natural Resources
LAN	Local Area Network
NGO	Non-Governmental Organisation
PAT	Polygon Attribute Table
PC	Personal Computer
RDBMS	Relational Database Management System
ROI	Return on Investment
SQL	Structured Query Language
UNEP	United Nations Environment Programme
WAN	Wide Area Network
WCMC	World Conservation Monitoring Centre
WWF	World Wildlife Fund/World Wide Fund for Nature

NB See also the index of acronyms and abbreviations in the *Resource Inventory* (Document 4).