A comprehensive reference model for biological collections and surveys

Walter G. Berendsohn[1], Anastasios Anagnostopoulos[2], Gregor Hagedorn[3], Jasmin Jakupovic[4], Pier Luigi Nimis[5], Benito Valdés[6], Anton Güntsch[1], Richard J. Pankhurst[7] & Richard J. White[8]

Summary

Berendsohn, W. G., Anagnostopoulos, A., Hagedorn, G., Jakupovic, J., Nimis, P. L., Valdés, B., Güntsch, A., Pankhurst, R. J. & White, R. J.: A comprehensive reference model for biological collections and surveys. – Taxon 48: 511-562. 1999. ISSN 0040-0262.

The article describes an extended entity-relationship model covering biological collections, i.e. natural history collections of biotic origin; data collections used in floristic or faunistic mapping, survey, and monitoring projects; live collections such as botanical or zoological gardens, seed banks, microbial strain collections and gene banks; as well as novel collection kinds such as of secondary metabolites or DNA samples. The central element in the model is the unit, which stands for any object containing, being or being part of a living, fossilised, or conserved organism. The unit may be gathered (observed or collected) in the field and derived units may recursively emerge from it through specimen processing, breeding or cultivation. In addition, units may form associations (e.g. host/parasite), ensembles (lichen on a rock with fossils), and assemblages (herd, artificial grouping). Gathering events, specimen management (acquisition, accession, storage, preservation, exchange, ownership), and taxonomic or other identifications relate to the unit and are treated in detail. Geographic and geo-ecological data have not been fully modelled; taxonomic (name) data and descriptive information are treated by reference to other published models.

1. Introduction

Global electronic access to biodiversity information has been identified as a priority task within the biological sciences (Anonymous, 1998a). Many data collections of widely different content and on very diverse scales have been constructed or are being planned, to deal with – or refer to – data on organisms. Examples are international or local checklists of organisms, floristic or faunistic mapping projects, and phytogeographic databases. The global scope of this effort has been emphasised by Soberón & al. (1996) and by the OECD Megascience Forum's Working Group on Biological Informatics (Anonymous, 1999e).

The easy access to information made possible by the Internet and particularly the World Wide Web has emphasised the problem of data quality. Scientific names as

[1] Botanischer Garten und Botanisches Museum Berlin-Dahlem, Freie Universität Berlin, Königin-Luise-Str. 6-8, D-14191 Berlin, Germany.

[2] Athens International Airport S.A., Technical Division, Environmental Control, 5th km Spata-Loutsa Ave. Gate 1, GR-19004 Spata, Greece.

[3] Biologische Bundesanstalt für Land- und Forstwirtschaft, Institut für Pflanzenvirologie, Mikrobiologie und Biologische Sicherheit, Königin-Luise-Str. 19, D-14195 Berlin, Germany.

[4] AnalytiCon AG, Haus 17, D-14473 Hermannswerder, Germany.

[5] Università di Trieste, Dipartimento di Biologia, Via Giorgieri 10, I-34127 Trieste, Italy.

[6] Universidad de Sevilla, Facultad de Biología, Departamento de Biología Vegetal y Ecología, Avda Reina Mercedes s/n, E-41080 Sevilla, Spain.

[7] Royal Botanic Garden, 20A Inverleith Row, Edinburgh EH3 5LR, U.K.

[8] University of Southampton, School of Biological Sciences, Biodiversity and Ecology Division, Southampton SO16 7PX, U.K.

the key index to biodiversity information have been scrutinised and found defective, and concepts were developed to remedy that situation (Berendsohn, 1995, 1999a). However, in most groups of organisms the only truly reproducible method to circumscribe a taxon is by means of its specimen contents. Making accessible the tremendous amount of information represented by the estimated 2.5 billion specimens (Duckworth & al., 1993) stored in natural history collections is thus a priority, and many projects to computerise natural history collections are under way. This will not only be of importance for biological systematics, but also aid other branches of basic science and its practical applications in many ways.

Recent developments in the field of information integration favour an interconnection of different databases with similar contents, thus preparing the way for a federate approach: common access to databases maintained and owned by the information providers (e. g. individual museums). However, a major obstacle to the process of connecting several databases into an efficient information network lies in the fact that even quite similar databanks are often organised on the basis of very different data structures. Projects like Species 2000 (Bisby, 1997) have managed to cope with this difficulty by means of a minimal common denominator approach. Common data structures and standardisation of data content may also overcome this limitation. However, computer scientists tend to believe that the forces promoting heterogeneity will always prevail over the desire to conform to an external standard (Blum, pers. comm.). The exposition and adoption of common conceptual models, like the one proposed in this article, is a prerequisite of truly transparent common access to distributed databases.

Another problem is created by the innovation speed in hard- and software technology, which entails regular data migration to new systems. Each new implementation of a database is an opportunity to improve the accuracy and flexibility of the underlying data structures. A well documented and analysed conceptual data model of shared information provides the cohesive force that is indispensable in a federate system.

Scope. – Information models which provide project-independent data structures are needed to design and compare databases that include biological data. CDEFD ("a Common Datastructure for European Floristic Databases"; Berendsohn & al., 1997b), a concerted action project financed under the European Commission's third framework programme, set out to provide such models to the biological community and to database designers. The group started by modelling herbarium collections, then included live collections (botanical gardens), and soon realised that there are no fundamental differences in information structure between biological collections at large. Consequently, the model was widened in scope to cover all kinds of biological collections.

The information structures presented here are rather complex, attempting as they do to incorporate into a single model all available information. To fit the particular needs of a given data bank, they can easily be modified and simplified. The model allows the designer to assess the consequences of the simplification process, particularly in regard to restraints on future extensions of information content and possible incompatibilities with other databases. The complex model thus provides a reference tool for the planning of specific databases and of interfaces to connect differently designed databases. In addition, the model supplies guidelines for the definition of data fields and thereby provides a base for discussing data standards.

Subdividing the information. – One of the main problems encountered by CDEFD was the tendency of individual biological sub-disciplines to develop their own terminology, thus obscuring their common ground. The model covers a skeleton of a dinosaur in a museum, a bacterial strain in the freezer of a culture collection, a tree in a botanical garden, and a substance sample in a collection of secondary metabolite isolates. These rather disparate objects turn out to be very similar when their data handling and collection processes are scrutinised. Collections of specimens and notes taken during field surveys are mostly considered separately, but are completely congruent as regards field observations, information related to the observer/collector, locality information, and taxonomic information, so they should be treated jointly. Another separation that this model aims to overcome is the distinction between live, dormant, and preserved collections. There again, the central data structures turn out to be identical, so that no reason exists to treat the data separately. Fortunately, the fact that different codes of nomenclature apply to different groups of organism, and the existence of a wide spectrum of taxonomic views, are irrelevant in the context of collection information.

In an attempt to subdivide the information, the following partially congruent large-scale data areas were identified:

- Field data: information about the who and where of a collection or field record, and descriptors resulting from field observations.
- Collection management data: information directly relating to the management of stored objects.
- Descriptors: all data which are the result of a process of observation or analysis carried out on what we term a “biological object” (i.e. a name, a potential taxon, a specific collection or observation site, an organism in the field, or a physical object in a collection; Berendsohn & al., 1997b).
- Taxonomic identification and nomenclatural data.

The present model is considered complete for biological collections of any scale and type, but it excludes taxonomy, bibliography, and descriptors, which are treated elsewhere (Berendsohn, 1994, 1995, 1997; Hagedorn, 2000). Place of storage and administration is completely covered. Field data are also analysed in detail; however, geographic and ecological collection site data proved to be too extensive to allow comprehensive coverage (see Section 4.2).

A draft of the model was presented in public by the first six authors during the European Science Foundation Workshop on “Disseminating Biodiversity Information” in Amsterdam, March 1996, then published on the Web (Berendsohn & al., 1996). The subject was considered again by a subgroup of the BioCISE project team (Biological Collection Information System in Europe; see Berendsohn & al., 1999), when the three additional authors joined in the revision of the model here presented.

Information modelling. – From the beginning, the CDEFD modelling effort was torn between two contradictory approaches. The purist’s approach to information modelling requires that analysis of the problem domain be entirely independent of implementation, because techniques and theoretical approaches change rapidly. On the other hand, people in need of organising their data here and now call for a function-oriented way of modelling, providing rapid solutions to imminent problems.

The model here presented is a compromise between the two approaches. On the one hand, it is the result of a comprehensive and time-consuming effort of data analysis and definition. For example, naming and delimitation of the model's major entities went through more than a dozen profound version changes in the course of the project. On the other hand, the rules of modelling, particularly those defined by Chen (1976) for entity-relationship models, were somewhat bent to achieve results significant for direct implementation. With the possible exception of subtypes, entities in the model can be transformed into relational tables, and keys are defined. In some cases, even implementation shortcuts are incorporated into the model or the discussion of the diagrams. However, the concepts described by the model are not restricted to a relational database system. Rather, the results of the modelling process may be incorporated into any database implementation.

2. *Modelling methods and terminology*

Entity-relationship model diagrams (ERD). – Berendsohn (1997) provides an introduction to modelling and to the interactive CASE techniques used by CDEFD. An ENTITY TYPE (designated by small capitals throughout this paper) can be thought of as a conceptual class of real-world objects or events (i.e., entities). The entity type defines a set of properties ("attributes") that are common to all instances (entities) of the entity type, each one having an individual 'value' for each "Attribute" (hereafter, in single and double quotes, respectively; the latter capitalised). For example, within the entity type PERSON NAME, 'Smith' may be the value of the attribute "Person last name". Transformed into a relational model, the entity types take the form of tables (relations), the column headers of the table are the attributes (e.g. "Person name"), every row in the table represents a record giving the values for each attribute. In ERD, two entity types may be connected by means of defined "relationships" between attributes ("keys") which are present in both. The relationships are read along the connecting lines, starting with the entity type name, followed by the descriptive text nearest to the other entity type, then the "cardinality" expressed by the shape of the arrowhead pointing to the second entity type, and finally the name of the second entity type. The cardinality expresses how many entities of the second entity type are referred to an entity of the first type. The cardinality may be 'exactly 1' (a filled-in single arrow), '0 or 1' (an open arrow), '1 to many' (a filled-in double arrow), or '0 to many' (double open arrow). The definition of cardinality as here employed also depicts "referential integrity rules", i.e. statements guaranteeing that a foreign key always corresponds to a primary key. Declarative referential integrity defined in a database is used to implement the cardinality types defined in the model. For example, in Fig. 1, declarative referential integrity rules are used to make it impossible to delete an entity of the type NAME RANK while there is still an entity of the type SCIENTIFIC NAME referring to it. In contrast, "data integrity rules" are semantic rules for the creation, deletion, or modification of records. These may have to be enforced by program code written for the specific application, although modern relational database systems allow the implementation of several types of data integrity rules within the database. In this presentation, several ERD (Fig. 3-15) illustrate different sections of the reference model. The tables listing the attributes for each entity type are cited it the figure's caption.

Tables. – In the tables describing entity types, each attribute is listed with its descriptive, long name and in some cases explanatory remarks and example values in the first column, the data type of the attribute in the second, and a short name (≤ 10 characters) in the third. In the case of "foreign keys" (i.e. attributes with values pointing to another entity), the long name is followed by the name of the related entity's key attribute in parenthesis. "Primary keys" of subtypes are similarly followed by the supertype's key attribute name. Foreign keys are marked by the suffix _Fk, primary keys by _Pk in the short name. The data types distinguished are 'int' for integer numbers, 'float' for values with decimals, for character data 'str' (up to a fixed length, e.g. 256 characters) and 'text' (almost unlimited, can normally not be indexed), 'bool' for yes/no values, 'date' for a complete date, and 'vdate', 'vtime', and 'vlength' for variable date, time and linear measurement data, respectively (see below). For independent primary key values the type 'int' was used, because it is less error prone in manual input.

Subtyping. – This term denotes the classification of some of the attributes of an entity "supertype" into several additional entity "subtypes". A subtype inherits all attributes of its supertype, but also has additional attributes (Anonymous, 1993). Another way to express this is that the supertype set is the intersection of the attributes of all subtype sets. In some cases it may be appropriate to assign some of the attributes of an entity type to a separate subtype, because they are used only under specific circumstances. For example, a collector as well as the author of a scientific name is a PERSON, but there are a number of attributes referring only to name authors or to collectors. Such "is-a" relationships may be expressed by classification relationships (PERSON being the supertype, AUTHOR OF SCIENTIFIC NAME and COLLECTOR its subtypes). Subtyping should also be applied when only certain entities can participate in a relationship, and others cannot. Conversely, if two or more entity types can participate in semantically equivalent relationships to other entities in the model, it may be appropriate to unite them under a common supertype (see example under agents in Section 6.3).

It is standard practice to include one or more classification variables in all entities that have subtypes or subclasses. These attributes indicate the subclass membership of every instance. A single attribute is used to handle sets of mutually exclusive subtypes, while multiple binary (boolean) attributes are used for subtypes that are not exclusive (Blum, pers. comm.). The latter kind of classification variables have generally not been specified explicitly in this model.

In diagrams, a supertype is connected to its subtypes with a line marked by a triangle. An example is given in Fig. 2. The triangle is empty if the relationship is inclusive (and/or, i.e. a person may be a collector and/or an author), it is filled if the relationship is exclusive. An equal sign within the triangle indicates that the classification is complete, i.e. only the subtypes shown exist (e.g. supertype AGENT with subtypes COMPANY OR ORGANISATION and PERSON TEAM, see Fig. 11).

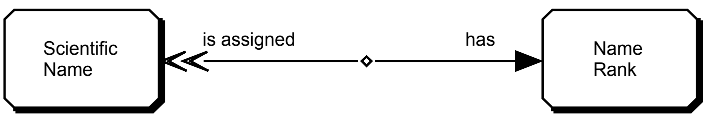

Fig. 1. ER-Diagram. Reads: a scientific name has exactly 1 corresponding name rank, a name rank is assigned to 0 to many scientific names.

Up to now, relatively few database management systems directly support subtyping. However, in traditional relational databases subtypes may be implemented either by directly including the attributes of all the subtypes in the supertype, or by including the attributes of the supertype in each of the subtypes, or by forming 1-to-C relationships between supertype and subtype (see Batini & al., 1992, for a discussion of the issue of "removing generalisation hierarchies").

Events. – Entity types bearing an "event" suffix contain or link data items, which describe a particular action or incident taking place at a defined point in time. They should not be confused with events acting on these data in an object oriented information flow model.

Terms and entities taken from other models. – The following entities are repeatedly used in the present model but are not elaborated here:

- TAXON NAME: the basic botanical name without any attached taxonomic or nomenclatural information, including hybrids, cultivars, races or strains, and unnamed taxa. (Treated in the IOPI-ISC information model: Berendsohn, 1997.)
- POTENTIAL TAXON NAME: taxon names with a circumscription reference, which clarifies the taxonomic concept of the name's application, including its classification, synonymy and nomenclatural status (see Berendsohn, 1995).
- REFERENCE TITLE, REFERENCE DETAIL, ACCOUNTING, ADDRESS: entire reference citations, including publications, databases, in-citations, and "informal references" such as personal communications or notes. (Treated in the draft IOPI-ISC information model: Berendsohn, 1994.) References, as well as the referenced entity types ACCOUNTING and ADDRESS should be treated by modular subsystems.
- Descriptors. Hagedorn (2000) describes a subsystem for descriptors.
- The entity types ISO-COUNTRY CODE and LANGUAGE are specified in the respective standards (Anonymous, 1997a, 1998f, 1999c; Anonymous, 1988, 1998e); the latter entity type should include the locally used language designation along with the translated equivalent. The implementation of MEDIA OBJECT strongly depends on the database system used, they may actually be stored in the database management system itself and thus become a data type specification, but they may also be referenced by a filename or a URL.

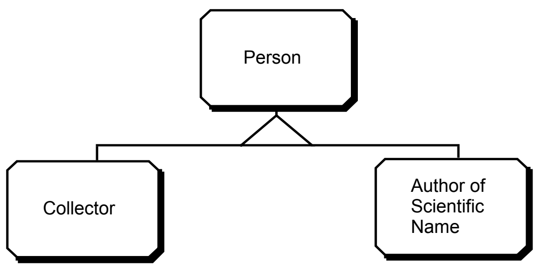

Fig. 2. Classification relationship (subtyping). Reads: a person may be a collector and/or author of a scientific name. Other subtypes may be defined, i.e. the classification is not exclusive (the triangle is not filled) and the classification is not complete (the triangle contains no equal sign). Compare Fig. 3, UNIT, for an exclusive and complete classification relationship.

Timestamping and metadata elements. – Each record should include timestamping information, which is automatically generated by the database management system. The minimum logging information should be a CreationDate, CreationOperator, UpdateDate, and UpdateOperator.

Metadata elements defining intellectual property rights and other properties of the data should be employed if data are exchanged. Some metadata apply to the individual attribute, some to entities, some to records, and most to entire data sets or output documents. For further reference see Anonymous (1999a), Weibel (1998), and Beach (1998).

Vdate, vtime, vlength. – For most date attributes a vague date type (vdate) is specified in this model. A vdate offers the possibility to enter a single date as well as a date range (first date to last date). Each date can be qualified ('approximately', 'year uncertain', etc.) and each part of a date may be missing. For example, the day may not be recorded, or the year may be unknown, as in the case of phenological data. Even cases where day and year, but not the month are known can arise, e.g. if a label is partly damaged. One also needs the option to define that a dated event happened before or after a given date, or that it happened in a certain month within a given period, e.g. 'June in the 1950's' (Lampinen, pers. comm.). If the database management system offers no support for a user-defined data type with these properties, each vdate may be implemented using a set of integer attributes (Date_D, Date_M, Date_Y, DateLast_D, DateLast_M, DateLast_Y) and text attributes for the qualifiers (Date_Q, DateLast_Q). A flag in combination with an open first or last date can indicate the 'before' or 'after' datum. Designation of a month or season within a period should probably be accommodated in a text field. A similar vague data type is defined for time. Vtime should accommodate exact or qualified single points in time of the day or ranges, in hour, minute and seconds. Vlength is used for all linear measurements, within this model particularly for altitude and other vertical location measures of the collection site. Apart from ranges (also incomplete or discontinous ranges), it should also provide the possibility to input measures of error.

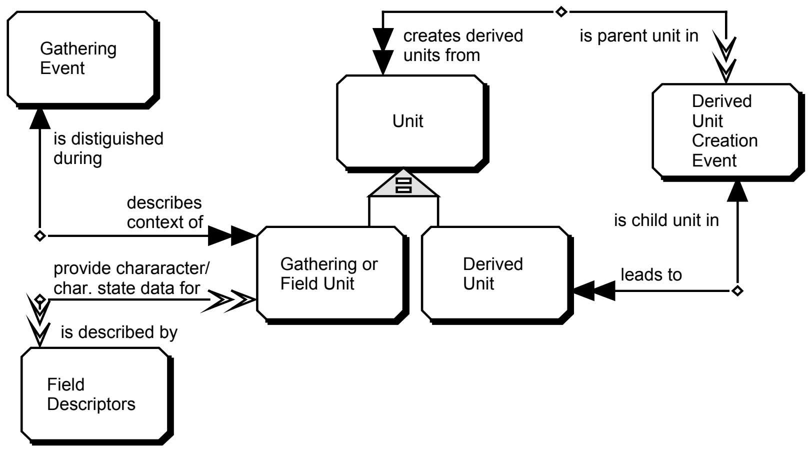

Fig. 3. Core ER-Model. Attribute definitions for GATHERING EVENT: Table 1; UNIT: Table 6; GATHERING OR FIELD UNIT: Table 7; DERIVED UNIT: Table 9; DERIVED UNIT CREATION EVENT: Table 10. The descriptor subsystem is here depicted as an entity type (see Section 2).

3. *Framework of the information model*

Units. – Fig. 3 illustrates the core of the information model. In some cases, we had to resort to the definition of new terms untainted by traditional use and loose definition. The concept of a "unit" as a physical object in the field or in a collection, or its "virtual" representation in an electronic medium, is central to the model. It embraces organisms observed in the field, soil samples taken, herbarium specimens, animals in a zoological garden, microbial strains, and even pure substances in a natural products collection. Field data, taxonomic identifications, curatorial activities, collection management data as well as all kinds of descriptive data are linked to units. We used the term "collection" when referring to the location and ownership of sets of physical units, not for the purpose of obtaining units in the field, which we refer to as gathering.

Field data. – The entity type GATHERING EVENT links the data on the observations or collected items with information about the gathering, including Field descriptors and site details (Fig. 4-6). The supertype collection event defined in the ASC model (Anonymous, 1993) is roughly analogous to this construct.

The entity type GATHERING OR FIELD UNIT, a subtype of UNIT, is the central interface for all field data belonging to a particular object or observation or a set of these. Normally, every unit is given a specific identifier, the field number (e.g. a collector's or recording number). The data are recorded while all collected materials are still united in the field (where they remain in the case of records in surveys). The delimitation of the gathering or field unit (i.e., establishing which observations or materials collected in a gathering event form a unit) depends on the collector's criteria and on the kind of organism (one or several taxa may be involved). While a collector of vascular plants or higher vertebrates will typically assign a field number to every object collected within a gathering event, students of e.g. ichtyology (Blum, pers. comm.), lichenology or phycology often unite all materials collected during a gathering event under a single number. In this case the field number effectively becomes an identifier of the gathering event.

The selection of gathered field information strongly depends on the objective of the sampling or observation. For example, ethnobotanists will use a field descriptor set which accentuates plant uses (Fig. 6), a dendrologist will record relatively abundant information about trees, while for some other botanist recording the life form of a plant may be important. Typically, such data found here refer to features that cannot be observed in the collected materials themselves or are likely to change, e.g. the height of a tree from which a herbarium specimen was taken, or many colour characteristics (see Section 4.4). Such data should be referred to a descriptor subsystem, if gathered in detail. However, in inventory projects carried out e.g. for floristic mapping field descriptors may be lacking, only taxonomic identity and presence/absence data being recorded.

Derived units. – A derived unit represents a physical item in a specific collection. From the gathering or field unit one or more derived units may be created (which in turn may again give rise to new derived units), e.g. segregation of duplicate herbarium specimens, accessioning in botanical gardens, sampling of fungi from leaves in a botanical garden, vegetative propagation in live collections, etc. Merging of several units is also possible, e.g. through sexual reproduction in live collections, resynthesis of lichens, mating studies in fungi, or mixing seed samples. Administrative data such as accession coding, storage, transactions (loans etc.) are linked to the DERIVED UNIT. The DERIVED UNIT CREATION EVENT is an entity type containing all data describing the creation of derived units, such as date of event, curator or scientist responsible for the event, and applied method (the latter handled by an entity type of its own, DERIVED UNIT CREATION METHOD).

4. Field data

In gatherings and observations we distinguish four main data areas:

- gathering event data refer to the act of collecting or observing organisms at a given site, i.e. time, person, and project data, as well as such locality data as are considered to be directly dependent on the event.
- The gathering site is a complex construct (geo-ecological subsystem) encompassing all geographical and ecological data that describe the area or point location where the organisms were collected or observed, when considered to be independent of a specific gathering event.
- Gathering or field unit data represent the non-descriptive field information specific to each item that was distinguished by the collector or observer during the gathering event (e.g. items distinguished by means of different field numbers).
- Field descriptors are the results of observations made in the field on the item specified in the entity type GATHERING OR FIELD UNIT. Field descriptors are deferred to the descriptor subsystem.

4.1. The gathering event

Fig. 4 illustrates the GATHERING EVENT within the ER model. The attributes used to describe the event are detailed in Table 1. The "Gathering date or period" is an attribute of the data-type vdate, i.e. it can accommodate incomplete dates or periods (frequently encountered in historical collection labels) as well as qualifiers ('c.'). The "First collector" is normally a single person, in charge of the field book to which the field number refers; or, in the case of unorthodox numbering schemes, it may be a collector team or institution, etc. Thus the link refers to an entity of the type AGENT. If the numbering refers to a project, expedition, or "platform", the link to PROJECT OR EXPEDITION is used. [A "platform" (see also Section 6.3) is a named inanimate object that is employed to conduct gathering events, e.g. a research vessel (Anonymous, 1993).] "Additional collectors" and "Per-collector" refer to PERSON TEAM. The field numbering bears no relation to "Additional collectors", who in herbarium labels may be listed after the first collector and field number, sometimes following a 'with' or 'cum'. The "Per-collector" is an amateur or casual collector who collected specimens for the first collector, who takes the credit for collecting by assigning his own field number. Croft (1992) introduces this attribute because "it is not possible to insert this type of information in any manner that does not misrepresent the situation".

Possession of specimens from some countries may be illegal if a valid permit (see Table 15) has not been obtained. Project or expedition information resides in a separate entity type (Table 1), where data such as project title(s) and sponsor acknowledgements (often used on herbarium labels), or descriptive information such as the purpose of a recording event (see Copp, 1998) can be stored.

A unit may in some specific cases be related to more than one site (e.g. seed collections from specific plots or trees in forestry, which were subsequently mixed). Although this practice discards information on among-site variation (and was thus considered scientifically unsound by one of the reviewers), such cases do occur, and can be interpreted as a merging of separately collected gathering or field units into a mixed derived unit (see Section 5.2). Alternatively, the gathering site may be defined in such a way as to include all individual plots, with a list of the plots put into the note field ("Stated locality").

Site information and the gathering event. – Site information may be time-dependent to varying degrees. For example, for insect collections climatic conditions like current rainfall or humidity may be important information describing the gathering site. Area names, although less short-lived, may change in time as well. Field

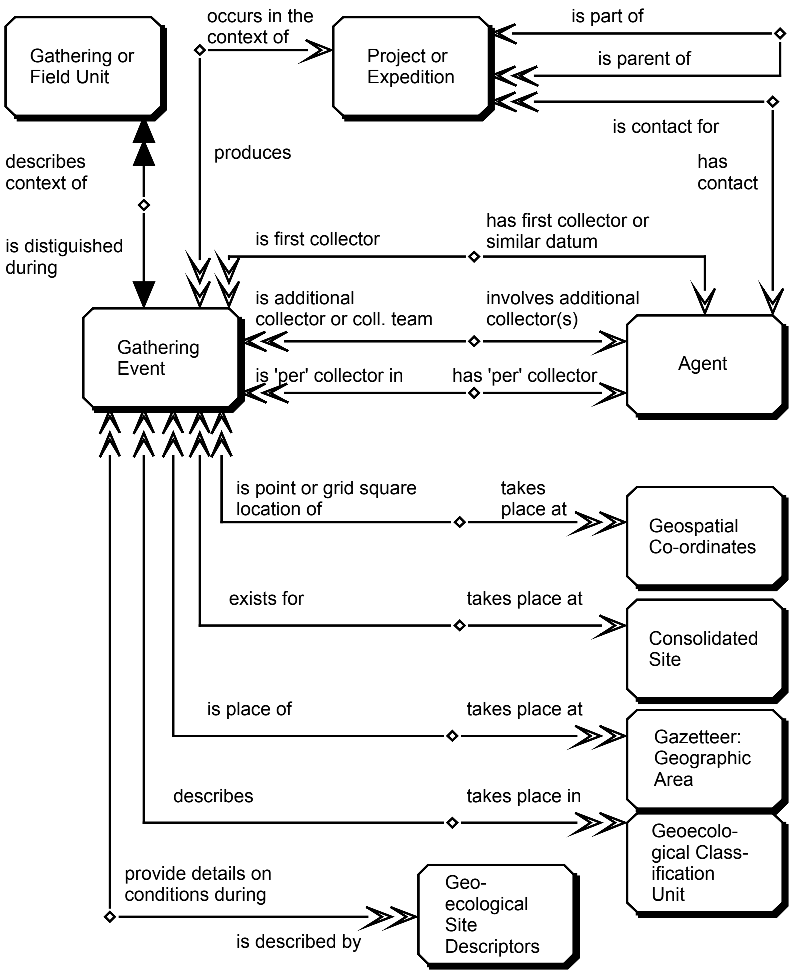

Fig. 4. Relationships of the GATHERING EVENT. Attribute definitions for AGENT: Table 14; GATHERING EVENT and PROJECT OR EXPEDITION: Table 1; GATHERING OR FIELD UNIT: Table 7; geo-ecological site descriptors: subsystem (see Section 2); remaining entity types: Table 3.

Table 1. Gathering event attributes (long name, data type [see text], short name).

Attributes of entity type GATHERING EVENT		
Gathering event key	int	GEvt_Pk
Gathering date or period	vdate	GDate
Gathering time or period	vtime	GTime
First collector(s) (person team key)	int	LegTm1_Fk
Additional collector(s) (person team key)	int	LegTmAl_Fk
Per-collector(s) (person team key)	int	LegTmPr_Fk
Project or expedition key	int	Proj_Fk
Stated locality	str	StatedLoc
Consolidated site key	int	Csite_Fk
Co-ordinates key	int	Crd_Fk
Permit key	int	LegPrmt_Fk
Attributes of entity type PROJECT OR EXPEDITION		
Project or expedition key	int	Proj_Pk
Key to the more inclusive project of which the present record is part (project or expedition key)	int	ParProj_Fk
Project or expedition title	str	ProjTitle
Project or expedition subtitle	str	ProjTitle2
Project or expedition platform name	str	PlatformNm
Project or expedition duration	vdate	Duration
Project or expedition funding acknowledgements	str	ProjAckn
Project or expedition description and notes	text	ProjTxt
Contact person for the project (Agent key)	int	Contact_Fk

records of such data are therefore best placed within the GATHERING EVENT entity type. However, many collectors and projects may wish to record site data independent of gathering events, because a given site is visited frequently (e.g. long term observations). Moreover, several gathering events may take place at the same time and place (several first collectors collecting different groups of organisms). As a consequence, the model provides an entity type for consolidated site data. In addition, users seeking information from a collection information system will often approach a system using geographical or geo-ecological search criteria, so that this information should ideally be highly structured.

We tried to solve these apparent contradictions. As one of the reviewers (Blum) suggested, we included a free text attribute "Stated locality" in the GATHERING EVENT, which captures the original field information on the collection site as noted or assigned by the collectors in the field. This attribute is also particularly useful for data entry in historic collections, where the stated locality of a gathering site may have to be extensively scrutinised to allocate a meaningful location, not to speak of a geo-ecological classification. It also allows verification of subsequent allocations of locality data. Such allocations are effected by links to the geo-ecological subsystem, either indirectly (by reference to a CONSOLIDATED SITE) or directly to GAZETTEER: AREA NAME, GEO-ECOLOGICAL CLASSIFICATION UNIT, GEO-SPATIAL CO-ORDINATES or geo- ecological site descriptors. Electronic data capture systems are now available that allow to establish such links to stored information directly during field work. However, as the data defined in these entities may be changed independent of the gathering event itself, it is important to copy any data selected to the "Stated locality" attribute.

Specialised collections may rely on a very specific sampling scheme, for example, vouchers taken from numbered trees in a forest sample plot, microbial samples from a defined spot on a dunghill, or core samples. Such data will normally be specified in the attribute "Small-scale locality description" of the GATHERING OR FIELD UNIT entity type. Information like "Depth or height of sampling relative to surface level" (e.g. water depth, height of epiphyte on tree) belongs here as well, if it is specific to an individual unit. Otherwise, it is accommodated by the GATHERING EVENT ("Stated locality" and references to the geo-ecological site descriptor subsystem).

A location ascription of a particular unit may be doubtful. Lampinen (pers. comm.) describes interesting cases from Finnish survey and herbarium data. As school children have to collect plants for class, they often re-use the specimens of their seniors and just invent the label data. Cases exist of a collector deliberately falsifying label data to get credit for interesting finds. A specimen label may have been accidentally mounted on a wrong sheet. In collections, such cases may be dealt with by adding annotation labels (e.g. "locus confirmationis indigens", in Helsinki). In the model, a flag may be set ("Gathering site doubtful"), and details can be stored in the notes attribute ("Unit notes") of the UNIT entity type. These data belong to the unit because they can be attributed to both a gathering or field unit in a survey, or to a derived unit in a collection).

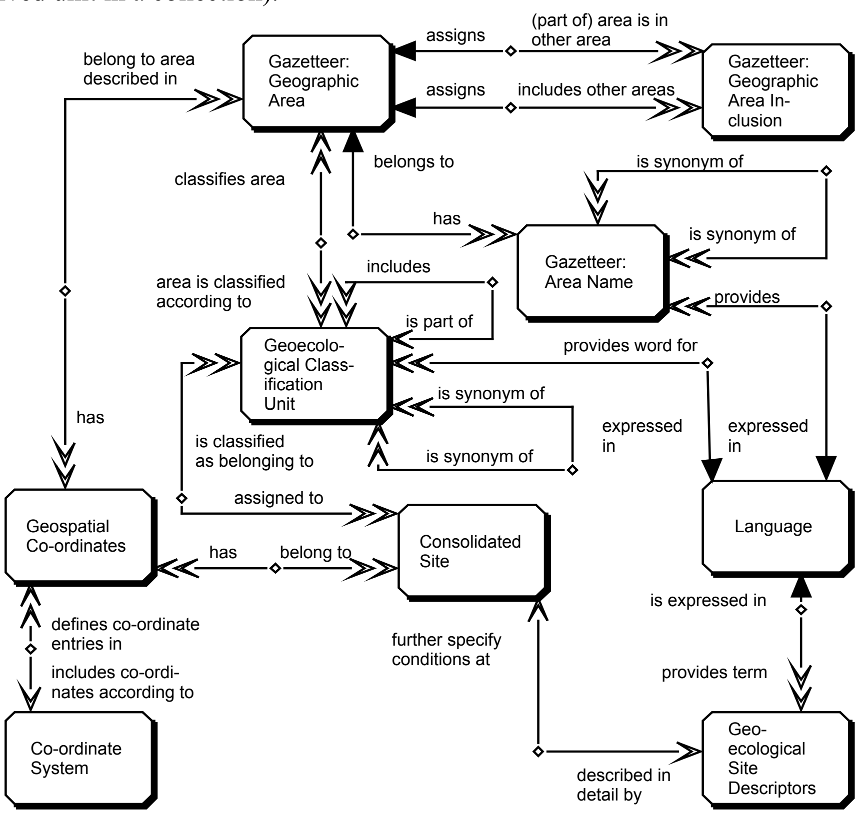

Fig. 5. Gathering site related entity types. Attribute definitions in Tables 2-3; LANGUAGE: see text. The descriptor subsystem is depicted as if it were an entity type (see Section 2).

Table 2. Co-ordinates attributes (description, data type, short name; Section 2).

Attributes of entity type GEOSPATIAL CO-ORDINATES		
Co-ordinates key	int	Crd_Pk
Co-ordinates measurement method (e.g.: GPS, GPS with local reference, from map in field, from map later)	str	CrdMsmtMtd
Co-ordinate source map reference key (reference detail key)	int	CrdMap_Fk
Details about the system used for point and grid co-ordinates (co-ordinate system key)	int	CrdSyst_Fk
Point co-ordinates x-value	float	CrdXVal
Point co-ordinates y-value	float	CrdYValue
Point co-ordinates precision qualifier (e.g.: c., about, estimated)	str	CrdPrecQlf
Point co-ordinates value absolute error	int	CrdValErr
Grid system description	text	GrdSystTxt
Grid system name	str	GrdSystNam
Grid cell code	str	GrdCode
Grid cell code assignment precision qualifier (see text)	str	GrdPrecQul
Altitude measurement	vlength	AltVal
Altitude unit of measurement (e.g.: m, ft)	str	AltUnit
Altitude measurement method (e.g.: GPS, barometric altimeter)	str	AltMsmtMtd
Altitude reference system	str	AltRefSyst
Attributes of entity type CO-ORDINATE SYSTEM		
Co-ordinate system key	int	CrdSyst_Pk
Co-ordinate system name	str	CrdSystNam
Co-ordinate system description (information on the system which is not accommodated by the other attributes, e.g. the geodetic datum)	text	CrdSystTxt
Co-ordinate system prefix for x-value	str	RrFxX
Co-ordinate system suffix for x-positive value	str	SffxXPos
Co-ordinate system suffix for x-negative value	str	SffxXNeg
Co-ordinate system prefix for y-value	str	PrFxY
Co-ordinate system suffix for y-positive value	str	SffxYPos
Co-ordinate system suffix for y-negative value	str	SffxYNeg
Co-ordinate system number formatting rules	str	NumFormat
Co-ordinate system default flag	bool	DfltSyst

4.2. Geographical and ecological data

Framework. – The geo-ecological subsystem (Fig. 5) consists of the entity types GAZETTEER: GEOGRAPHIC AREA, GEO-ECOLOGICAL CLASSIFICATION UNIT, and SITE CO-ORDINATES, as well as the geo-ecological site descriptor subsystem. As mentioned above, these may be linked directly to gathering events, thus providing access to all such events that have been originally classified as belonging to one of the items described (e.g. a certain country, a specific ecological classification unit, etc.). At the same time, they serve to describe and access synthesised consolidated sites, to which gathering events may be assigned.

A detailed and complete coverage of all data items that may be incorporated into the geographical and ecological site description would exceed the scope of this model. The US Federal Geographic Data Committee (Anonymous, 1994b, 1998c) lists more than 300 individual data elements and compound elements for geo-spatial data alone. A draft standard biological data profile of the content standard for digital geo-spatial metadata is under discussion (Anonymous, 1998b). Many more references to existing standards, metadata collections and GIS specifications were compiled by Bryan [illegible] (1998). The IOPI team came to the conclusion that for extensive coverage of geographical and other data related to field collections, a database system should rely on one of the available commercial GIS (geographical information system) programs. However, as the GIS option still imposes a considerable load on an institution’s informatics resources, the model will preferably have a relational structure. The model presented here is intended to mirror current usage of geo-ecological site data and thus to aid in the definition of requirements for a linked GIS or for the basic structure to be implemented in a proprietary database design.

Table 3. Gazetteer attributes (description, data type, short name; see Section 2).

Attributes of entity type CONSOLIDATED SITE		
Consolidated site key	int	GSite_Pk
Consolidated site user defined name	str	GSiteUNam
Consolidated site individual locality description	str	GSiteDtl
Originator of site definition (agent key)	int	GSiteOr_Fk
Attributes of entity type GAZETTEER: AREA NAME		
Area name key	int	GANam_Pk
Area category name	str	GACategNam
Area name	str	GANam
Language key	int	Lang_Fk
Area name default flag. native language designation	bool	DefltNam
Attributes of entity type GAZETTEER: GEOGRAPHIC AREA		
Area key	int	GA_Pk
Area subtype designator	str	GASubtDsg
Area boundary	text	GABoundary
Area validity time period	date	GAValidTme
Area lowest and highest point elevation	vlength	GAAltRange
Altitude unit of measurement (e.g.: m, ft)	str	GAAltUnit
Area co-ordinates minimum x-value	float	GACrdXMin
Area co-ordinates maximum x-value	float	GACrdXMax
Area co-ordinates minimum y-value	float	GACrdYMin
Area co-ordinates maximum y-value	float	GACrdYMax
System of min./max. values used for data validation purposes (co-ordinate system key)	int	CrdSyst_Fk
Area is land and/or water marker	str	LandWater
Area description	text	GAFreeTxt
Area data source reference (reference title key)	int	GARef_Fk
Attributes of entity type GAZETTEER: GEOGRAPHIC AREA INCLUSION		
Child area (area key)	int	GAChild_Fk
Parent area (area key)	int	GAPar_Fk
Child area only partially included flag	bool	GAPartial
Attributes of entity type GEO-ECOLOGICAL CLASSIFICATION UNIT		
Geo-ecological classification unit key	int	GECU_Pk
Geo-ecological classification unit subtype designation	str	GECSubtDsg
Geo-ecological classification unit category	str	GECCateg
Geo-ecological classification unit name	str	GECUNam
Geo-ecological classification unit detailed description	text	GECUDtl
Geo-ecological classification source reference (reference title key)	int	GECRef_Fk
Language key	int	Lang_Fk
Higher unit in classification (geo-ecological classification unit key)	int	GECUPar_Fk

Geo-spatial co-ordinate data. – Geo-spatial co-ordinates define "a place in the modern world" (Anonymous, 1993), either in the form of point locations (latitude, longitude, altitude), or in the form of grid locations. The former are represented by a flat data structure, while grid data may include a hierarchical element. All co-ordinate data are unequivocal as long as the base system (including projection and geodetic datum; see Banta, 1999) and the method of measurement is cited (see Anonymous, 1999d, for extensive information). As a gathering site can be expressed simultaneously with several types of co-ordinates, a many-to-many relationship exists between the entity types GATHERING EVENT and GEO-SPATIAL CO-ORDINATES.

From the data processing point of view the assignment of co-ordinates is the most satisfying site demarcation. For specimen collections, the geographic latitude-longitude system with the geodetic datum defined by WGS84 (World Geodetic System 1984; see Anonymous, 1997b) may be used as the default. It provides global coverage, is accurate, and is very widely used, although for national surveys a national co-ordinate system may seem to be more practical (printed maps!). Global positioning systems (GPS) now permit to measure directly latitude-longitude data (cf. Dana, 1998). Powerful tools for the transformation of existing gazetteer data exist (e.g. Anonymous, 1998d, 1998g), although caution is necessary as the precision of gazetteer co-ordinates is seldom specified. However, more than one point co-ordinate system may be used in surveys and for specimens, and many mapping projects prefer using a grid system, which relates the presence/absence data for a surveyed species to a defined area (usually a grid cell). In many cases, specimen label or field book information will not suffice to assign accurate co-ordinates. The attribute "Stated locality" in the GATHERING EVENT entity type serves to capture original information, which may or may not be converted or interpreted at a later stage.

For users outside the realm of systematics, the main way of accessing collection information is via geographical queries. To present uniform information on a given area, site information should ideally conform to a single co-ordinate system. However, functions that transform one co-ordinate system into another are notoriously complex, and data that include several different types of co-ordinates are not handled by today's off-the-shelf GIS programs. Automated management of different co-ordinate systems is a task for specialised system modules, which depend on extensive mathematical algorithms (see Voser, 1999). We thus devised a model capable of accommodating different kinds of original geographic co-ordinate data found on collection labels or in survey data sets, while recognising that these data in most cases cannot be directly accessed by a geographical information system.

Point locations for all geographical co-ordinate systems (including polar co-ordinates) can be expressed by a combination of the co-ordinate system name, the altitude, and two floating point numbers. In geographical co-ordinates, north vs.

Table 4. Characterisation of geographical co-ordinates in the entity type CO-ORDINATE SYSTEM.

Attribute	*Value*	*Attribute*	*Value*
Co-ordinate system name	Geographic co-ordinates	C. system description	Geodetic Datum: WGS84
Prefix-for-X	Latitude	Suffix-for-X-positive	N
Suffix-for-X-negative	S	Prefix for Y	Longitude
Suffix for Y positive	E	Suffix for Y negative	W
Number formatting	DMS	Default system	True

south and east vs. west can be expressed by positive and negative values, respectively (ISO, 1983). In a general implementation, the definition, data entry rules, and formatting rules for the x- and y-value can be defined in the entity type CO-ORDINATE SYSTEM (Table 2). The attribute "Co-ordinate system description" may accommodate further descriptors of a specific system, e.g. the geodetic datum (Daxinger, 1998). For geographical co-ordinates the entity would contain the information given in Table 4. Choices for number formatting are decimal degrees (DD), decimal minutes (DM), or decimal seconds (DMS) – e.g. 61.25670° (DD), 61° 15.4021' (DM), 61° 15' 24.1" (DMS). The input formatting routine for degree values should accept floating point values in the degrees or minutes part (e.g. 41.50° should be understood as 41° 30'), thus allowing a mixed data entry of degree-minute-second, degree-minute (with fractional part), and degree-decimal. The "Default system" attribute indicates preference of the system for data input purposes.

Grid references may be expressed similarly, but their values can be numeric (UTM, e. g. 35 N 6762000 N 456789 E) or alphanumeric (MGRS; 35JMH5678862000, and British National Grid). These can be converted into float data type, but then data on grid cell size and precision must be stored. A simpler solution is here proposed: A flag is set to indicate that the entry represents a grid cell ("Grid flag"). The attributes "Grid system name" (e.g. 'German MTB' [Meßtischblatt]), "Grid cell code", ('7413/14'), and "Grid system description" (in some cases needed to give further details, such as the geodesic datum) suffice to define a location by means of a grid cell.

Several attributes are provided to express the precision of the data. The "Co-ordinates measurement method" refers to the source of horizontal point co-ordinates and/or grid values, and may hold entries such as 'manual measurement from a 1 : 20,000 map', 'GPS', 'GPS with local reference'. The "Measurement error" contains a numeric value read as plus-minus measurement error of the co-ordinate values. "Precision qualifiers" include expressions such as 'about' or 'c.'. The grid precision qualifier is used to record uncertainty in grid assignment or proximity to grid line (indicating possibility of duplicate recording in different grid cells). For altitude, the 'vlength' data type covers error and modifiers. However, the unit of measurement must be cited, the measurement method often differs from that used for horizontal co-ordinates, and the reference system (e.g. 'mean sea level', or 'AMSL'; see Anonymous, 1996) may be known.

Gazetteer data. – Gazetteers refer to a wide variety of areas types, i.e. bounded, continuous or sometimes discontinuous, named portions of the earth's surface that are delimited by political, administrative, traditional, geomorphological and/or ecological boundaries (see Copp, 1998, for an excellent analysis of sites and bounded areas in biological surveys). Named areas are often part of a more or less well-defined hierarchy, they may change over time, and many synonyms as well as homonyms may exist.

In the entity type CONSOLIDATED SITE (Table 3), the attribute "User defined site name" provides a shortcut to a previously used defined site in an implemented system (Humphreys, pers. comm.). It can also be used to store personal gathering site numbers ('locality RL245/1998'), or those of an expedition ('Finnish botanical expedition to W Turkey in 1991, locality 25'), as well as self-made gazetteer names ('forest area between the roads from A to B, B to C and C to A'). Descriptive information on site position that cannot be accommodated by means of the gazetteer or co-ordinates are accommodated by the attribute "Site location detail" (e.g. 'Road-side, road between Jucurán and Casas Viejas, about 2 km from Jucurán'). The attribute "Small-scale locality description" of the entity type GATHERING OR FIELD UNIT allows further textual specification of the individual collection site. Data on the boundary of the site may be added by analogy to the respective attributes in the entity type GAZETTEER: GEOGRAPHIC AREA ("Area boundary", "Area elevation", and area co-ordinates minimum and maximum values).

The gathering event and the consolidated site can be linked to one or several geographic areas or geo-ecological classification units (Table 3). For individual geographic area names and categories, names in different languages may exist (attributes "Language", "Area category name", "Area name"). One of these synonyms must be marked by means of an "Area name default flag" for a defined system. The "Area category name" is a designation that may or may not be cited with the name, e.g. 'department', 'kreis', 'municipio', 'eparchia', 'island', 'TDWG botanical country'. Area categories may be necessary to identify a specific area (e.g. 'New York': 'city' and 'state').

Area validation data can be used for quality control. It is necessary to cite the temporal validity period, because the circumscription of named areas may change over time. For example, the 'Federal Republic of Germany 1949-1989' differs significantly from the area covered by that name after 1989. The entity type GEOGRAPHIC AREA INCLUSION handles the relationship between the two, as well as the relationship between the reunified Germany and the former German Democratic Republic. The actual circumscription (planar and altitudinal) of the area may be expressed in a GIS (and, nowadays, in some relational database management systems) as a series of vectors describing the perimeter of the area in some kind of coordinate system or by reference to a scanned image ("Area boundary"). In practice this should be handled by a GIS system. The maximum-minimum data for coordinates and altitude, provided for here, do not attain such a high level of data validation but can be stored easily in a normal database and are useful for input control.

Geo-ecological classification units. – Geo-ecological classification units are named classes of areas distinguished by some more or less sharply defined climatic, edaphic, geomorphological, geological, palaeontological, or synecological characteristics. Outside of published systems, little standardisation exists. By and consequence, many commonly used terms may be equivocal. Geo-ecological classification data include a defined category, e.g. 'formation', its value, e.g. 'gallery forest' and, preferably, a bibliographic reference detailing the system used (e.g. 'Beard, 1946'). A classification usually involves hierarchy, so a pointer to a higher category may be used.

Geo-ecological site descriptors. – Geo-ecological site descriptors are individual measurements or observations of ecological parameters at the collection site itself, or synthesised data referring to a consolidated site. They can be referred to the descriptor subsystem because their basic structure (name and value) is analogous to e.g. morphological descriptors (character and character state). Examples of site descriptors are aspect (slope and direction), exposure ('open', 'shady'), measures for water flow, physical substrate properties (nutrients, pH, etc.), other microclimatic and soil or water conditions (see Copp, 1998), the thickness of a sediment layer sampled, or geological age parameters (Anonymous, 1993). As with all descriptors, especially those involving measurements, a link to a system or an entity detailing methodology may be needed (compare Berendsohn & al., 1997a). As mentioned above, site descriptors may refer to individual gathering events or may be synthesised, forming part of the collection site data.

Consolidated sites. – These are sites that were not defined in the field, but are to provide access to information stemming from one or more gathering events. They represent derived information, based on knowledge on these events but incorporating data from other sources (gazetteers, itineraries, etc.). Thus, the consolidated site may be created independently from the actual gathering event, and a person or team is to take responsibility for its creation ("Site originator").

Subtyping. – Because of the widely varying degree of details recorded, the general model of the gathering site data must be kept flexible, to allow for adaptation to specific needs without sacrificing over-all compatibility. Subtypes enable the use of ad-hoc-defined tables, and standard data tables may be incorporated as they become available. Such subtypes have not been included in the diagram. Examples are:

- For the entity type GAZETTEER: GEOGRAPHICAL AREA: ISO countries (Anonymous, 1997a, 1998f, 1999c), TDWG recording units (Hollis & Brummitt, 1992; new version in preparation); see Berendsohn (1999b) for further references. Also, national and regional lists of administrative areas, protected areas, and properties (land parcels).
- For GEO-ECOLOGICAL CLASSIFICATION UNIT: formations (physiognomic vegetation classification), life-zones (Holdridge, 1967), biomes (Walter & Breckle, 1983), plant sociological units (syntaxa, e.g. Braun-Blanquet & Fuller, 1929), FAO-UNESCO soil classification (Anonymous, 1974), new soil taxonomy (Anonymous, 1975, 1998h, 1999f), CORINE potential natural vegetation (Anonymous, 1987a), geologic time units, rock units, biostratigraphic zonation units (Theodor & Lindberg, 1996).
- The geo-ecological site descriptor set may consist of edaphological, climatological, limnological, and lithological descriptor sets.

The attribute "Area subtype designator" is a classification variable that tells the application program which of the possible area subtypes is to be included. Analogous subtype designations are given for geo-ecological classifications and for ecological descriptors.

4.3. The gathering or field unit

Function and circumscription. – The entity subtype GATHERING OR FIELD UNIT provides an interface to field-oriented data considered by the collector or observer to belong to a specific subset of data or materials accumulated in the gathering event. Several samples and/or observations may be made in a gathering event. In larger organisms, the gathering usually pertains to one or more individuals belonging to the same taxon. Yet in other groups several taxa will be included, e.g. a piece of rock with several species of lichens, a water sample with microscopic algae, or a soil sample for microbiological investigation.

The option to refer directly to observations for which no voucher specimens were obtained can be used for presence/absence statements in biological recording projects. When material is collected, the gathering or field unit data are those which are recorded before the object becomes part of a collection. However, as the collector and curator is often one and the same person, this definition is somewhat arbitrary. The gathering or field unit, together with the information connected with the gathering event and the field descriptor sets, can best be viewed as the information contained in a collector's field book. In higher plants, the individual gathering is routinely identified by a field number in the collector's field notes.

For parasites, data on the host or substrate are often recorded without actually sampling the substrate. In this case, parasite and substrate are separate gatherings of the same gathering event, which are associated (see under Section 5.1).

Gatherings made within collections (e.g. herbarium specimens collected in a botanical garden) are normally treated as derived unit creation events (see Section 5.2). However, a new gathering or field unit results if the provenance of the first organism is irrelevant to the second one (e.g. the gathering of a fungal parasite on a long-cultivated plant in a greenhouse).

Data elements. – Table 7 details the attributes of the entity subtype GATHERING OR FIELD UNIT. The attributes comprising the field number refer to the first collector's or observer's field book numbering. At least in higher plants it is standard practice to assign a single sequential number, but some individual collectors have preferences of their own. To accommodate any type of numbering, a prefix and a suffix are provided, as well as a field for different identifier schemes or for lot-identifiers as used in the sampling of micro-organisms. This also accommodates institutional series, which often consist of sequential numbers. However, in historical collections and collections of fungi, algae, or animals, this information may be incomplete or lacking. Depending on the degree of implemented detail, output routines have to be provided to once more concatenate the data. Curatorial additions to field numbers (e.g. suffixes assigned to the parts of a mixed specimen) are treated in Section 5.2 under *Derived unit identifiers*.

If collectors divide material among them, each assigning a field number, different gatherings for the same material are created. Such a relationship can be expressed by a unit association.

The gathering method is usually evident in higher plants, but in some cases notes may be taken (e.g. drying temperature, when the of material is destined for chemical analysis). In algae (e.g. 'dredge from a boat'), micro-organism (e.g. 'mixed soil sample from 1 square meter'), or zoological collections this kind of data is often required, and a separate entity type may be commendable to accommodate standard methods. For presence/absence data, aerial photographs may be used and recorded as the method used to gather the information.

4.4. Field descriptor sets

Field descriptors are a group of data items which have been obtained by direct observation of the unit at the collection site. In larger units, such data are often recorded in the body text of the label (attribute "Field description text" of entity type GATHERING OR FIELD UNIT).

CDEFD initially attempted to develop a set of descriptors for plants. However, even in this restricted field the type and number of descriptors recorded separately may vary greatly among collectors and according to the research interest pursued. The selection of morphological descriptors also depends heavily on the observed taxon. Often, no clear separation can be made between field descriptors and those used for specimens in a collection; therefore such data should be linked directly to the unit. However, a number of characters can only be recorded in the field, such as size measurements of larger plants, some colour characters which may be lost in preserved material, occurrence quantification, phenological characters, and local ethnobiological information.

The development of computerised authority files of descriptors, to define the terminology to be used for characters and their states, is a very important task. Its difficulty is illustrated by the so far unsuccessful attempts by the TDWG Descriptors Subgroup (Higher Plants) to develop a minimal set of morphological descriptors. However, attempts to cope with more narrowly delimited taxa have been successful (e.g. FlyBase controlled vocabulary: Anonymous, 1999b; Grasses: Watson & Dallwitz, 1994). It is not the purpose of this model to propose such sets of characters. The model allows for the addition of special-purpose field descriptors in user-defined entity types. Moreover, the present model can be used to define a collection subsystem in a larger context, which can include a system for descriptive information (see Hagedorn, 2000).

While standardisation of morphological characters is notoriously difficult, for economic and other uses of organisms such standards exist (e.g. Cook, 1995) and

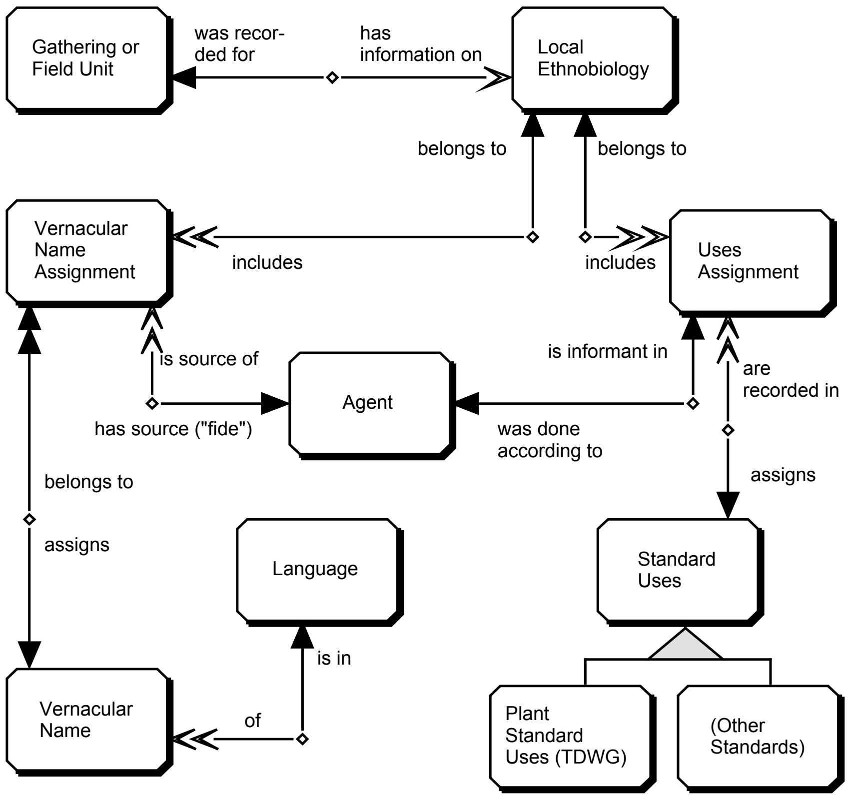

Fig. 6. Ethnobiological field information. Attributes for GATHERING OR FIELD UNIT: Table 7; Agent: Table 14; all other entity types except LANGUAGE: Table 5.

Table 5. Ethnobiology and vernacular names attributes (description, data type, short name; see Section 2).

Attributes of entity type LOCAL ETHNOBIOLOGY		
Local ethnobiology key	int	LoclEth_Pk
Local uses description	text	LoclUses
Attributes of entity type STANDARD USES		
Standard uses key	int	StndUse_Pk
Standard uses subtype designator	str	StndUseDsg
Attributes of entity type USES ASSIGNMENT		
Local ethnobiology key	int	LoclEth_Fk
Standard uses key	int	StndUse_Fk
Fide reference (person team key)	int	Source_Fk
Attributes of entity type VERNACULAR NAME		
Vernacular name key	int	VernNam_Pk
Vernacular name	str	VernNam
Language key	int	Lang_Fk
Attributes of entity type VERNACULAR NAME ASSIGNMENT		
Local ethnobiology key	int	LoclEth_Fk
Vernacular name assigned (vernacular name key)	int	VernNam_Fk
Assignment according to (person name key)	int	Fide_Fk

should be followed wherever possible. Recording such uses at the collection site becomes increasingly important especially in tropical countries. Useful local ethnobiology information (Fig. 6) for which a structure can be defined also includes the vernacular name of the plant. For both standard uses and vernacular names, a source should be cited, as the question of intellectual property rights must not be ignored.

5. *Units*

Definition and concepts. – A unit is a physical object either in the field or in a specific collection. The term "specimen" can often be used as a synonym; however, it lacks a precise definition and consequently has caused some confusion in the modelling process. The term "collection units" was used by Wilson (1993) in a modelling context, although with a different definition. The "collecting unit" used in the ASC model (Anonymous, 1993) for physical objects is functionally analogous, although a fixed number of classes (subtypes) is used and the concept of the field unit (see Section 4.3) is not included.

The general context of the UNIT is given in Fig. 3. The following list provides the principal properties of the unit and its linked entity types, which are depicted in Fig. 7 and in other diagrams specified in the text.

- The delimitation of a unit is defined in the process of the gathering, curation, preparation, or cultivation of materials.
- While still in the field and unprocessed, the material is referred to as a gathering or field unit (see Section 4.3).
- In the process of curation, preparation, cultivation, or transfer of materials, derived units are created. This is the process referred to as a derived unit creation event (see Section 5.2 and Fig. 8).

Table 6. Unit attributes (description, data type, short name; see Section 2).

Attributes of entity type UNIT		
Unit key	int	Unit_ID
Unit is derived flag (classification variable, indicating the subtype derived unit when set to true)	bool	DerivFlag
Material category key	int	MtCateg_Fk
Owned by (agent key)	int	Owner_Fk
Unit notes	text	UTxt
Unit data transcribed flag	bool	Trnscr
Gathering site doubtful	bool	GSiteDbt
Attributes of entity type UNIT DATA SOURCE		
Unit key	int	Unit_Fk
Data source category (e.g.: field book, original entry, literature, label)	char	SrcCateg
Unit data transcribed by (person team key)	int	Trnscr_Fk
Unit data transcription notes	text	TrnscrNote
Original source text of transcription	text	SrcText
Attributes of entity type MATERIAL CATEGORY		
Material category key	int	MtCateg_Pk
Material category name	str	MtCategNam
Material category description	text	MtCategTxt

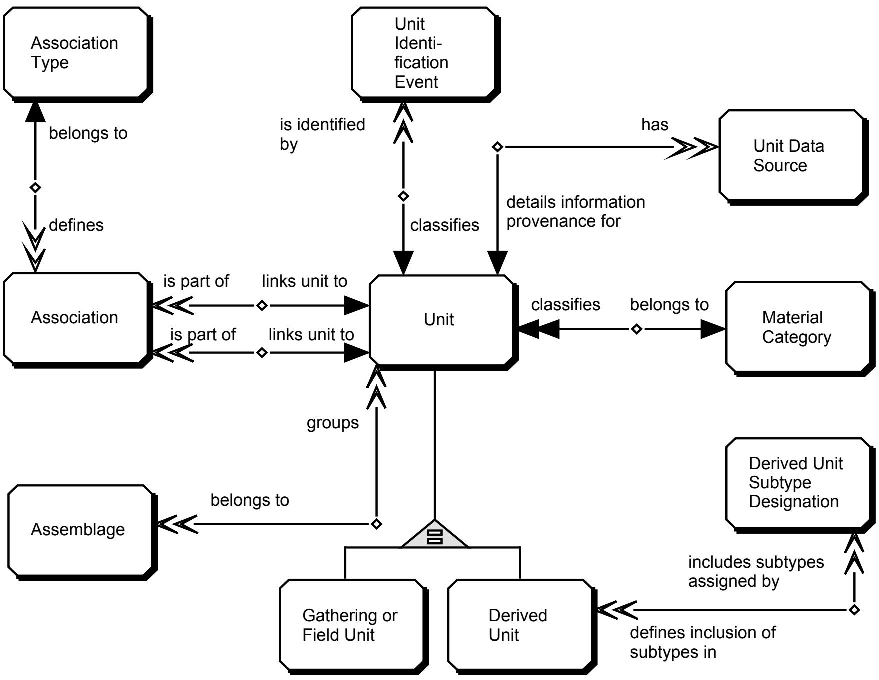

Fig. 7. Some unit-related entities. Attributes for ASSOCIATION and ASSEMBLAGE: Table 8; UNIT, UNIT DATA SOURCE and MATERIAL CATEGORY: Table 6; UNIT IDENTIFICATION EVENT: Table 17; DERIVED UNIT: Table 9; DERIVED UNIT SUBTYPE DESIGNATION: Table 11.

Table 7. Gathering or field unit attributes (description, data type, short name; see Section 2).

Attributes of entity type GATHERING OR FIELD UNIT		
Gathering or field unit key (unit key)	int	GFldU_Pk
Gathering event key	int	GEvt_Fk
First collector's field number prefix	str	FldNoPrfx
First collector's sequential field number	int	FldNo
First collector's field number suffix	str	FldNoSffx
Other field identifier	str	FldCode
Gathering method	str	GMtd
Small-scale locality description	str	SmSclLoc
Field description text	text	FldTxt
Provenance type (e.g.: cult. ex wild, cult. non wild, unknown, wild, cultivated)	str	ProvType
Depth or height of sampling relative to surface level	int	RelAlt

Table 8. Association and assemblage attributes (description, data type, short name; see Section 2).

Attributes of entity type ASSOCIATION		
Association type key	int	AssnTyp_Fk
First unit in association (unit key)	int	Unit1_Fk
Second unit in association (unit key)	int	Unit2_Fk
Association doubtful	bool	AssnDoubt
Attributes of entity type ASSOCIATION TYPE		
Association type key	int	AssnTyp_Pk
Association type (e.g.: living on, parasite on, host of)	str	AssnTyp
Association type (e.g.: living on, parasite on, host of)	str	AssnTypInv
Association type description	text	AssnTxt
Attributes of entity type ASSEMBLAGE		
Assemblage key	int	Assem_Pk
Assemblage name	str	AssemNm
Assemblage description	text	AssemTxt

- A unit may be identified (i.e. being classified) by means of a unit identification event (see Section 7 and Fig. 14). Identification may consist of an assignment to a class within a specific classification system (taxonomic identification, chemical substance identification, etc.), or of the assignation of a special, well defined purpose to the unit (e.g. nomenclatural type), or of a free assignment to a class or term (unconstrained identification event), or a combination of these.
- A relationship between units may be expressed by means of an association or an assemblage of units (see under Section 5.1).The derivation history (Section 5.2) and ensemble sets (Section 6.2) are additional possibilities for derived units only.
- Collection management data are mostly linked to derived units (see Fig. 12). Each derived unit has at most one storage location (collection or subcollection). The physical media, container, or mounts may be described as the storage medium (Anonymous, 1993).

– Each derived unit may include one or more subtypes, freely definable entities containing attributes not included in the main UNIT or DERIVED UNIT entity types. The DERIVED UNIT SUBTYPE DESIGNATION determines which unit subtypes apply (Fig. 9).

5.1. The unit supertype

Data common to all units are defined in Table 6. Every unit can be uniquely identified by the "Unit key". Every unit in a collection belongs to an owner ("Owned by", a reference to an agent, see Section 6.3), not necessarily identical with the "Collection where unit is stored" (an attribute of DERIVED UNIT). For example, a permanent loan may be owned by one institution but stored and handled by another, which effectively acts as if it were the owner. Ownership of field units is rarely specified, but survey data records may have an owner. Such unit-level metadata assignments can be managed by the creation of derived units.

Unit information may be classified to contain confidential data. For example, for a rare plant the information about the existence of the plant may be public, but the point co-ordinates and exact locality description may not be freely accessible. Data security is intimately related to the implementation of a system and is typically deeply embedded in application and/or server programming code. We have hence refrained from specifying attributes in the model.

Unit data source. – The entity type UNIT DATA SOURCE (Table 6) is used to document the way in which the information was recorded. The attribute "Data source category" defines the method. The "Unit data transcribed flag" in the entity type UNIT is set in case that the recorded information was transcribed, i.e. made to fit the attributes of the database (as opposed to original data entry). Further cases thus marked include the addition (e.g. hierarchical levels for geographic references), conversion (e.g. old spelling to new, calendar dates), and correction of data (e.g. correctable errors in co-ordinates). The person responsible for the transcription is recorded, who might add notes indicating the types of transcription or data change effected. The "Original source text" can contain the entire label text as on the label. Alternatively a picture may be used (entity type MEDIA OBJECT) to provide a view on the entire original information unaltered by transcription processes. Pictures have been used in the South-eastern Regional Floristic Information System (R. R. Haynes, pers. comm.), where herbarium label information was scanned, transcribed into full text, and finally assigned to the respective attributes. The graphic information is also useful in the case of older specimens, where labels tend to disintegrate, or where the handwriting is important. Another example for media objects serving as data sources are sound records such as of bird calls, serving as quasi-vouchers for survey information.

Association and assemblage. – Where units are so intimately related that they are usually cited together, an association or an assemblage is defined. However, care should be taken to restrict use of this feature to groupings that cannot be retrieved from other data in the system (e.g. unit derivation).

We have to distinguish between symmetric and asymmetric relationships. For example, the host-parasite relationship is asymmetric: A is parasite of B, B is host of A. The relationship expression changes when the relationship is inverted. In contrast, a mutualistic or commensal relationship is symmetric (e.g. a shrimp and goby living in the same burrow).

Associations express asymmetric relationships between units. They can be used to express a variety of ecological or physical (many-to-many) relationships between units. Entities of the type ASSOCIATION (Table 8) define pairs of associated units from which the complete association set can be compiled. The attributes "Association type" and "Association type inverted" specify the relationship, e.g. 'living on' / 'substrate of'; 'parasity of' / 'host of'; 'pollinator of' / 'pollen source for'; 'mycorrhiza of' / 'mycorrizal host of'; 'parasitic egg in nest' / 'host nest of'). Field observations of parental relationships also belong here, same as mating pair definitions ('is female mate of' / 'is male mate of') gathered from field observations (breeding and cultivation is handled by the unit derivation process). If the association construct is used to describe multiple field observations of the same organism ('first observed as' / 'again observerved as'), the first sighting is designated as the root to which all further observations point. However, this case may also be considered symmetric and accommodated by an assemblage, since the sequence of observations can be retrieved from the gathering data and time. The "Association doubtful" flag can be used to express insecurity (e.g. a toadstool is collected and presumed to be a mycorrhizal fungus of a nearby observed tree).

While the association defines an asymmetric relationship between units (mostly pairs of units), the assemblage serves to handle symmetric relationships, mostly among several units, according to a single criterion ("Assemblage name", "Assemblage description"). A typical example is a herd of animals, either in the wild or in a zoological garden, or any kind of symmetric symbiontic, mutualistic or commensalistic relationship . Other applications of assemblages are batches of eggs taken from the same nest (however, the egg-nest relationship is an association!) and parts of the same material to which two collectors have assigned their own collection numbers (unfortunately a rather frequently encountered practice among botanists).

The ensemble set (see Section 6.2) is a special case of assemblage (physical association) which is treated separately because of its importance in specimen administration and storage.

Material category. – Each unit belongs to exactly one material category (see Table 6), describing the kind of material contained in the unit. The category can be a rather broad term (e.g. 'herbarium material') or very specific (e.g. 'dried and pressed plant material' or 'photograph'). If organs or organisms are stored separately, the organ designation is part of the material category (e.g. 'wood sample', 'excrement'). The degree of categorisation depends on the degree of detail which is to be recorded. Collection managers should carefully consider what categories to include; on the one hand simple categories facilitate data capture, on the other hand later addition of separate categories may involve the revision of the entire collection.

5.2. Derived units and derived unit creation events

The derived unit represents a physical item which is or has been present in a collection. A process of curation, preparation, cultivation, breeding, or a transfer event may give rise to one or more derived units from one (or more) parent unit(s). The model does not limit the number of iterations of this process. It is thus possible to store highly iterative processes, such as cultivation and propagation histories or pedigrees. The DERIVED UNIT's attributes are specified in Table 9, those of the DERIVED UNIT CREATION EVENT and the DERIVED UNIT CREATION METHOD in Table 10. The relationships of the entities is depicted in Fig. 8.

The derived unit creation method. – This entity type provides a dictionary of available methods which are used in the creation of new units. For in-house preparations, collection managers will establish their own subtypes of methods, if necessary. For example, in chemical substance collections several standard extraction methods may be specified. In the preparation of diatom specimens, various separation methods may be used. If a recipe structure (i.e. a sequence of applied methods) is deemed necessary, a cn : cn relationship to DERIVED UNIT is to be implemented. Alternatively, a new derived unit may be created for every procedural step. However, for many collection types a single standard method covers almost all preparations (e.g. 'separation' for herbarium specimens). The attribute "... short name" serves to rapidly select available methods, while the "... method description" gives room for detailed explanations. Two not so obvious albeit very important "methods" to create new derived units are the acquisition of material from an external source (in which case the entity types UNIT and UNIT TRANSFER must be linked, see Fig. 12) and the taxonomic determination of heterogeneous material: for every different taxon identified a separate derived unit is to be created.

The derived unit creation event. – For every derived unit, the DERIVED UNIT CREATION EVENT provides a link to the immediate "Parent unit". The "Person responsible for the event" (e.g. a curator) may differ from the "Person effecting the event", e.g. a laboratory technician. The "... event date" refers to the point in time the unit was physically created, or received at the collection. The "... inheritance type similar flag" expresses the similarity of the parent data with the derived unit's data. A derived unit is considered similar to the parent unit if it is assumed to be genetically similar (for examples see Section 5.3).

Three further attributes related to the creation of the derived unit belong to the DERIVED UNIT entity type itself. The attribute "Gathering or field unit shortcut" provides a means to bypass the derivation history and to directly access field data (see under 5.3, Implementing derived unit creation events, below). The "Tag for members of a similar unit set" eliminates the need to inspect the derivation tree for membership, thus facilitating e.g. the attribution of taxonomic identifications (see Section 7.1). The attribute "Derivation link certain flag" can be set to indicate that the link to the parent unit can be trusted. The flag is set to 'true' when the derivation from the parent is securely known, it is set to 'false' when there might be additional units might be in-between. This setting allows to properly mark cases of incomplete

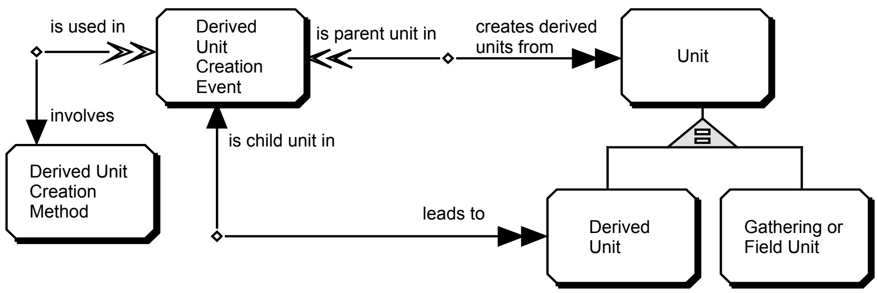

Fig. 8. Derived unit creation event. Attributes for UNIT: Table 6; DERIVED UNIT: Table 9; DERIVED UNIT CREATION EVENT and DERIVED UNIT CREATION METHOD: Table 10.

Table 9. Derived unit attributes (description, data type, short name; Section 2).

Attributes of entity type DERIVED UNIT		
Derived unit creation event key	int	DUCrEvt_Fk
Additional curatorial identifier	str	AddCurID
Preparation identifier (a laboratory identifier other than an accession no. or storage location; usually an internal number used in the creation process, e.g. isolation or laboratory number in culture collections)	str	PrepID
Gathering or field unit shortcut (unit key)	int	GFU_Fk
Storage medium key	int	StorMed_Fk
Unit on permanent loan flag	bool	PermLoan
Temporary storage flag	bool	TmpStorFlag
Collection where unit is stored (collection or subcollection key)	int	Colln_Fk
Storage location detail (e.g.: 3A12, lower right corner of herbarium sheet, drawer 1; not for taxon name under which the unit is stored: see StorNm_Fk)	str	StorLocDtl
Name under which unit is stored (taxon name key)	int	StorNm_Fk
Tag for members of an ensemble set. Unit key	int	EnsemblTag
Tag for members of a similar unit set. Unit key	int	SimUnitTag
Access restriction category (e.g.: only to defined institutions, only to ..., generally available; refers to access to material, not to information access or confidentiality)	str	RestrCateg
Access restriction set by (person team key)	int	RestrBy_Fk
Access restriction date	date	RestrDate
Access restriction notes	text	RestrNote
Derivation link certain	bool	DerivCert
Derived unit presence in collection (e.g.: available, on loan, in quarantine, lost, sold, given away as gift, given away in exchange, discarded, dead and discarded, consumed as experimental material)	str	Presence
Derived unit specimen count	int	SpecmCount
Derived unit specimen count type (e.g.: actual, approximate, at least)	str	SpecmCType

Table 10. Derived unit creation event and method attributes (description, data type, short name; see Section 2).

Attributes of entity type DERIVED UNIT CREATION EVENT		
Derived unit creation event key	int	DUCrEvt_Pk
Person responsible for the event (person team key)	int	DUCrRsp_Fk
Derived unit creation event date	date	DUCrDate
Derived unit creation method key	int	DUCrMtd_Fk
Parent unit (unit key)	int	ParUnit_Fk
Derived unit creation inheritance type similar flag	bool	InhTypeSim
Person effecting the event (person team key)	int	DUCrTec_Fk
Attributes of entity type DERIVED UNIT CREATION METHOD		
Derived unit creation method key	int	DUCrMtd_Pk
Derived unit creation method short name	str	DUCrMtdNam
Derived unit creation method description	text	DUCrMtdTxt

or questionable derivation histories. For example, microbial cultures are frequently sent from one collection to another collection. The recipient collection has information about the sender collection and (usually) the gathering. Yet, often it is not known whether the sender collection received the material directly from the field collector, or from a third collection. The sender collection data must then be entered in the first derived unit after the gathering or field unit, with “Parent link certain flag” set to ‘false’.

The derived unit. – The DERIVED UNIT subtype of the entity type UNIT contains attributes (Table 9) that are specific to the individual processed physical unit (as opposed to the field unit), but at the same time general enough not to be consigned to a subtype of their own (see Section 5.4). Attributes relating to the location and storage (medium, ensemble sets) of the unit are treated in Section 6.2.

Derived unit identifiers. – The primary means to identify a derived unit in the system is its primary key (“Unit key”), which is also the only actually unique identifier assigned. However, this is normally a number invisible to the user, and several schemes may be employed in the work process to identify a derived unit. Accession system coding is detailed in Section 6.1, loan and exchange codes under 6.4. In addition, a “Preparation identifier” may be used to store temporary identification numbers, e.g. laboratory numbers for a preparation, an isolation number for a microbe, etc. These numbers are often used in research notes or laboratory protocols before permanent accession numbers are assigned. They are not necessarily unique even within a laboratory, but may sometimes apply only in a given project. The “Additional curatorial identifier” is used e.g. to distinguish an individual plant on a herbarium sheet that was found to originate from a different collection site, or to belong to a different taxon, than the other plants on the same sheet. This identifier may take the form of a suffix to the field identifier, but it may also include some kind of description of the exact location on the herbarium sheet, or the co-ordinates of a diatom in a microscopic slide.

Unit availability. – The availability of a unit in the collection may be conditional, either because access restrictions apply, or because the unit is not present in the collection. Even if a unit ceases to exist as a physical object in the collection (because it has been processed into new derived units, or because it has been lost, destroyed or transferred elsewhere), the record is maintained as part of the curatorial history of the derived units. The “Derived unit presence in the collection” is explicitly recorded in the respective attribute (see Table 9), which is also needed to trace the fate of units in loans (together with associated derived unit creation events). An “Access restrictions category” may be assigned by law (e.g. pathogenic microorganisms) or by management decisions (e.g. because of ongoing research). The “Unit on permanent loan flag” indicates, for example, that a unit may not be given away as a gift.

5.3. Implementing derived unit creation events

The scheme developed for the unit uses a recursive structure to avoid fixed hierarchical levels. This allows for a very flexible approach, accommodating simple storage data as well as highly recurrent information such as cultivation histories. However, retrieving the data for a particular unit may be considerably complicated by having to step through a recursive structure of undefined depth. This problem is caused by the data model itself and must therefore be addressed here.

The application program should always offer the possibility to trace the events that led to a unit’s creation, and to look at the parent unit’s data. However, the derived unit should be able to stand on its own, i.e., data which refer to both parent unit and derived unit, should be accessible from the derived unit without having to step back to the parent unit's data. For field data, this is achieved by the shortcut link to the gathering or field unit. Unit-specific data, and those stored in entities linked to the UNIT, may or may not be copied from the parent to the derived unit (i.e., inherited by the latter).

Within the creation event, one of the following procedures applies to each datum of the parent unit:

- The datum remains unchanged, i.e. it is inherited by the derived unit.
- The datum usually remains unchanged, i.e. it may be inherited as a default but is to be revised.
- The datum must be changed because it is unique or otherwise specific to the parent unit; i.e. it can not be inherited.

A revision of all attributes of the unit and of its linked entities shows that the criterion of "biological similarity" (attribute "Derived unit creation inheritance type similar flag") is paramount to decide which of the above procedures is to be applied.

- Processes creating "similar" derived units, e.g.: division of soil samples; duplication or translocation of herbarium specimens or microbial strains; removal of samples.
- Derived units dissimilar, e.g.: separation of leaf containing a fungus from a phanerogam specimen for a fungal collection; isolation of a microbial strain from a soil sample; separation of a specimen from a mixed collection (e.g. a specimen of a different taxon mixed with others on the same herbarium sheet).

To automate the derived unit creation process an additional entity type may be implemented, which determines the inheritance procedure for every combination of attribute (or entire entity type), inheritance type, and derived unit creation method. This implementation has the advantage of being easily adaptable to varying implementations of the data structure (e.g. different descriptor sets attached to the gathering).

5.4. Derived unit subtypes

Units in biological collections range from palaeontological samples and herbarium specimens to animals in a zoological garden, bird's nests, microbial isolates or chemically pure natural substances. A variety of attributes exist that are specific to only some of these. Subtyping the DERIVED UNIT provides a possibility to assign such attributes flexibly. It also allows for the extension of the model to include other types, which had not yet been considered. This is done by appending the new type to the entity type UNIT SUBTYPE DESIGNATION (giving it a name and providing a description) and adding a new UNIT subtype. Subtyping derived units is depicted in Fig. 9, the corresponding attributes are provided in Table 11. Apart from the UNIT SUBTYPE DESIGNATION itself, all entity types listed here are examples, any number of additional subtypes may be defined, and several subtypes may be linked to a single unit. In the logical model of the Museum of Vertebrate Zoology (Blum, 1996) many such attributes can be found. However, most of these can be accommodated by the unit-unit relationships defined above (associations, assemblages, derivation) or by a descriptor subsystem, if these features are implemented.

Derived unit cultivation or breeding. – In this derived unit, parentage and technical propagation information are handled by the entity types DERIVED UNIT CREATION PROCESS and DERIVED UNIT CREATION METHODOLOGY. However, attributes may be needed to alert technical staff to special procedures, or store experience gained with the unit. Often a single attribute, "Cultivation or breeding unit notes", will suffice.

Derived unit herbarium label. – This subtype contains attributes that may also apply to labels in zoological collections. The application program may offer to automatically assign to these attributes values from gathering information (e.g. project sponsors as label footer, collector or 'Flora of' and country as title, etc.). Other types of collections, especially those for which very small labels are used (entomological collections), may not need such a subtype at all.

Derived unit quantification. – In e.g. chemical substance collections or seed banks, where the availability of material may depend on the quantities in store, a subtype DERIVED UNIT QUANTIFICATION may be useful. If measurements are stored, the measurement unit (cm, g, ml, etc.) is to be recorded. If the measurement unit is left empty, the quantity is a count (e.g. duplicates in an insect collection, number of vials in a preservation batch of micro-organisms). This kind of data may also be used to detect depletion of type specimens by the removal of "kleptotype" fragments: one of the present authors uses to record the weight before sending types on loan.

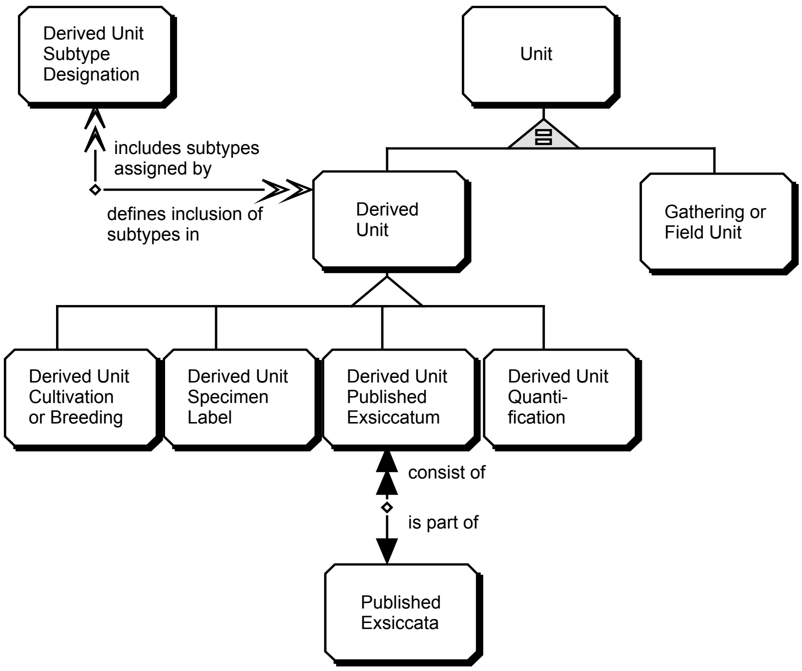

Fig. 9. DERIVED UNIT subtypes. Attributes for UNIT: Table 6; DERIVED UNIT: Table 9; all other entity types: Table 11.

Table 11. Derived unit subtypes attributes (description, data type, short name; see Section 2).

Attributes of entity type DERIVED UNIT SUBTYPE DESIGNATION		
Derived unit subtype key	int	USubt_Pk
Unit subtype name	str	USubtNam
Unit subtype description	str	USubtTxt
Attributes of entity type DERIVED UNIT CULTIVATION OR BREEDING		
Cultivation unit notes	memo	CultNote
Next scheduled transfer	date	NextTransf
Attributes of entity type DERIVED UNIT PUBLISHED EXSICCATUM		
Exsiccatum number	str	ExsNo
Exsiccata serial number	str	SerNo
Exsiccatum taxon (taxon name key)	int	TaxNam_Fk
Published exsiccata key	int	Exs_Fk
Attributes of entity type PUBLISHED EXSICCATA		
Published exsiccata key	int	Exs_Pk
Exsiccata name	str	ExsNam
Exsiccata fascicle number. Number of the present exsiccata within the entire series	str	ExsFascNo
The publication reference for the exsiccata (reference title key)	int	ExsRef_Fk
Published exsiccatum booklet (reference title key)	int	ExsBook_Fk
Additional published notes on the exsiccatum (reference title key)	int	ExsNotes_Fk
Exsiccata serial numbers (range of numbers, when the duplicates of each exsiccatum are numbered (e.g.: Vanky 1979-)	str	ExsSerNos
Attributes of entity type DERIVED UNIT QUANTIFICATION		
Quantity measured	int	QuanVal
Quantification unit of measurement	str	QuanUnit
Quantification value error	int	QuanValErr
Quantification value precision qualifier	str	QuanValQfr
Quantification alternative textual value (e.g.: large, minute, small, much)	str	QuanTxtVal
Attributes of entity type DERIVED UNIT SPECIMEN LABEL		
Label later text additions	text	LblAddTxt
Label first title	str	LblTitle
Label subtitle	str	LblTitle2
Label footer	str	LablFooter

Published exsiccata. – They are specific to botanical collections, where they play a major role. Exsiccata (*plantae exsiccatae;* Latin terminology according to Hawksworth & al., 1995) are numbered sets of duplicate specimens, which are published under an "Exsiccata name". Entire sets or individual items are usually distributed to several herbaria and thus serve as reference specimens. Publication may occur in numbered fascicles, which may be published in different years and thus are treated as different entities of the type REFERENCE TITLE in a publication series. Every exsiccatum (i.e. the individual specimen) is a unit with an "Exsiccatum number", which refers to a specific identification ("Exsiccatum taxon"). Strictly speaking, the supertype-subtype relationship between UNIT and DERIVED UNIT PUBLISHED EXSICCATUM exists only for the bulk unit prior to distribution. However, since it is part of the definition of an exsiccatum that information and material content be the same for all duplicates, this information may be copied to all such derived units.

After 1952, publication of a new botanical name on exsiccatum labels is not effective for the purposes of botanical nomenclature, unless the separately printed text is distributed independently of the exsiccata proper. A separate publication reference is supplied for this case ("Exsiccata booklet reference"). Another reference may be added to provide additional notes on the exsiccata.

Chemical identification. – This was treated as a subtype of DERIVED UNIT in the original CDEFD model. It exemplifies a set of attributes that some may consider to be descriptors gained in a study. However, in a natural substance collection the determination of the structure of a compound is a routine procedure and may even determine the physical arrangement under which the collection units are stored. (Note that processes like extraction, purification, and isolation belong to the derived unit creation event and derived unit creation method.) Analogous "identification subtypes" may be formulated for geological substrate in palaeontological collections, or the soil samples used in the isolation of microbes, etc., always provided that in the collection where these units are housed such properties are considered an integral part of the unit data. All these are now treated as identification events (see Section 7).

6. *Collection management*

Collection management is here understood to include accession coding, location, and storage (Fig. 10); the management of person-related data (Fig. 11); various tasks

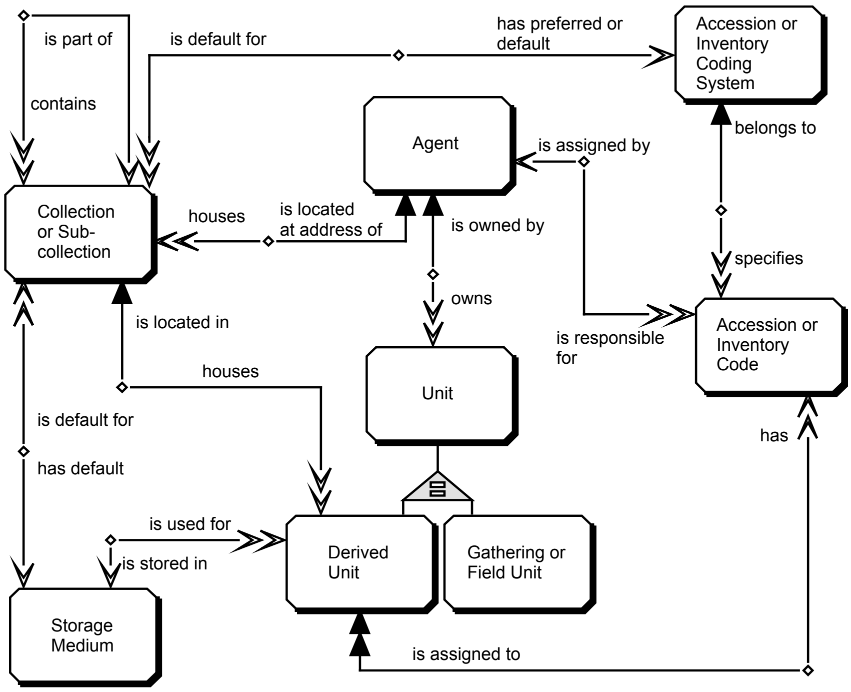

Fig. 10. Unit accession, location and ownership. Attributes for AGENT: Table 14; ACCESSION: Table 12; COLLECTION OR SUBCOLLECTION and STORAGE MEDIUM: Table 13; DERIVED UNIT: Table 9, UNIT: Table 6.

Table 12. Unit accession attributes (description, data type, short name; Section 2).

Attributes of entity type ACCESSION OR INVENTORY CODE		
Accession or inventory code key	int	AccCode_Pk
Accession or inventory code or number	str	AccCode
Coding system key	int	AccSyst_Fk
Person(s) responsible for code assignment (agent key)	int	AccRsp_Fk
Accession date	date	AccDate
Attributes of entity type ACCESSION OR INVENTORY CODING SYSTEM		
Coding system key	int	AccS_Pk
Coding system name	str	AccSNam
Coding system abbreviation	str	AccSAbbr
Coding system description	text	AccSTxt
Coding system internal use only flag	bool	Internal
Coding system unique codes used flag	bool	Unique
Coding system bulk accession possible flag	bool	Bulk
Coding system digits allowed flag	bool	DigitsOK
Coding system alphabet characters allowed flag	bool	CharOK
Coding system upper case only flag	bool	UCaseOnly
Coding system list of other allowed characters	str	OtherChars
Machine readable coding system type (e.g.: responder, bar code 3 of 9)	str	MRCType

connected to transfer management (Fig. 12); as well as preservation treatments (Fig. 13). An important feature of the management model is that it is not limited to the view of the owner of a collection. The model provides a symmetrical view of transactions between collections. It can therefore serve to design a truly distributed collection information system, which can be used efficiently to support the exchange of specimens and the sharing of information records between a large number of collections and institutions.

6.1. Accession or inventory codes

Accession or inventory codes apply to derived units and are thus issued by a particular holder of a collection (e.g. a herbarium). Loan identifiers assigned by lending or receiving institutes are not included (see Section 6.4), but seed list numbers (important in botanical gardens) are. A single unit may receive several accession or inventory codes from different code systems of the same institution, and in the case of bulk accessions a single code may apply to many units. A default or standard accession system can be set for a specific collection or subcollection. Table 12 details the attributes of the two entities involved.

The attribute “Accession or inventory code or number” is a textual expression of the entire code. Additional attributes may be introduced to conform to specific institutional coding schemes. The date and the person responsible for the coding may be specified. The accessioning or inventory system has a name (e.g.: live collection accession system at Berlin; Herbarium Willdenow accession at Berlin, U.S. National Herbarium bar-code), an abbreviation, and a description. Some implementation-oriented data items may be used to maintain data integrity: the “... unique codes used flag” indicates that an error is to be generated if entry of identical codes for different units is attempted; the range of permitted characters is defined by the flags “... digits allowed”, “... characters allowed”, and “... upper case only”, together with the “... list of other allowed characters" attribute (giving a list of punctuation marks or other characters that may be used in the code). The suitability of the system for referencing is expressed by the "... bulk accessions possible flag" (a single code may refer to several dissimilar units) and the "... internal use only flag" (indicating that the code should not be cited). If the code is machine-readable, the specific type is to be indicated (e.g. 'bar-code 3 of 9').

Table 13. Unit storage and storage location attributes (description, data type, short name; see Section 2).

Attributes of entity type COLLECTION OR SUBCOLLECTION		
Collection or subcollection key	int	Collection_Pk
Collection or subcollection name	str	CollNam
Collection or subcollection standard abbreviation	str	CollAbbr
Parent collection (collection or subcollection key)	int	CollPar_Fk
Collection located at (agent key)	int	CollLoc_Fk
Default accession system (coding system key)	int	AccSyst_Fk
Storage medium key	int	StorMed_Fk
Collection or subcollection stored under name entry required flag	bool	StorNmRequ
Attributes of entity type STORAGE MEDIUM		
Storage medium key	int	StorMed_Pk
Storage medium designation	str	StorMedNam
Storage medium description	text	StorMedTxt

6.2. Unit location and storage

The name of a collection or subcollection housing a specific unit will often correspond to the name of an agent (see 6.3: person, person team, company or organisation). However, a specific subcollection may have to be cited (e.g. 'type herbarium, Institut Scientifique de Rabat'). Such subcollection names are also instances of the entity type COLLECTION OR SUBCOLLECTION (Table 13), the hierarchical structure being expressed by a recursion.

The unit's location within the collection is specified in attributes of the entity type DERIVED UNIT. The "Storage location detail" is used for a specific storage site, e.g. the position of a sample in liquid nitrogen storage. Unit storage is often organised according to taxon name, in which case the attribute "Name under which unit is stored" should be used to form a link to TAXON NAME. The entry is made obligatory when a corresponding flag in the entity type COLLECTION OR SUBCOLLECTION is set, as will usually happen when units are often stored under a name that differs from the determined one, e.g. in the case of mixed collections forming associated units or of type material of synonyms.

The linked entity type STORAGE MEDIUM (see Fig. 10) includes attributes to describe containers, mounts or preservation fluids used to store the individual unit (Anonymous, 1993). As in the case of material categories, collection managers should carefully consider the standard media to be specified, because later separation of categories may involve the revision of the entire collection. Often the units of an entire collection or subcollection are using the same storage medium. In this case, a default value can be defined.

An ensemble is a set of units that are usually handled together, because they are physically united (e.g. several lichens on a piece of rock) or otherwise combined (e.g. a microscopic slide stored together with dried material in a herbarium capsule). The primary key of one of the units in the set is used as the "Tag for members of an ensemble set", which marks every member of the set (using the unit key ensures that each ensemble set is identified by a unique value). The position of a unit within an ensemble set, e.g. a separately accessioned herb on a herbarium sheet, is covered by the attribute "Additional curatorial identifier" in DERIVED UNIT.

6.3. Persons, teams, institutes, companies, and organisations

An entity type AGENT is needed to provide a common interface to person(s), institutes, organisations and companies, which may act as owners, recipients, sources, etc. of materials. The construct (Fig. 11, Table 14) resembles the supertype "Agent" specified in the ASC model (Anonymous, 1993). The ASC model's subtype "platform" (used there to designate e.g. vessel or automated observation station) has here been included with PROJECT OR EXPEDITION, because it primarily relates to gathering events. Individual persons and groups have been united in the entity type PERSON TEAM. Entities of the type AGENT, in the present model, have a defined (mailing) address; telecommunication contacts and email addresses are provided by the entity type TELECOMMUNICATION NUMBER OR ADDRESS (both entity types are identical with the corresponding ones in the CRIS model: Anonymous, 1994a; see below). However, these can and should be replaced with an address database application module. The entity type AGENT ROLE provides a list of functions a specific agent may fulfil, thus providing the possibility to create lists of collectors, carrier companies, taxonomic experts, authorised personnel, etc. for user selection in data input, for data retrieval and data security purposes.

Agents belong to one of two defined subtypes. The first, COMPANY OR ORGANISATION, may be further subtyped to include existing standard lists (e.g. Holmgren & al., 1990, for herbaria; Heywood & al., 1990, for botanical gardens). The second subtype, PERSON TEAM ("committee" in Pankhurst, 1991) is a construct which is also used in the context of collectors, authors, etc. in the IOPI model (Berendsohn, 1994; see Elankovan & al., 1996, for implementation details). It contains an attribute "Person group description", which allows the entry of teams for which the members are not known or unspecified (e.g. 'local shepherds'). In implementations, a additional (calculated) field can be used to store a concatenated string with the designation of the team, single person, or group, to ease access to this information. The entity type PERSON TEAM MEMBER links the team with the individual person and defines the position of the person in a sequence of names. The attribute structure of *Authors of plant names* (Brummitt & Powell, 1992) in its Kew database version may be used for the entity type PERSON, with the attributes "Person name suffix" and "Middle name" added. A person's individual address may differ from the address given for a team of which the person is a member, and from the institution to which the person is affiliated.

6.4. Unit transfers: acquisitions, loans, gifts, and sales

Transfer management comprises all tasks which involve moving a unit from or to a collection or subcollection. This includes loans, permanent loans, gifts (also "reciprocal gifts", i.e. exchanges), staff collections, purchase or sale of units, and the moving of units within a collection. The model is to support the tracking of units, issuing reminders, establishing statistics, etc., all to be implemented by the application program. The SI-NMNH transaction management model (CRIS: Anonymous, 1994a) was based on a yearlong review of legal context and best practices in museum collection management (Blum, pers. comm.). It provides a very detailed analysis of these items for any kind of collection and has already been implemented, e.g. at the Smithsonian Institution and at the Royal Botanic Gardens, Kew. Many CRIS entities and attributes have been integrated into the model here presented. However, two major differences exist. First, the present model closely integrates transfer management with collection unit management, because in many collections the department that handles transfers will be (one of) the principal data entry point(s) for the system. This refers both to primary data capture (e.g. when specimens are sent out on loan) and secondary changes (e.g. when specimens are returned from a specialist who provided new determinations, type assignments, etc.). Many of the attributes which the CRIS model defines only in the context of transfers (transactions) are covered in the context of derived units in the present model (e.g. permanent and temporary storage location, accession numbers and other inventoried item data, specimen counts, disposition of acquisition items, primary acquisitions, items found in a collection, item parts). Second, the new model provides a symmetrical, i.e. institution-independent solution. However, as one reviewer (Blum) posited, the transacting parties do have different perspectives on the same event, and an implementation of the system as here modelled pre-supposes the use of essentially the same system by all parties involved. At any rate we recommend study of the CRIS model for implementation of unit transfers. We decided to avoid the term transaction because of its use in computer science.

The TRANSFER EVENT (see Fig. 12) includes the data of the actual despatch, transport and receipt of the material. It is analogous to the entity type "shipment" in

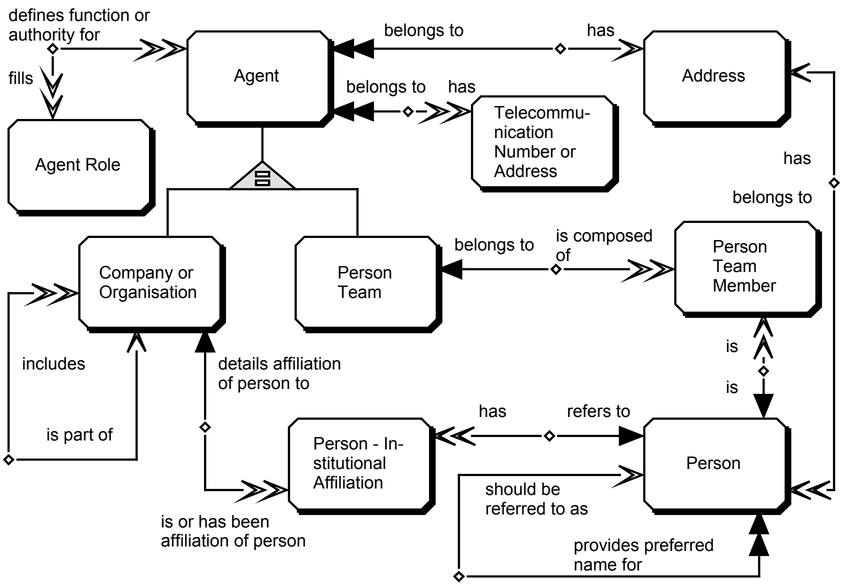

Fig. 11. Agents: person(s), institutes, companies, or organisations. Most attributes: Table 15.

Table 14. Agents attributes (description, data type, short name; see Section 2).

Attributes of entity type AGENT		
Agent key	int	Agnt_Pk
Agent type (classification variable, designating subtype of agent)	str	AgntTyp
Address key	int	AgntAdr_Fk
Agent inactive flag	bool	InactFlag
Attributes of entity type AGENT ROLE		
Agent role key	int	AgRole_Pk
Agent role description	text	AgRoleTxt
Attributes of entity type COMPANY OR ORGANISATION		
Company or organisation key (agent key)	int	CO_Pk
Company or organisation type (e.g.: museum transfer unit)	ext	COType
Company or organisation name	str	CONm
Parent company or organisation (agent key)	int	ParCO_Fk
Company or organisation standard abbreviation	str	COStnd
Source reference for standard abbreviation (reference title key)	int	StndSrc_Fk
Company or organisation comments	text	COTxt
Attributes of entity type PERSON		
Person name key	int	PersNm_Pk
Preferred name for person (person name key)	int	PersNm_Fk
Person title	str	Title
Person first names and prefix	str	ForePrfx
Person middle name	str	MiddleNam
Person initials and prefix	str	InitPrfx
Person last name	str	LastNm
Person name suffix (e.g.: jr., II)	str	NamSffx
Person inactive flag	bool	PInactFlag
Attributes of entity type PERSON TEAM		
Person team key (agent key)	int	PersTm_Pk
Person group description	str	PersTmTxt
Attributes of entity type PERSON TEAM MEMBER		
Person Team Key (agent key)	int	PersTm_Fk
Person name key	int	Pers_Fk
Position of name in team citation	int	Seniority
Attributes of entity type PERSON, INSTITUTIONAL AFFILIATION		
Person name key	int	PersNm_Fk
Organisation the person belongs to (agent key)	int	Inst_Fk
Person, institutional affiliation type (e.g.: employed by)	str	AffilType
Person, institutional affiliation job title	text	AffilTitle
Person, institutional affiliation time period	vdate	AffilDate
Person, institutional affiliation notes	text	AffilTxt

the CRIS model. A single transfer event may handle a mixture of sent and returned or partially returned loans as well as permanently transferred materials. Based on responsibility for the transfer steps, three subtypes may be defined: SENDING EVENT, CARRYING EVENT, and RECEIPT EVENT (not depicted in the diagram and tables). The attributes of the carrying event will not normally be handled by the carrier's own system. The many-to-many relationship of the TRANSFER EVENT with the UNIT entity type is resolved by means of the entity type UNIT TRANSFER.

Charges for sending, purchase, insurance, handling, etc. may accompany the event, or even a single unit transfer. The value may be fixed for the unit, the container, or for the entire shipment, and may differ from the insured value. Sources for the funding of these costs as well as payment methods may be specified. Such data should be integrated into an institutional accounting system module, which is not further treated in this model (CRIS specifies several attributes for commonly used data).

The gift, exchange, purchase, sale, and permanent loan of units are permanent transfers that change the physical custody and location of a unit . With the exception of loans, the title to the unit is also transferred. Every permanent transfer of a unit from one collection to another triggers a derived unit creation event. Once a unit has been stored in the system, it should remain there. A move within a collection may be considered a permanent transfer or only a change of storage (in the latter case, ownership, accession number, and subcollection assignation are to remain unchanged). A unit that has been lost, given away, or destroyed is marked accordingly (attribute "Derived unit presence"; see under unit availability in Section 5.2) but retains its link to the last storage location. The new derived unit is assigned to the new owner (and may receive an accession or inventory code, etc.). The data involved in transfers of units are specified as follows.

The entity type TRANSFER EVENT can be envisaged as a shipment containing one to several units in one or more containers. It is transferred by a "Carrier" from a "Sender" to a "Recipient", all three of them agents (see above), and as such, providing a link to address information (but the attribute "Deliver to" also allows to specify an address independent of the recipient agent address). The carrier specifies a transport agency, a broker, or a private person who is assigned temporary responsibility during the transport. The parcel is sent and received at given dates. Free text notes may be made upon despatch and/or receipt, and the recipient may acknowledge receipt. A number of additional attributes have been defined by the CRIS model and are included in Table 15. The application software is to ensure that in a transfer event the data for ensemble sets are kept together.

The entity type UNIT TRANSFER describes the context of the event. We distinguish the following "Unit transfer types": 'staff collection', 'loan sent', 'loan received', 'loan return sent', 'loan return received', 'gift or exchange sent', 'gift or exchange received', 'purchase received', 'sale sent', 'permanent loan sent', and 'permanent loan received'.

UNIT TRANSFER can be linked to one or more entities of the type PERMIT, which may be required for the transfer. The "Permit type" describes standard permits (e.g. those required by CITES, or export permits and sanitary certificates needed in international transfers). The "Permit number or identifier" and the "Permit description" may be used to further specify a permit, which is given by an agent representing the "Permit issuing authority". In an implementation, these attributes may take the form of a link to a correspondence filing system. A unit transfer can also be linked to an entity of the type TRANSFER AGREEMENT, e.g. a sales contract or exchange agreement, documents which mostly exist at least in the case of permanent transfers. permits are issued by government or other legal authorities (NGO, landowners) and may apply to interactions between several agents (e.g. a collector and several recipient institutions). In contrast, transfer agreements are bilateral (attributes "Party1" and "Party2"). The CRIS model (Anonymous, 1994a) distinguishes the "Transfer

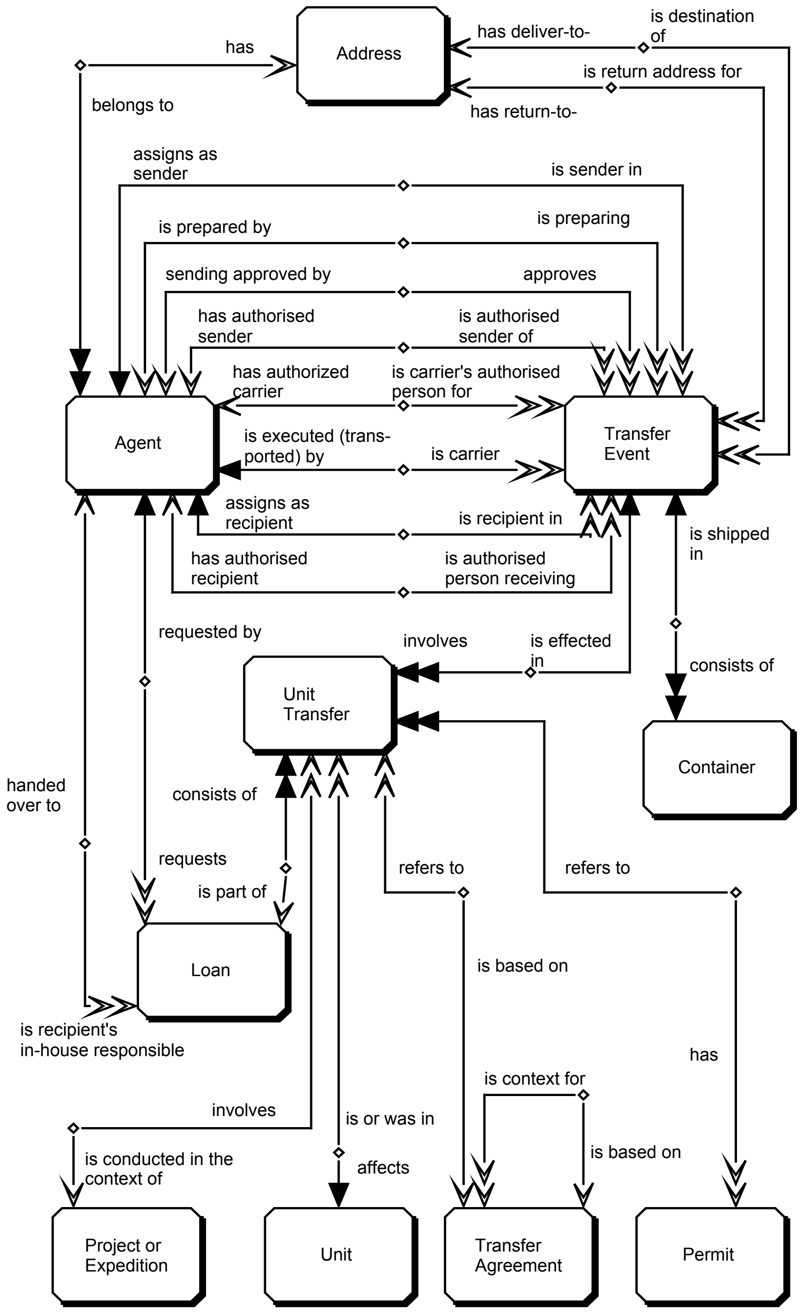

Fig. 12. Unit management. Relationships between AGENT and TRANSFER AGREEMENT as well as between AGENT and PERMIT are not depicted (see text and tables). Attributes for AGENT: Table 14; PROJECT OR EXPEDITION: Table 1; UNIT: Table 6; all other entity types: Table 15. Address deferred to subsystem (see Section 2).

Table 15. Unit transfer and loans attributes (description, data type, short name; see Section 2).

Attributes of entity type TRANSFER EVENT		
Transfer event key	int	TrEvt_Pk
Shipment status	str	ShpmntStat
Sender (agent key)	int	Sender_Fk
Deliver to (address key)	int	Delivr2_Fk
Return address provided by sender (address key)	int	Return2_Fk
Sending prepared by (agent key)	int	SndPrep_Fk
Sending approved by (agent key)	int	SndAppr_Fk
Authorised person sending the shipment (agent key)	int	AutSndr_Fk
Date of sending	date	SentDate
Notes on sending	text	SendNote
Hazardous material flag	bool	HazMatFlag
Hazardous material description	text	HazMatTxt
No. of individual containers	int	ContCount
Sender instructions for carrier	text	InstrucTxt
Hand carried flag	bool	HCarryFlag
Carrier carrying out transfer (agent key)	int	Carrier_Fk
Carrier shipment ID,. "Accountable shipment ID" or "waybill number" in CRIS	str	ShpmntID
Carrier shipment description	str	ShpmntTxt
Carrier arrival at destination date	date	CarrierArr
Authorised person in the carrier's office (agent key)	int	AutCarr_Fk
Recipient (agent key)	int	Rcp_Fk
Date received	date	RcpDate
Authorised person receiving the shipment (agent key)	int	AutRcp_Fk
Received damaged flag	bool	Damaged
Notes upon receiving	text	RcpNote
Temporal storage location	str	TempStor
Receipt acknowledged by recipient date	date	RcpAckDate
Attributes of entity type UNIT TRANSFER		
Unit transfer key	int	UnitTr_Pk
Unit transferred (unit key)	int	Unit_Fk
Transfer event key	int	TrEvt_Fk
Unit transfer type	str	UnitTrType
Transfer agreement key	int	TAgrmnt_Fk
Loan key	int	Loan_Fk
Unit description for transfer purposes	str	UnitTxt
Unit specific instructions	str	UnitInstr
Unit further comments	text	UnitCommts
Unit transfer priority flag (indicating that a transfer should be processed quickly)	bool	PriorFlag
Unit transfer insurance required flag	bool	InsureFlag
Attributes of entity type PERMIT		
Permit key	int	Pmt_Pk
Permit holder (agent key)	int	PmtHold_Fk
Permit type (e.g.: export permit, CITES, collection permit)	str	PmtType
Permit-issuing authority (agent key)	int	PmtAuth_Fk
Permit number or identifier	str	PmtNo
Permit description	text	PmtTxt
Permit start date	date	StartDate
Permit end date	date	EndDate
Last renewal date of permit	date	RenewDate
Contact for permit in holder's organisation (agent key)	int	Contct_Fk

Table 15 (continued).

Attributes of entity type TRANSFER AGREEMENT		
Transfer agreement key	int	TAgrmnt_Pk
Parent agreement (transfer agreement key)	int	ParAgrm_Fk
First party in agreement (agent key)	int	Party1_Fk
Second party in agreement (agent key)	int	Party2_Fk
Contact person for first party (agent key)	int	Contct1_Fk
Contact person for second party (agent key)	int	Contct2_Fk
Transfer agreement type	str	TAgrmntTyp
Transfer agreement name	str	TAgrmntNam
Permanent transfer agreement text	text	TAgrmntTxt
Transfer agreement period	vdate	StartDate
Transfer agreement automatic prolongation period	str	AutProlng
Initial balance	int	InitBal
Attributes of entity type LOAN		
Loan key	int	Loan_Pk
Requested by (agent key)	int	Request_Fk
Loan initiated by sender flag	bool	SInitFlag
Purpose of loan (e.g.: exhibit, study, identification)	str	Purpose
Loan sender loan identifier code	str	SndrLoanID
Loan recipient loan identifier code	str	RecpLoanID
Responsible (person team key)	int	Rsp_Fk
Loan deadline for return date	date	DeadlnDate
Loan extended until	date	ExtendTill
Loan reminder letters sent or received	text	RemSent
Loan completely returned flag	bool	AllBack

agreement type" 'open exchange agreement' from 'collecting agreement', 'contract', and 'restriction'. The "Transfer agreement name" enables users to select an agreement from a list. The "Parent agreement key" provides a reference for a more inclusive transfer agreement. As for permits, validity (start and end) dates can be specified, and a contact person may be designated by each party. Several additional attributes may be defined, e.g. a flag indicating that types have to be returned to the country of origin and an attribute specifying the number of duplicates that have to be returned (for collection agreements), or a flag indicating that the sender is to be alerted to identification events involving the units transferred. Usually exchange programmes are not related to single units but an agreement exists to exchange a certain number of units within a time period. Contracts and collecting agreements may also specify the number of units to be delivered. The "Initial balance" attribute provides the number of specimens sent by each party before the system based on this model was implemented. The model itself provides the possibility to calculate the number of items sent and received that are related to a specific agreement. No additional data elements are necessary to account for the balance. Attribution of unit transfers to a project may be used to generate reports of the transaction activity related to it.

A loan needs a number of specific attributes. Institutes generally separate the management of loans received from that of loans sent. However, the data strongly overlap and, true to the present attempt to create a symmetrical model of transfer management, they have not been separated. The loan has usually been "Requested by" a person or person team and on the receiving side there is a "Responsible" person or team. If the sender has initiated the loan, the respective flag is set. The "Loan purpose" may be specified (e.g. 'acquisition consideration', 'exhibit', 'identification', 'study', 'temporary custody'). Usually, a "Loan deadline for return date" is set, and in case that the recipient did not meet the deadline, "Loan reminder letters" may be sent or received. The latter free text attribute allows entering references to several reminders, specifying the circumstances and dates. The deadline may have been extended ("Loan extended until"). Sender as well as recipient may assign a loan identifier (number or code). Because of the frequent occurrence of partial loan returns, the loan status needs to be calculated by the application. For each unit transfer with type 'loan sent' a corresponding unit transfer with type 'loan sent returned' must be present. The "Loan completely returned flag" is a calculated attribute included to simplify management of loan data.

Table 16. Preservation or health treatments attributes (description, data type, short name; see Section 2).

Attributes of entity type PRESERVATION OR HEALTH ASSESSMENT EVENT		
Preservation or health assessment event key	int	PHA_Pk
Assessed unit (unit key)	int	Unit_Fk
Preservation or health state	str	PHState
Preservation or health state assessment method	str	PHAMtd
Preservation or health state assessment event date	date	PHADate
Agent responsible for assessment (agent key)	int	PHAAgnt_Fk
Attributes of entity type PRESERVATION OR HEALTH TREATMENT EVENT		
Preservation or health treatment event key	int	PHT_Pk
Unit treated (unit key)	int	Unit_Fk
Preservation or health treatment description	str	PHTTxt
Preservation or health treatment date	date	PHTDate
Agent responsible for treatment (agent key)	int	PHTAgnt_Fk
Preservation or health treatment method key	int	PHTMtd_Pt
Attributes of entity type PRESERVATION OR HEALTH TREATMENT METHOD		
Preservation or health treatment method key	int	PHTMtd_Pk
Preservation treatment method name	str	PHTMtdNam
Preservation treatment method description	text	PHTMtdTxt

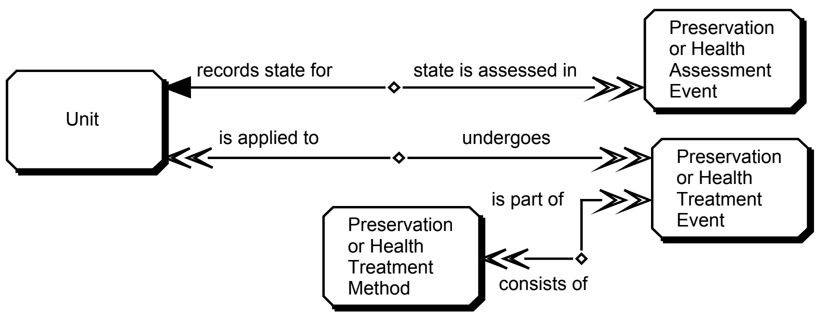

Fig. 13. Preservation or health assessment and treatments. Attributes: Table 16; Unit: Table 6.

The intra-institutional responsibility for certain groups of organisms may be important for tranfer management. This can be recorded where needed in an entity type linked to TAXON NAME. The link to UNIT is realised via the preferred determination or the attribute "Name under which unit is stored".

6.5. Preservation or health state and treatments

A preservation or health state assessment event (Fig. 13) for a unit may occur during the accessioning of new units, in the course of loan management (when a foreign loan is received, before a unit is sent on loan, and upon receipt of the return), or, especially in culture collections, as a regular maintenance measure. The attributes of the this entity type (Table 16) include the method used, the agent carrying out the assessment, the date of the event, and the result of the assessment (which may in turn be standardised).

Preservation or health treatments act on existing units presumably without actually changing them (as opposed to the derived unit creation methodology). However, knowledge about past treatments may be important if a unit is used in a new context (e.g. chemical analysis of herbarium specimens). Examples for treatments include transfer of fungal cultures on an antibiotic growth medium, treatment of an individual herbarium sheet with chemical substances to prevent or fight moulds or pests, or pesticide treatments in live collections. The treatment may apply to several units at once (e.g. all herbarium specimens in a fumigated room; preliminary liquid preservation of plants during expeditions). The attributes (see Table 16) include the date and method, and the agent applying the treatment. The treatment description may be used to specify parts of the unit treated, etc. If the treatment consists of several steps, a recipe structure may be added (a loop consisting of attributes giving a method and the sequence in time of its application within the treatment).

The concept of preservation or health treatments may be widened to describe other significant measures taken in the maintenance of live units (e.g. pruning). However, their demarcation against derived unit creation events must be well defined.

7. Identifications

In the original CDEFD model (Berendsohn & al., 1996) identifications were restricted to taxonomic identifications, which still form the probably most important item modelled. However, it was decided to provide a subtype structure (Fig. 14) which would make the model more flexible and which would generate less duplication of structures. Examples are chemical substance identification events, which the CDEFD model treated as a DERIVED UNIT subtype, and nomenclatural types, which were handled by a completely separate structure. All identifications have in common that an event takes place by which a unit is assigned to a specific category according to some kind of classification system or fixed convention. In the case of both taxonomic and chemical identifications, the event links the unit to a class in a systematic classification system. Another example would be the identification of clearly defined larval stages and/or of the sex of specimens, which might be defined as another identification subtype. The unconstrained identification events do not differ from the other subtypes except by the fact that the classification system is not accessed by the system. Nomenclatural types are well defined units within the conventions of nomenclatural Codes, so that the identification scheme can be applied to them.

Identifications refer to the supertype UNIT, because they can usually be made on objects in the field as well as objects in a collection. The supertype UNIT IDENTIFICATION EVENT (Table 17) includes the who and when of the event, provision for free text notes, and a link to a literature item used as a source.

7.1. *Taxonomic identifications*

The result of taxonomic identification is a potential taxon name (Berendsohn, 1995) or, where potential taxa are not implemented, a taxon name, preferably of standard form (Bisby, 1995).

The attribute "Taxon identification type" allows to distinguish the cases detailed in Table 18. This list is presumably complete for taxonomic identifications, and the identification type can therefore be handled by a controlled vocabulary. Nevertheless, more thought is needed to decide whether a separate entity type linked to the

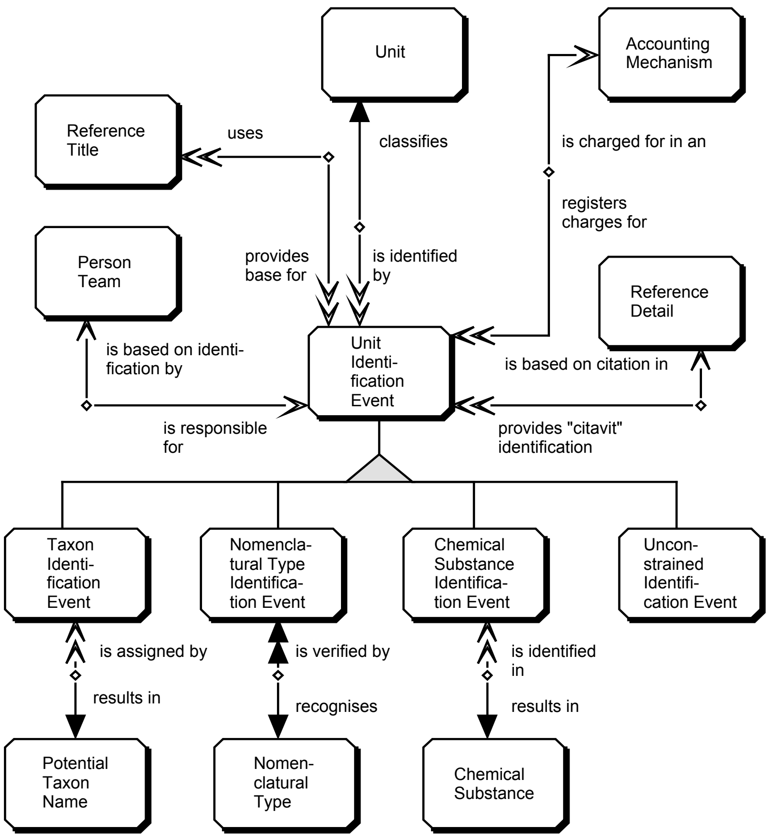

Fig. 14. Unit identification events. Attributes for ACCOUNTING, REFERENCE TITLE and REFERENCE DETAIL deferred to subsystems (see Section 2); UNIT: Table 6; PERSON TEAM: Table 14; NOMENCLATURAL TYPE IDENTIFICATION EVENT: Table 19; CHEMICAL SUBSTANCE and CHEMICAL SUBSTANCE IDENTIFICATION EVENT not treated in detail; all other entity types: Table 17.

Table 17. Identification events attributes (description, data type, short name; see Section 2).

Attributes of entity type UNIT IDENTIFICATION EVENT		
Identification event key	int	IdEvt_Pk
Person(s) who identified (person team key)	int	Identf_Fk
Unit identified (unit key)	int	Unit_Fk
Identification event date	date	Ivdate
Identification event notes	text	IdNote
Identification found in literature (reference detail key)	int	IdFrom_Fk
Attributes of entity type UNCONSTRAINED IDENTIFICATION EVENT		
Free description for non-categorised unit material	str	MtDescr
Attributes of entity type TAXON IDENTIFICATION EVENT		
Taxon identification type	str	TIType
Confirmed or preferred identification (identification event key)	int	Conf_Fk
Verification level for identification (ITF-1 and -2 standard values; see Wyse Jackson, 1997)	str	TIVerifLvl
Taxon identification qualifier (expresses insecurity in positive identifications)	str	TIQual
Rank of name element the qualifier refers to (may be either the monomial, or the epithet of a binomial, or the last epithet of a trinomial)	int	TIQualRnk
Identification reference (reference title key)	int	TIRef_Fk
Original taxon name string from identification	str	TaxonName
Assigned taxon name (the "life link" to the taxonomic reference subsystem: potential taxon name object identifier).	int	Taxon_Fk

supertype might provide a more general but still practical solution. The different identification types are exclusive, i.e. every event can only be of one type. Entities of the type TAXONOMIC IDENTIFICATION EVENT thus constitute elements of the determination history of a unit.

The question of the relationship between taxonomic reference system and identification results must briefly be considered. Obviously, taxon names and their status may change over time. If identification events are directly linked to the taxonomic reference system, such changes directly effect identifications. This may be desirable for data retrieval, but care is to be taken to preserve the original data. In the present model, two solutions are given to this problem. First, the name that was given by the determiner can be stored as a string value with the record of the determination event, thus effectively isolating it from the taxonomic reference system. Second, any change of name is triggering a new identification event of the type "Rename", which refers back to the original determination on which it was based. So, the system can assign a correct name to a synonym (with all due caution!), but the credit for the identification remains with the original determiner. Such redeterminations are often cited by adding one of the terms "as", "under", "pro", or "sub", followed by the taxon name as by the original taxon name identification.

In new identifications or confirmations, the certitude of the identification may be expressed by a "Taxon Identification qualifier". Qualifiers include 'cf. ...' or '? ...' with the meaning of "to be compared to" or "could be", and 'aff. ...', 'near ...', or 'c. ...' with the meaning of "is not identical with but related to". The "Rank of name element the qualifier refers to" is to be specified. For example, the modifier 'cf.' for an identification may refer to genus ("cf. *Inula conyza*"), or species level *("Ficus cf.*

trigonata”). An additional qualification of the degree of confidence that can be placed on the identification is the “Verification level” (ITF standard: Anonymous, 1987b), which is used to express the degree of authority attributed to the determiner.

If inclusion in or comparison with more than one taxon is intended, e.g. in multiple qualified identifications (“... vel aff.” or “... vel cf. ...”), or if combined positive and negative identifications are meant (“... non ...”), these have to be treated as separate identification events, each carrying the respective modifier and type. Concatenation of such results into a single string is to be handled by the output system (see Meyer & al., 1996) pulling together all events with identical date and determiner.

All units that are linked by derived unit creation events specifying a ‘similar’ inheritance type form a “similar unit set”, which is identified by an attribute in DERIVED UNIT CREATION EVENT. For a specific unit, all identifications referring to a unit in the same similar unit set should be retrieved (see Meyer & al., 1996). If a unit was involved in more than one identification event, the most recent identification normally takes precedence. Where the date of an original identification is not known, one must make sure that it is indeed prior to all other, dated ones.

Table 18. The different values of taxonomic identifications events (attribute TIType of Table 17). Exam. = identification implies object examination.

Value	*Description*	*Exam.*
‘Det.’	The examination of the material results in a new identification or correction	Yes
‘Conf.’	The examination confirms a previous identification, specified by the attribute Conf_Fk	Yes
‘Pref.’	Assigning preference to an older determination specified by the attribute Conf_Fk over the most recent one. Differs from a confirmation only in that the object has not been re-examined. The choice is based on the expertise of the person effecting the earlier identification	No
‘Non’	In a negative identification, the material is identified as not belonging to the identified taxon. The implementation must enforce simultaneous creation of a positive or confirming identification event (which may determine the object only to the level of genus or higher, if no better identification is currently possible)	Yes
‘Dub.’	Same as ‘Non’, but without examination of the material – e.g. an amateur observation of a taxon completely out of its range (according to the determiner = creator of this event)	No
‘Abs.’	An identification event may also express the absence of a specific taxon from a searched site. This primarily refers to field observations, when a certain site has been searched unsuccessfully for an organism thought to exist there (valuable information in floristic databases). A dummy gathering or field unit is created to link the identification event with the site and gathering event. The search effort (e.g. cursory search, or searched for several hours) can be detailed in “Identification event notes”. Absence of a taxon can also apply to collection units, e.g. in microfungi when nomenclatural type material has been used up in previous studies, or if a mixed collection contains several fungi but none fits the protologue	Yes
‘Ren.’	Renaming (e.g. assigning a species to a new genus, or a purely nomenclatural change) of a unit by somebody who has not re-examined the material is a novel event that cannot be attributed directly to the original determiner, who, however, remains the authority for the determination (expressed by the link in Conf_Fk)	No
‘Corr.’	Correction of a misidentification supposedly due to a typing error, e.g. the northernmost Finnish record of *Calla palustris,* published in a relevé, probably refers *Caltha palustris* (Lampinen, pers. comm.). This is a new identification event	No
‘Implicit’	The identification can be logically deduced from the material; e.g. in the case of type specimens	No/Yes

Identification references (keys, floras, etc.) should be specified by using the link to REFERENCE TITLE. In the case of a system that refers only to taxon names, citing the major reference relied upon for determination opens a back-door to later reconstruction of a potential taxon name: The identification reference then takes the place of the "sec." citation (circumscription reference) for the potential taxon.

In culture collections, identifications often carry a charge, especially if requested for commercial purposes. In natural history collections like herbaria, charging for identification is still rather unusual.

7.2. Unconstrained identification events

The entity type UNCONSTRAINED IDENTIFICATION EVENT (Table 17) was created to accommodate closer descriptions or identifications of samples that are too specific to form a material category of their own, or whose taxonomic identification is foreign to the scope of the collection. For example, it would be inappropriate to force a mycologist to enter an insect taxonomy system for an insect from which a fungus has been isolated, when entomological data are not normally used in this context. In soil samples, 'soil' might be a material category; however, a more detailed description will usually be given (e.g. 'sandy soil under 70 year old pine trees', 'arable soil from a wheat field'). Material from which microbes are isolated can be quite exotic, e.g. 'glass of optical lenses' or 'aluminium alloy of aeroplane kerosene plumbing'. Introducing a material category of its own for each such case would over-structure the database. Therefore the material category should remain general and the attribute "Free description ..." can be used to further specify the material.

7.3. Nomenclatural types

Storage and management of nomenclatural type specimens is the most noble responsibility of natural history collections. The entity types NOMENCLATURAL TYPE IDENTIFICATION EVENT and NOMENCLATURAL TYPE (see Fig. 15 and Table 19) are used to record the additional information that is needed for type specimens. Types present in foreign collections may be included, and photographs etc. representing them in the home collection may be registered. The structure also accommodates illustrations that serve as nomenclatural types.

The model accommodates all attributes resulting from discussions in the TDWG Type Databases Working Group (Farr, unpubl.), such as attributes or links to other entities, to which several were added. The attributes cover the typified name (as a link to TAXON NAME), the bibliographic reference to the protologue, in some cases to an illustration that serves as type, and for lectotypes and neotypes, to the bibliographic item where these sere designated (see Fig. 15).

The NOMENCLATURAL TYPE IDENTIFICATION EVENT handles information created by later curatorial work, i.e. the verification of the presence, identity, and status of a type specimen in a collection. One attribute of the NOMENCLATURAL TYPE IDENTIFICATION EVENT is the "Nomenclatural type status", which defines the kind of type entered (holotype, isolectotype, etc.; see Croft, 1992). The link to the identified unit provides all type-data related to the gathering, as well as to storage location (including 'stored under name ...'), accession number, material category (e.g. illustration), and current determination. Copies of the protologue or of published illustrations, photographs of type material, etc., when stored as units in a collection, can be linked by means of a derived unit creation event to the original type unit.

Table 19. Nomenclatural type attributes (description, data type, short name; see Section 2).

Attributes of entity type NOMENCLATURAL TYPE IDENTIFICATION EVENT		
Nomenclatural type key	int	Type_Fk
Nomenclatural type status	str	TypeStatus
Original type unit of illustration or fragment (unit key)	int	OType_Fk
Attributes of entity type NOMENCLATURAL TYPE		
Nomenclatural type key	int	Type_Pk
Typified name (taxon name key)	int	TaxNam_Fk
Protologue citation (reference detail key)	int	PtlRefD_Fk
Illustration reference (reference detail key)	int	IllRefD_Fk
Lectotypification reference (reference detail key)	int	LTypeRf_Fk

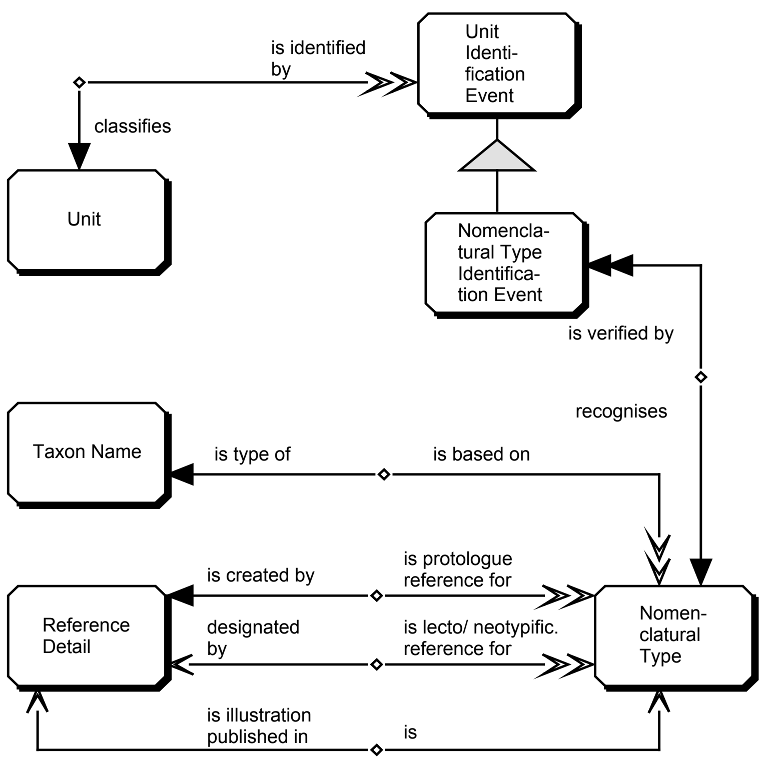

Fig. 15. Nomenclatural types. Attributes for REFERENCE DETAIL and TAXON NAME deferred to subsystems (see Section 2); UNIT: Table 6; UNIT IDENTIFICATION EVENT: Table 17; NOMENCLATURAL TYPE AND IDENTIFICATION EVENT: Table 19.

8. Conclusions

Knowledge about the data needed in collection management systems has improved significantly over the last decade. Notable congruence exists between recently developed databases and data models. The CDEFD model was discussed by various groups of researchers, collection managers, and IT professionals from various systematic fields, and it was used to evaluate implementation designs in workshops and reviews by the first author. It is obviously possible and useful to formulate a common backbone data structure covering diverse biological fields, live and preserved specimens, collection and survey data. The backbone data structure originally proposed by CDEFD and further developed by the present group provides adequate general coverage of such data. Descriptive data must needs be excluded from such a general standard, because they largely depend on the collection purpose and group of organisms. Existing geographical data standards and standard data should be used whenever possible, and an overall consensus on authority files or systems is to be achieved within the biological discipline. Funding bodies should encourage researchers to document and reference the databases used to publish their research.

Acknowledgements

CDEFD and BioCISE are Concerted Action Projects funded by the European Commission, DG XII, Biotechnology. We express our acknowledgement to E. Magnien, K. Beese, A. Hoeveler and C. Martínez, responsible for the projects within the Biotechnology Programme. We are indebted to Frank Bisby, driving force behind the initiation of the taxonomic section of that programme, and to Werner Greuter, who triggered our application for funding. Administrative and technical support for the CDEFD project was provided by C. Fritz and I. Mischke, for BioCISE by S. Steinmann (project secretaries) as well as S. Elankovan, T. and K. HradeÅny, H. Meyer, X. Villavicencio, and B. Felinks (data verification and publication). The following have contributed by participating in meetings: G. Bedini, Pisa; F. Bisby, Reading; J. Cooper, Egham; E. Feoli, Trieste; P. Ganis, Trieste; D. Green, Albury; J. Greilhuber, Vienna; H. Haeupler, Bochum; H. Hovenkamp, Leiden; H. Kremers, Berlin; J. Lebbe, Paris; W. Loader, Kew; R. May, Bonn; D. Minter, Egham; C. Oberprieler, Berlin; E. Pastor, Sevilla; D. Phitos, Patras; G. Rambold, Munich; J. le Renard, Paris; A. Rissone, London; J. Rubio Recio, Sevilla; K. Siems, Berlin; and M. Tretiach, Trieste. Special thanks go to Stan Blum, Raino Lampinen, and Tapani Lahti for their most thorough reviews of the present paper.

Literature cited

Anonymous [FAO] 1974. *Map of the world,* 1. Paris.
– [USDA] 1975. *Soil taxonomy*. [*USDA SCS Agric. Handb.,* 436]. Washington, DC.
– [ISO] 1983. ISO 6709:1983 *Standard representation of latitude, longitude and altitude for geographic point locations.* Falls Church.
– [CEC] 1987a. *Natural vegetation map of Europe.* Bruxelles.
– [IUCN/WWF] 1987b. *The international transfer format (ITF) for botanic garden plant records.* [*Pl. Taxon. Database Standards,* 1]. Pittsburgh, PA.
– [ISO] 1988. *ISO 639. Codes for the representation of names of languages.* Wien.
*– [ASC] 1993. *An information model for biological collections (draft). Report of the Biological Collections Data Standards Workshop, August 18-24, 1992.* [gopher://biodiversity.bio.uno.edu:70/11/standards/asc].
*– 1994a [5 May]. *Logical data model for museum collections transaction management,* Version 1.0. [http://nmnhwww.si.edu/gopher-menus/LogicalDataModelforMuseumCollectionsTransactionManagement.html].

*– [FGDC] 1994b [8 Jun]. *Content standards for digital geospatial metadata, Federal Geographic Data Committee.* [http://geology.usgs.gov/tools/metadata/standard/metadata.html].
*– [NIMA] 1996 [26 Jan]. *Vertical datums, elevations, and heights.* [http://164.214.2.59/publications/guides/VerticalDatums/].
– [ISO] 1997a. *ISO 3166-1. Codes for the representation of names of countries and their subdivisions – part 1: country codes*. Berlin.
– [NIMA] 1997b. *Department of Defense world geodetic system 1984: its definition and relationships with local geodetic systems,* ed. 3. [NIMA TR8350.2]. Bethesda.
– 1998a. *Museum world gears up to the 'big bang' of informatics*. *Nature* 394:118-119.
*– [FGDC] 1998b [before Oct]. *Biological data profile of the content standard for digital geospatial metadata.* [http://www.fgdc.gov/standards/documents/standards/biodata/col-letr.html].
*– [FGDC] 1998c. *Content standards for digital geospatial metadata, Federal Geographic Data Committee,* version 2. [http://www.fgdc.gov/metadata/contstan.html].
*– [GII] 1998d. *Getty thesaurus of geographic names (TGN).* [http://shiva.pub.getty.edu/tgn_browser/].
– [ISO] 1998e. *Codes for the representation of names of languages – part 2: alpha-3 code.* [ISO 639-2]. Wien.
– [ISO] 1998f. *Codes for the representation of names of countries and their subdivisions – part 2: country subdivision code*. [ISO 3166-2]. Berlin.
*– [NIMA] 1998g [25 Sep]. *GEONet Names Server. National Imagery and Mapping Agency, a combat support agency of the US Department of Defense.* [http://www.nima.mil/gns/html/].
*– [USDA] 1998h. *Keys to soil taxonomy,* ed. 8. [http://www.statlab.iastate.edu/soils/keytax/].
*– 1999a [1 Jan]. *The Dublin core: a simple content description model for electronic resources*. [http://purl.oclc.org/dc/].
*– 1999b [14 Mai]. *FlyBase, a database of the* Drosophila *genome. Proforma controlled vocabularies, body parts.* [http://gin.ebi.ac.uk:7081/docs/lk/bodyparts-cv.txt].
– [ISO] 1999c. *Codes for the representation of names of countries and their subdivisions – part 3: code for formerly used names of countries.* [ISO 3166-3]. Berlin.
*– [NIMA] 1999d [17 Jun]. *NIMA geodesy and geophysics publications listing.* [http://164.214.2.59/GandG/pubs.html].
*– [OECD] 1999e [Jan]. *Final report of the OECD Megascience Forum Working Group on Biological Informatics.* [http://www.oecd.org/dsti/sti/s_t/ms/prod/birepfin.pdf].
*– [USDA] 1999f. *Revisions to the soil taxonomy handbook*. [http://www.statlab.iastate.edu/soils/soiltax/].
*Banta, J. 1999 [undated, v. 1 Jun]. *A brief examination of projections and datums.* [http://www.connect.net/jbanta/Datumbs.html].
Batini, C., Ceri, S. & Navathe, S. B. 1992. *Conceptual database design: an entity-relationship approach.* Redwood City.
Beach, J. H. 1998. Employing metadata for biological collections. *NFAIS Newslett.* 40: 117, 120.
Beard, J. S. 1946. *The natural vegetation of Trinidad.* Oxford.
Berendsohn, W. G. 1994. IOPI world vascular plant checklist. A CASE model of checklist system data, version 6.0. *In:* Wilson, K. (ed.), *Global Plant Checklist project plan,* version 1.2. Sidney. [See also *version 7.3. 23 Jun 1995; http://www.bgbm.fu-berlin.de/IOPI/iopimodel73/7301root.htm].
– 1995. The concept of "potential taxa" in databases. *Taxon* 44: 207-212.
– 1997. A taxonomic information model for botanical databases: the IOPI model. *Taxon* 46: 283-309.
*– 1999a [Apr]. Names, Taxa, and Information. *In:* Blum, S. (ed.), *Proceedings of the Taxonomic Authority Files Workshop, Washington, DC, June 22-23, 1998*. [http://research.calacademy.org/taf/proceedings/Berendsohn.html].
*– 1999b [3 Jan]. *Standards, information models, and data dictionaries for biological collections*. [http://www.bgbm.fu-berlin.de/TDWG/acc/Referenc.htm].

*– , Anagnostopoulos, G., Hagedorn, G., Jakupovic, J., Nimis, P. L. & Valdés, B. 1996 [5 Jul]. *The CDEFD information model for biological collections.* Berlin. [http://www.bgbm.fu-berlin.de/CDEFD/CollectionModel/cdefd.htm].

– , – , Jakupovic, J., Nimis, P. L. & Valdés, B. 1997b. A framework for biological information models. *Lagascalia* 19: 667-672.

– , Greilhuber, J., Anagnostopoulos, A., Bedini, G., Jakupovic, J., Nimis, P. L. & Valdés, B. 1997a. A comprehensive datamodel for karyological databases. *Plant Syst. Evol*. 205: 85-98.

*– , Felinks, B., Güntsch, A. & Hahn, A. 1999 [3 Jan]. *BioCISE – Resource identification for a biological collection information service in Europe.* Berlin. [http://www.bgbm.fu-berlin.de/biocise/].

Bisby, F. 1995. *Plant names in botanical databases.* [*Pl. Taxon. Database Standards,* 3]. Pittsburgh, PA.

– 1997. Putting names to things and keeping track: the Species 2000 programme for a coordinated catalogue of life. Pp. 59-68 *in:* Bridge, P., Jeffries, P., Morse, D. R. & Scott, P. R. (ed.), *Information technology, plant pathology and biodiversity.* Wallingford.

*Blum, S. D. 1996. *The MVZ collections information model. Conceptual and logical models.* [http://www.mip.berkeley.edu/mvz/cis.html].

Braun-Blanquet, J. & Fuller, G. 1929. *Plant sociology.* New York.

Brummitt, R. K. & Powell C. E. 1992. *Authors of plant names.* Kew.

*Bryan, M. & Li, M.-S. 1998 [Nov]. *Info 2000 OII standards and specifications list, geographic data exchange standards.* [http://www2.echo.lu/oii/en/gis.html].

Chen, P. P. 1976. The entity-relationship model – toward a unified view of data. *ACM Trans. Database Syst.* 1: 9-36.

Cook, E. M. 1995. *Economic botany data collection standard. Prepared for the International Working Group on Taxonomic Databases for Plant Sciences (TDWG).* Kew.

*Copp, C. J. T. 1998 [8 Apr]. *Recorder system analysis.* [http://www.nbn.org.uk/projects/recorder/].

*Croft, J. R. 1992. *HISPID – herbarium information standards and protocols for interchange of data. Summary paper and data dictionary.* [gopher://biodiversity.bio.uno.edu/00/standards/hispid/hispid91.std].

*Dana, P. H. 1998. *Global positioning system overview.* [http://www.utexas.edu/depts/grg/gcraft/notes/gps/gps.html].

*Daxinger, W. 1996 [16 Dec]. *Datumsfestlegung in Ingenieurnetzen.* [http://news.uibk.ac.at/c/c8/c802/vortrag/daxinger.html].

Duckworth, W. D., Genoways, H. H. & Rose, C. L. 1993. *Preserving natural science collections: chronicle of our environmental heritage.* Washington.

*Elankovan, S., Berendsohn, W. G. & Meyer, H. 1996 [15 Jul]. *Person teams in the CDEFD and IOPI models: an implementation.* [http://www.bgbm.fu-berlin.de/CDEFD/PersonTeams/title.htm].

Hagedorn, G. 2000. Making DELTA accessible: databasing descriptive information. *Bocconea* (in press).

Hawksworth, D. L., Kirk, P. M., Sutton, B. C. & Pegler, D. N. 1995. *Ainsworth & Bisby's dictionary of the fungi,* ed. 8. Egham.

Heywood, C. A., Heywood, V. H. & Wyse Jackson, P. 1990. *International directory of botanical gardens.* Königstein.

Holdridge, L. R. 1967. *Life zone ecology.* San José.

Hollis, S. & Brummitt, R. K. 1992. *World geographical scheme for recording plant distributions.* [*Pl. Taxon. Database Standards,* 2]. Pittsburgh, PA.

Holmgren, P. K., Holmgren, N. H. & Barnett, L. C. 1990. Index herbariorum. Part I. The herbaria of the world, ed. 8. *Regnum Veg.* 120.

*Meyer, H., Berendsohn, W. & Elankovan, S. 1996 [15 Jul]. *Taxonomic identification within the CDEFD information model.* [http://www.bgbm.fu-berlin.de/cdefd/identifications/title.htm].

Pankhurst, R. J. 1991. *Practical taxonomic computing.* Cambridge, U.K.

Soberón, J., Llorente, J. & Benítez, H. 1996. An international view of national biological surveys. *Ann. Missouri Bot. Gard.* 83: 562-573.

*Theodor, J. & Lindberg, D. 1996 [15 Jul]. *UCMP [University of California Museum of Paleontology] data model for paleontological collections*. [http://www.ucmp.Berkeley.Edu/museum/datamodel/datamodel.html].

Voser, S. 1999. Schritte für ein automatisiertes Koordinatenmanagement beim Datenaustausch in GIS und Kartographie. *Nachr. Karten- Vermessungswesen (NaKaVerm), Bundesamt Kartogr. Geodäsie* 1/118 (in press).

Walter, H. & Breckle, S.-W. 1983. *Ökologie der Erde,* 1. Stuttgart.

Watson, L. & Dallwitz, M. J. 1994. *The grass genera of the world,* ed. 2. Wallingford.

Weibel, S. 1998. The Dublin core: a simple content description model for electronic resources. *NFAIS Newslett.* 40: 117-119.

*Wilson, P. D. 1993. *Missouri Botanical Garden research data model. Release 4.0. Prepared for the Research and Steering Committee*. [gopher://cissus.mobot.org:70/11/pub].

*Wyse Jackson, D. (compiler) 1997 [May]. *International transfer format for botanical garden plant records. Version 2.00, draft 3.2.* [http://www.rbgkew.org.uk/BGCI/itf2-d31.doc].