

# **Specification of the database interface(s) of the MoReTax system**

*24 June 2002*

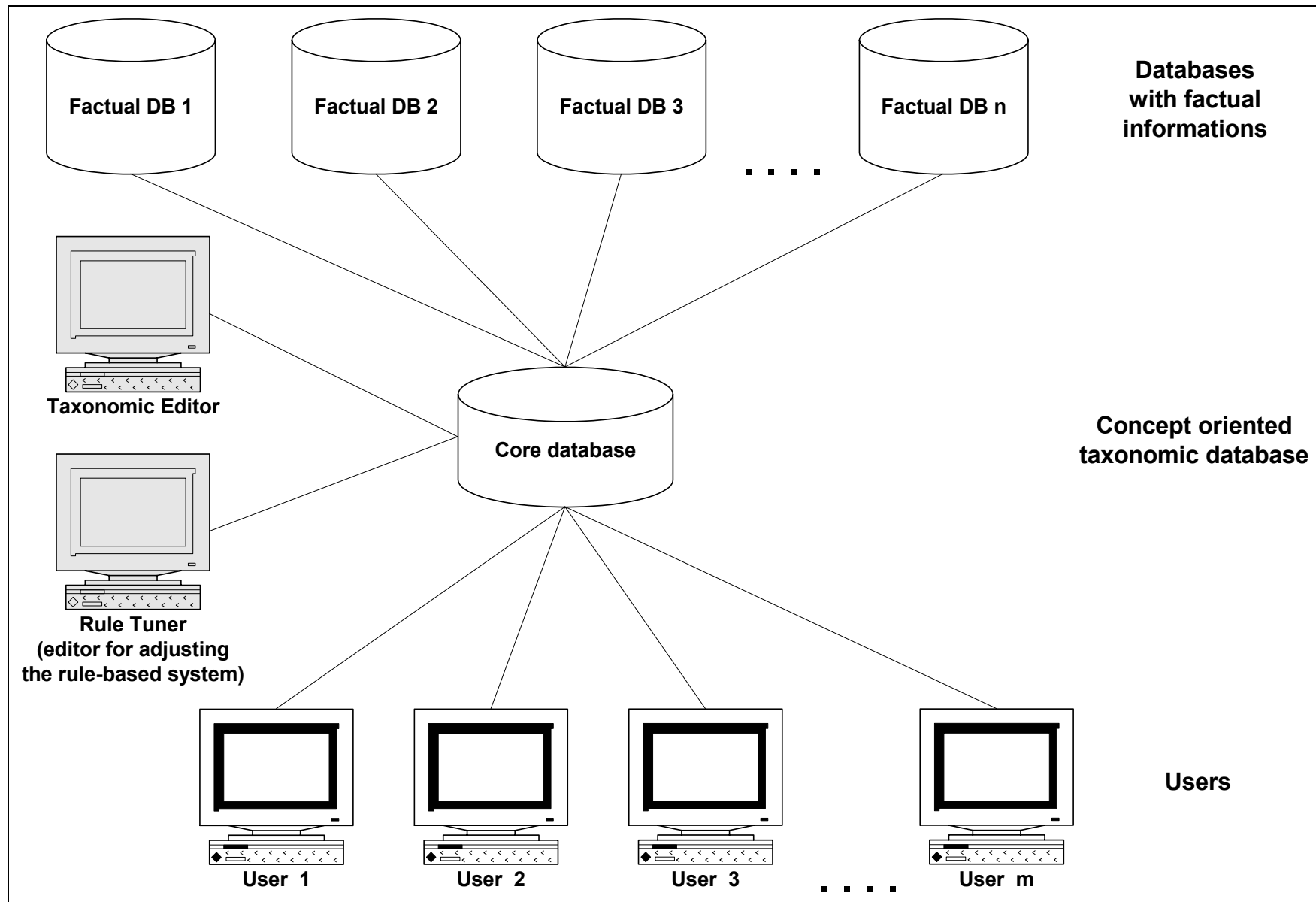
## **1. Introduction**

The architecture of the entire MoReTax system is centered on a core database, which in essence stores potential taxa, the relationships between them, and inference rules. Connections with this core database are done through the "World Wide Web" in order to achieve a decentralized and platform-independent use of the system. Important system components include:

- The databases, which are the sources for the potential taxa included in the core database and which also contain the factual information linked to them.
- The "taxonomic editor", which enables experts to add or edit the taxonomic or nomenclatural data and assign relationships between potential taxa within the core database.
- The "rule tuner", which enables system managers to undertake rule adjustments.
- A User interface for query generation and result presentation.

Figure 1 sketches the architecture of such a system with World Wide Web interface.

Figure 1: Architecture of the system



For the last three system components user-interfaces must be created in order to make it possible to edit the taxonomic data and the concept relationships between potential taxa ("taxonomic editor"), to adjust the rules by giving values to parameters ("rule tuner"), and to handle queries from users. The communication between the different users ("clients") and the core database ("server") should be operating system independent as far as users are concerned. This implies that apart from a web browser no special software will be necessary on the client side. Appropriate specifications for these interfaces are future deliverables within the present project.

The purpose of the present document is to specify interfaces for communication between the databases containing factual information (first system component mentioned above) and the concept-based taxonomic core. This communication occurs in two different processes:

- On importing or updating taxonomic information from factual databases into the taxonomic core (e.g. initiated by the taxonomic editor).
- On retrieval of factual information linked to taxon names in databases, responding to enquiries based on user queries expanded into sub graphs<sup>1</sup> of potential taxa.

## 2. Database interfaces

### Retrieving data from factual databases

Since queries are executed from the core database to retrieve data from factual databases, in this process the former plays the role of the "client" and the latter the role of "servers".

The client uses a module to dynamically create the queries to be transferred to the servers providing information. The queries will be XML encoded and send by means of an http-request (e.g. "post").

A so-called "wrapper" (a piece of software, which translates the XML encoded queries into the native query language of the factual data server – e.g. as SQL -) intercepts these XML queries on the server side.

If wrappers are to be independent from the respective structure of the databases with factual information, then there ought to be common views against which the queries will be formulated. This means that all databases with factual information must make a part of their data available through such views for the core system.

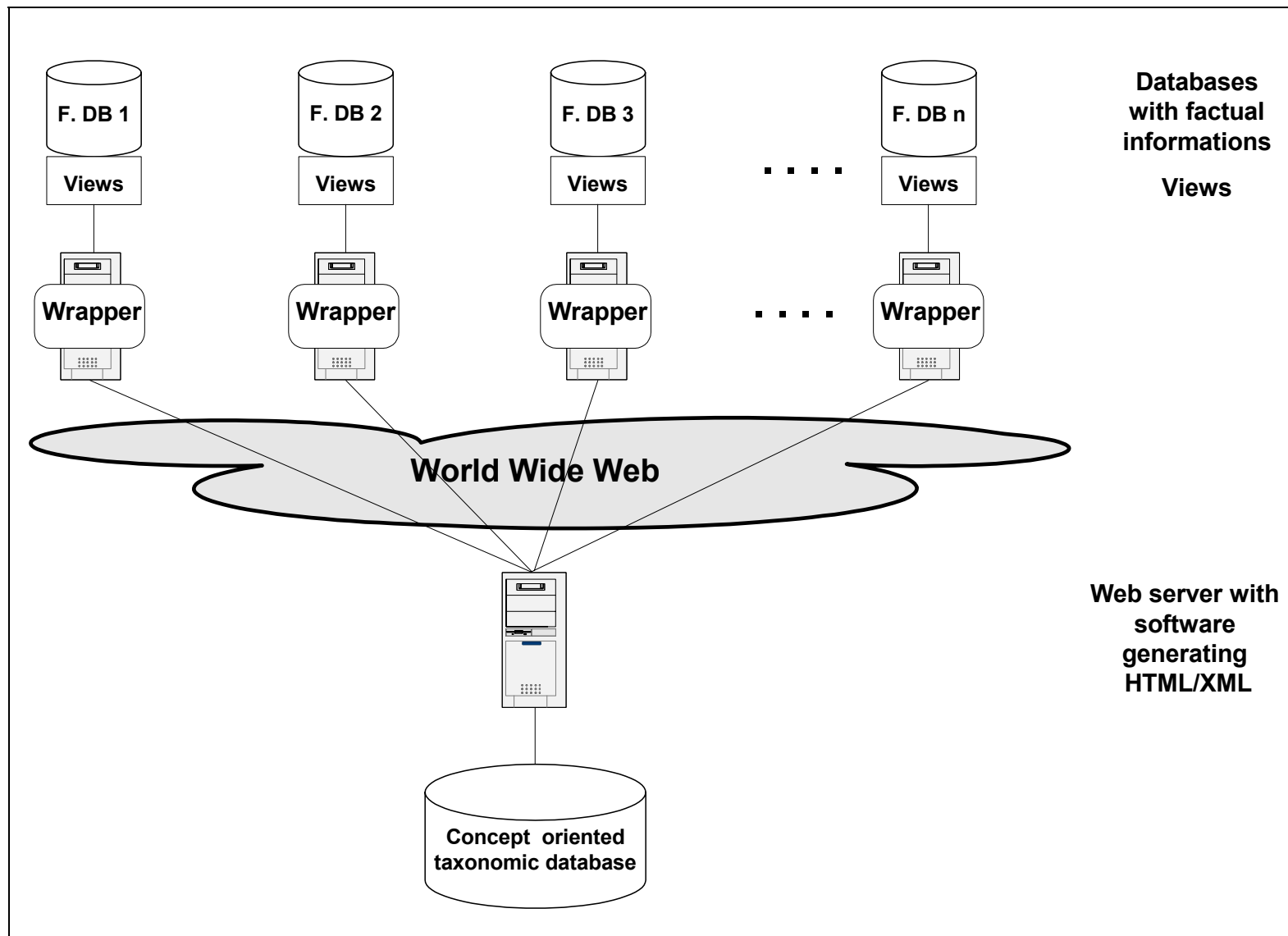
The wrapper is also responsible for returning the query results (e.g. the content of record sets) embedded into an XML document to the client (core database). There it will be parsed, integrated, and, if needed, transformed into the user interface format (e.g. html).

Figure 2 illustrates these database relations.

---

<sup>1</sup> Set of potential taxa which are connected to each other through edges and allowed paths.

Figure 2: Relations between databases



### **The import and update interface**

In order to allow for the clarification of concept relationships, the names used in the non-core databases must be present in the core, preferably including a clarification of the concept represented by the name (potential taxon). As a side effect, this also increases the efficiency of the system, because database will not be queried for names for which they do not hold information.

Consequently, an interface is needed for the purpose of importing and updating data from the databases containing factual information. This interface applies to the taxon names domain, to the nomenclature references domain and eventually to the "potential taxa" references domain ("sec. xyz")<sup>2</sup>. For this purpose we need a view on factual databases. In order to suit possible different database structures this view must be able to contain these data in an atomized and also in a not atomized form. It should be possible to offer further information with some notes fields and to check by means of date fields whether the same data have already been imported. We present such a view, which we call "V\_PotentialTaxa". The fields of this view refer to the three domains we mentioned above. Fields of the third domain will only be filled if potential taxa already exist in the factual database, otherwise the empty fields of this domain will be interpreted so that the factual database itself will be considered as the reference.

### **The factual information interface**

In order to retrieve from each factual database the facts, which are linked to taxon names or even to potential taxa, the appropriate view on these databases should have a structure, which does not only just indicate the facts, but also enables the system to know if the fact implies a new concept of the taxon, to evaluate results of calculations and to restrict output. In the view "V\_Fact" we took in account these requirements.

---

<sup>2</sup> See: [http://www.bgbm.fu-berlin.de/BioDivInf/Projects/MoreTax/standard\\_liste\\_en.htm](http://www.bgbm.fu-berlin.de/BioDivInf/Projects/MoreTax/standard_liste_en.htm)

## View V\_PotentialTaxa

Field name	Field type	Field description
PtaxonId	Int	Unique identifier for the view
NameFk	Int	Unique identifier of the taxon name
FullName	String	Complete latin name string including author strings
Name	String	Complete latin name string
RankAbbrev	String	Abbreviation of rank (e.g. 'subsp.')
SupragenericName	String	Name of taxon with a rank above genus
Genus	String	Genus name
GenusSubdivisionEpi	String	Genus subdivision epithet
SpeciesEpi	String	Species epithet
InfraSpeciesEpi	String	Infraspecific epithet
AuthorTeam	String	Complete author's team string
HybridFormulaFlag	Boolean	Indication of whether this name is a hybrid formula (unnamed hybrid)
MonomHybFlag	Boolean	Indication of monomial hybrid (e.g. generic hybrid)
BinomHybFlag	Boolean	Indication of binomial hybrid
TrinomHybFlag	Boolean	Indication of trinomial hybrid
CultivarGroupName	String	Cultivar group designation
CultivarName	String	Cultivar epithet
NomStatus	String	Nomenclatural status (e.g. "nom. cons.")
NameCreated_When	Date	Date and time when taxon name was created/updated
NameNotes	String	Remarks and notes for further details on taxon name
NomRefFk	Int	Unique identifier for the nomenclatural reference
NomFullRef	String	Complete nomenclatural reference citation
NomTitle	String	Full title of the nomenclatural reference
NomTitleAbbrev	String	Abbreviation of full title as used in nomenclatural citations (e.g. 'Sp. Pl. 2')
NomEdition	String	Edition string (e.g. 'ed.2') for nomenclatural reference
NomVolume	String	Volume as a string (e.g. '33' or 'ser.3, 2') for nomenclatural reference
NomSeries	String	Publication series the reference belongs to, for nomenclatural reference
NomRefYear	String	Year of publication as a string, for nomenclatural reference
NomPageString	String	Pages of an article or other part in a book or journal for nomenclatural reference
NomDetails	String	String of nomenclatural reference details such as page, page range, no. of figure, etc.

# R&D Project " Model for taxonomical-systematic assignment of species " (MoReTax)

NomISSN	String	ISSN code of the publication, for nomenclatural reference
NomISBN	String	ISBN code of the publication, for nomenclatural reference
NomPublicationTown	String	Place of publication, for nomenclatural reference
NomPublisher	String	Publisher string, for nomenclatural reference
NomCreated_When	Date	Date and time when nomenclatural reference was created/updated
NomNotes	String	Remarks and notes for further details on nomenclatural reference
RefFk	Int	Unique identifier for the reference
FullRef	String	Complete reference citation
Title	String	Full title of the reference
TitleAbbrev	String	Abbreviation of full title as used in nomenclatural citations (e.g. 'Sp. Pl. 2')
RefAuthorTeam	String	Complete author's team string of the reference
Edition	String	Edition string (e.g. 'ed.2') for reference
Volume	String	Volume as a string (e.g. '33' or 'ser.3, 2') for reference
Series	String	Publication series the reference belongs to, for reference
RefYear	String	Year of publication as a string, for reference
PageString	String	Pages of an article or other part in a book or journal
Details	String	String of reference details such as page, page range, no. of figure, etc.
ISSN	String	ISSN code of the publication
ISBN	String	ISBN code of the publication
PublicationTown	String	Place of publication
Publisher	String	Publisher string
URL	String	Full URL ('http://...') if the data source is already accessible on the www
ThesisFlag	Boolean	Indicating if the reference is a thesis
RefDepositedAt	String	Place where the reference is located (e.g. the library where a manuscript was deposited)
RefCategory	String	Nature of the reference (e.g. database)
InformalRefCategory	String	Informal reference category
IsPaper	Boolean	Indication that the reference is printed publication or not
RefCreated_When	Date	Date and time when reference was created/updated
RefNotes	String	Remarks and notes for further details on reference
Status	String	Status as per the reference (e.g. "accepted", "synonym", etc.)
Created_When	Date	Date and time when potential taxon record was created/updated
Notes	String	Remarks and notes for further details on potential taxon

View V\_Fact

Field name	Field type	Field description
FactId	Int	Unique identifier for the record
Fact	String	Factual information linked to the potential taxon
FactCategory	String	Category of the factual information (e.g. "ecology")
Scope	String	Validity scope of the factual information (default is "fully applicable")
AccessRestriction	String	Access restriction for the factual information (default is "unrestricted")
PtaxonFk	Int	Unique identifier of the potential Taxon the fact is linked to
IsConceptRelevant	Boolean	Indication whether this fact is relevant for the concept about the taxon
Created_When	Date	Date and time when factual information was created/updated
Notes	String	Remarks and notes for further details on factual information



### **Retrieving data from the core database**

In a further phase it can be thought about services, which the core database could offer to the other databases. This would mean that a new view, in which the needed information is held, should be created, now the other way around, on the core database. It also requires of course a new wrapper. Concrete steps in this sense can only be undertaken, when requirements of the databases that will communicate with the core database are known.