Project no. 018340

Project acronym: EDIT

Project title: Toward the European Distributed Institute of Taxonomy

Instrument: Network of Excellence

Thematic Priority: Sub-Priority 1.1.6.3: "Global Change and Ecosystems"

C5.113 CDM power user interface design study and pilot implementation

Due date of component: Month 41
Actual submission date: Month 41

Start date of project: 01/03/2006 Duration: 5 years

Organisation name of lead contractor for this component: 9 FUB-BGBM

Revision: final

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

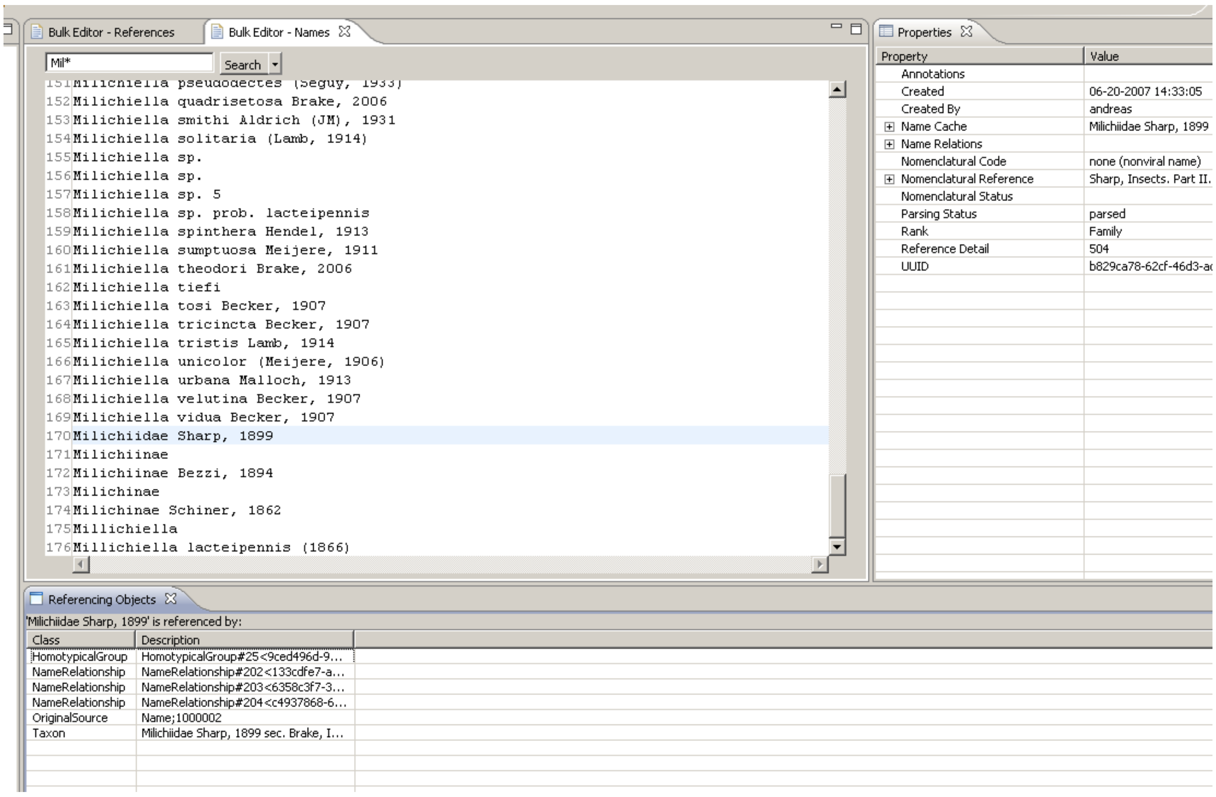
CDM power user interface design study and pilot implementation

The power user of a taxonomic data storage model is a non-technical user responsible for the integrity of the data. While the normal user creates data – names, references, authors, etc. – as part of the workflow of editing a taxon, the power user can create, edit, and delete data outside of this workflow. Moreover, the normal user should be restricted from editing certain shared data. For instance, if a name within a shared data source is referenced by two different taxa, changing the name affects both taxa; if the taxa are maintained by separate users, the consequences may be disruptive. The power user is aware of these consequences.

The design of the power user interface is largely based on a workflow observed working with the Berlin Model relational database at the Botanic Garden and Botanical Museum Berlin-Dahlem. The first step in most projects using the Berlin Model is a mass import of unstructured data, e.g. from a monographic text or a checklist table. Parsing software atomizes the data in the text file into its constituent parts for storage in the database. A main task of the parser is duplicate detection, but for a number of reasons – misspellings, format variations, and abbreviations – the import process inevitably results in a certain amount of cases where multiple records for the same object exist in the database.

The problem of duplicate references in the Berlin Model-based Euro+Med Plantbase project (http://ww2.bgbm.org/EuroPlusMed/) was resolved by preparing output in the form of an Excel spreadsheet containing all references listed in alphabetical order. The taxonomist then marked which groups of references belonged together, and which reference should be considered the correct one in each group. An algorithm then processed the marked spreadsheet and grouped the duplicates in the database. This solution proved to be quick and cost-effective, and struck a satisfactory balance between taxonomist and programmer time resources.

The pilot implementation of the Taxonomic Editor's power user interface (see Figure 1) is mainly based on this experience. It consists of a search field and a list editor. The list is populated when a search is executed. The property sheet view familiar from taxon editing is re-used here to display and edit details of the individual object focussed; changes made in the property sheet are reflected immediately in the list. New instances of the class currently being edited (e.g. name, reference) can be created here as well.



```
151Milichiella pseudodectes (Seguy, 1933)
152Milichiella quadrisetosa Brake, 2006
153Milichiella smithi Aldrich (JM), 1931
154Milichiella solitaria (Lamb, 1914)
155Milichiella sp.
156Milichiella sp.
157Milichiella sp. 5
158Milichiella sp. prob. lacteipennis
159Milichiella spinthera Hendel, 1913
160Milichiella sumptuosa Meijere, 1911
161Milichiella theodori Brake, 2006
162Milichiella tiefi
163Milichiella tosi Becker, 1907
164Milichiella tricincta Becker, 1907
165Milichiella tristis Lamb, 1914
166Milichiella unicolor (Meijere, 1906)
167Milichiella urbana Malloch, 1913
168Milichiella velutina Becker, 1907
169Milichiella vidua Becker, 1907
170Milichiidae Sharp, 1899
171Milichiinae
172Milichiinae Bezzi, 1894
173Milichinae
174Milichinae Schiner, 1862
175Millichiella
176Millichiella lacteipennis (1866)
```

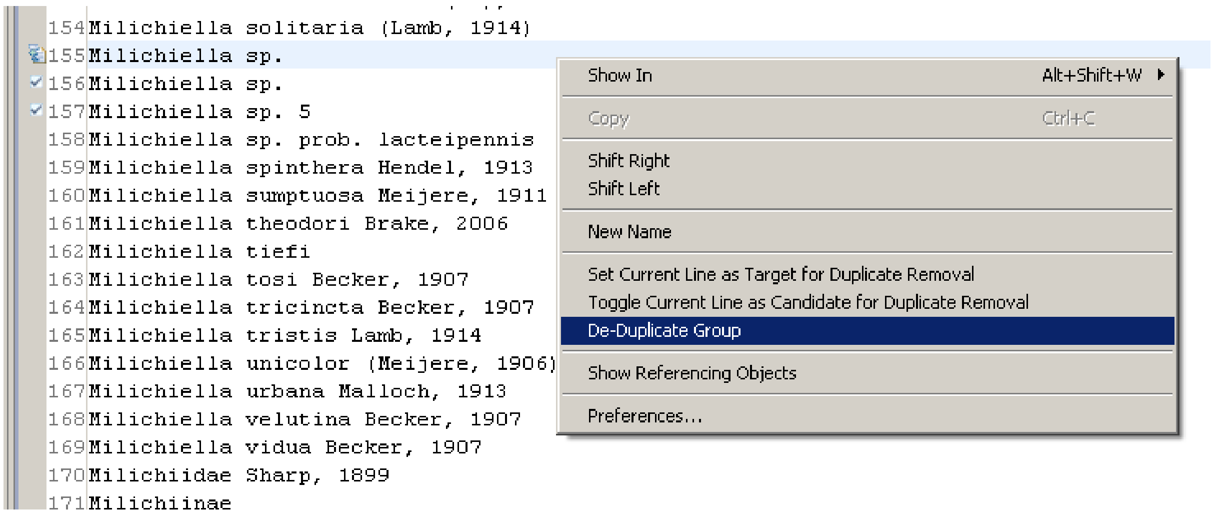
Property	Value
Annotations	
Created	06-20-2007 14:33:05
Created By	andreas
⊞ Name Cache	Milichiidae Sharp, 1899
⊞ Name Relations	
Nomenclatural Code	none (nonviral name)
⊞ Nomenclatural Reference	Sharp, Insects. Part II.
Nomenclatural Status	
Parsing Status	parsed
Rank	Family
Reference Detail	504
UUID	b829ca78-62cf-46d3-a

'Milichiidae Sharp, 1899' is referenced by:

Class	Description
HomotypicalGroup	HomotypicalGroup#25<9ced496d-9...
NameRelationship	NameRelationship#202<133cdfe7-a...
NameRelationship	NameRelationship#203<6358c3f7-3...
NameRelationship	NameRelationship#204<c4937868-6...
OriginalSource	Name;1000002
Taxon	Milichiidae Sharp, 1899 sec. Brake, I...

Figure 1: The power user interface ("Bulk Editor")

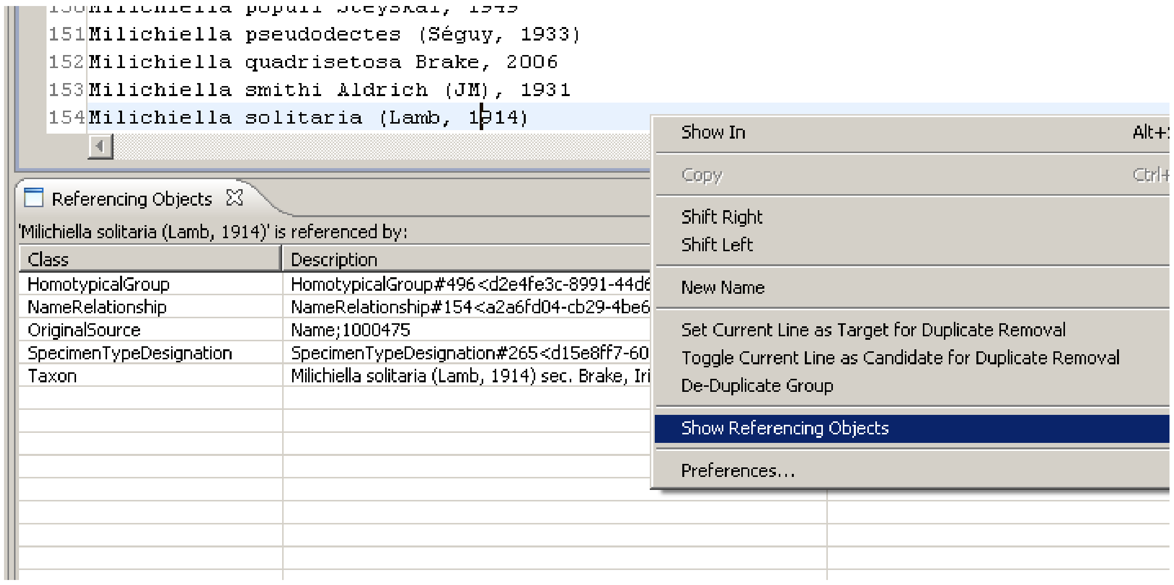
The workflow for removal of duplications is as follows: the user identifies all duplicates, and chooses one of them as a target. When he or she executes the duplicate removal command, all but the target are removed from the list (see Figure 2). Behind the scenes within the data source, all information associated with the duplicates is transferred to the target, and then the duplicates are deleted.



```
154Milichiella solitaria (Lamb, 1914)
155Milichiella sp.
156Milichiella sp.
157Milichiella sp. 5
158Milichiella sp. prob. lacteipennis
159Milichiella spinthera Hendel, 1913
160Milichiella sumptuosa Meijere, 1911
161Milichiella theodori Brake, 2006
162Milichiella tiefi
163Milichiella tosi Becker, 1907
164Milichiella tricincta Becker, 1907
165Milichiella tristis Lamb, 1914
166Milichiella unicolor (Meijere, 1906)
167Milichiella urbana Malloch, 1913
168Milichiella velutina Becker, 1907
169Milichiella vidua Becker, 1907
170Milichiidae Sharp, 1899
171Milichiinae
```

Figure 2: the user has identified the names in lines 156 and 157 as duplicates of the name in line 155, and is about to execute "De-Duplicate Group".

Because the change may imply changes to a number of objects, the user has to be able to view all the locations within the data referenced by a possible duplicate (see Figure 3). For instance, this view could show all the descriptive elements and names that cite a particular reference, or all relationships using a particular name (synonymy etc.).



```
150 Milichiella populi Steyskal, 1949
151 Milichiella pseudodectes (Séguy, 1933)
152 Milichiella quadrisetosa Brake, 2006
153 Milichiella smithi Aldrich (JM), 1931
154 Milichiella solitaria (Lamb, 1914)
```

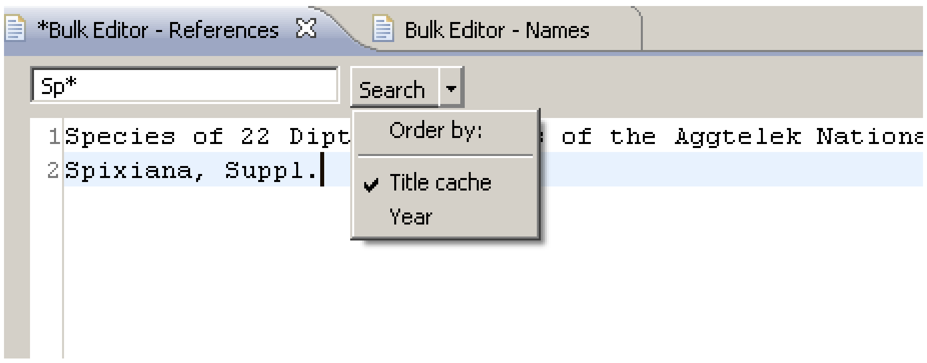
Referencing Objects

'Milichiella solitaria (Lamb, 1914)' is referenced by:

Class	Description
HomotypicalGroup	HomotypicalGroup#496<d2e4fe3c-8991-44d6
NameRelationship	NameRelationship#154<a2a6fd04-cb29-4be6
OriginalSource	Name;1000475
SpecimenTypeDesignation	SpecimenTypeDesignation#265<d15e8ff7-60
Taxon	Milichiella solitaria (Lamb, 1914) sec. Brake, Iri

Figure 3: "Referencing Objects" lists all references to "*Milichiella solitaria* (Lamb, 1914)" in the current data source

The interface has been designed as generically as possible, so that it is easy to configure it for different data types. However, that configuration involves the implementation of Java interfaces, and thus can not be done on-the-fly by the non-technical user. Configuration includes sorting options (see Figure 4), the format used to display an object in the list, data type-specific interaction with the data source, and the display of the object in the property sheet.



```
1 Species of 22 Dipt        of the Aggtelek Nationa
2 Spixiana, Suppl.
```

Figure 4: sorting options specific to References.